

## Gestión de la deuda de procesos en equipos de desarrollo de software: una revisión sistemática de la literatura

### Process debt management in software development teams: a systematic review of the literature

Astrid Carolina Cruz Trochez<sup>1</sup>, Braian Alexis Bastidas Diaz<sup>2</sup>, César Jesús Pardo Calvache<sup>3</sup>

<sup>1</sup> Facultad de Ingeniería Electrónica y Telecomunicaciones, Universidad del Cauca, GTI Research Group, Calle 5 # 4-70, 19001, Popayán, Colombia. <https://orcid.org/0009-0007-7905-9820>

<sup>2</sup> Facultad de Ingeniería Electrónica y Telecomunicaciones, Universidad del Cauca, GTI Research Group, Calle 5 # 4-70, 19001, Popayán, Colombia. <https://orcid.org/0009-0009-2141-0301>

<sup>3</sup>PhD and MSc in Computer Science. Facultad de Ingeniería Electrónica y Telecomunicaciones, Universidad del Cauca, GTI Research Group, Calle 5 # 4-70, 19001, Popayán, Colombia.  
<https://orcid.org/0000-0002-6907-2905>

[brabadi@unicauca.edu.co](mailto:brabadi@unicauca.edu.co)

Cite this article as: A. Cruz Trochez, B. Bastidas Diaz, C. Pardo Calvache “*Gestión de la deuda de procesos en equipos de desarrollo de software: una revisión sistemática de la literatura*”, Prospectiva, Vol 24, Nº 1, 2026.

Recibido: 14/02/2025 / Aceptado: 11/09/2025

<http://doi.org/10.15665/rp.v24i1.3741>

#### RESUMEN

La deuda de proceso se refiere a la ocurrencia de deficiencias en el diseño o infraestructura de procesos subóptimos que; aunque puedan ser beneficiosos en el corto plazo, pueden generar un impacto negativo en el largo plazo para las partes interesadas en el proceso, por ejemplo: sobrecostos, retrasos, desmotivación del equipo, entre otros. Se analizaron varias soluciones propuestas en la literatura, incluyendo marcos de evaluación, herramientas de automatización y modelos predictivos, así como los desafíos inherentes a la gestión de la deuda de procesos, tales como: falta de estandarización, dificultad de identificación, entre otros. También se observó la interrelación entre la deuda de procesos y otros tipos de deuda, como: la deuda técnica, social y de documentación, subrayando la necesidad de enfoques más integrales. Las conclusiones sugieren que se requiere un mayor desarrollo y validación empírica de las mismas, así como un esfuerzo concertado para alinear las prácticas de la industria con los modelos teóricos emergentes. Esta revisión ofrece a investigadores y profesionales la oportunidad de reconocer los artefactos, las lagunas y las técnicas de investigación presentes, lo que les permitirá analizar, perfeccionar, mejorar o reforzar sus propias aportaciones, así como profundizar en las ya establecidas.

**Palabras clave:** Procesos de software, optimización de procesos, estrategias, gestión, deuda de proceso, artefactos.

**ABSTRACT**

*Process debt refers to the occurrence of deficiencies in the design or infrastructure of suboptimal processes that; although they may be beneficial in the short term, can generate a negative impact in the long term for the stakeholders in the process, for example: cost overruns, delays, team demotivation, among others. Several solutions proposed in the literature were analyzed, including evaluation frameworks, automation tools and predictive models, as well as the challenges inherent to process debt management, such as: lack of standardization, difficulty of identification, among others. The interrelationship between process debt and other types of debt, such as: technical, social and documentation debt, was also noted, highlighting the need for more comprehensive approaches. The findings suggest that further development and empirical validation is required, as well as a concerted effort to align industry practices with emerging theoretical models. This review offers researchers and practitioners the opportunity to recognize artifacts, gaps and research techniques present, which will allow them to analyze, refine, improve or strengthen their own contributions, as well as deepen those already established.*

**Keywords:** Software processes, process optimization, strategies, management, process debt, artifacts.

**1. INTRODUCCIÓN**

En el desarrollo de software, la eficiencia y calidad de los procesos son fundamentales para el éxito organizacional [1]. La deuda de procesos, entendida como la acumulación de ineficiencias y compromisos que afectan la productividad y calidad del software [2], ha sido menos investigada que otros tipos de deuda como la técnica, social o documental, lo que crea un vacío en su comprensión y gestión.

Esta revisión sistemática de la literatura (RSL) aborda la escasez de estudios sobre la deuda de procesos, explorando metodologías para identificarla, gestionarla y entender su impacto en el ciclo de vida del desarrollo de software. Se analiza la falta de una cultura organizativa orientada a la mejora continua, lo cual puede generar resistencia al cambio y dificultar la identificación y resolución de problemas relacionados con la deuda de procesos [3]. Además, su acumulación afecta la calidad del producto final, la moral del equipo y la eficiencia operativa [1], y la falta de consenso sobre la terminología y métricas resalta la necesidad de marcos teóricos sólidos y herramientas prácticas [2].

La identificación de barreras técnicas, organizativas y culturales es también esencial, ya que limitan la adopción de soluciones y contribuyen a la acumulación de deuda, comprometiendo la sostenibilidad y competitividad organizacional [4]. Esta RSL busca llenar ese vacío, proporcionando una visión integral de las causas, efectos, prácticas y herramientas asociadas a la deuda de procesos, con el fin de contribuir a la optimización de procesos y mantener la competitividad a largo plazo.

**2. METODOLOGÍA**

La metodología de esta RSL sobre la deuda de procesos en el desarrollo de software siguió el enfoque sistemático propuesto por Kitchenham et al. [5], que incluye la definición de objetivos, el desarrollo de un protocolo detallado, la búsqueda exhaustiva en bases de datos académicas y la selección de estudios primarios según criterios de inclusión y exclusión.

**2.1. Definir objetivos de investigación**

El objetivo principal de esta RSL fue responder preguntas clave sobre la DP, evaluando estudios sobre su identificación, gestión y mitigación, se analizó las soluciones propuestas, tipos de investigación, beneficios, causas, efectos y desafíos, organizando la información cronológicamente para identificar tendencias y desafíos futuros. La Tabla 1 presenta los objetivos de investigación propuestos para esta RSL; el objetivo 1 se tratará en la sección III, donde se detallará la ejecución de esta RSL. Los objetivos 2 y 3 se abordaron

mediante el establecimiento de preguntas de investigación, que se presentan en el Cuadro 2 y a las que se dará respuesta en la Sección IV

**Tabla 1.** Objetivos de investigación.

**Table 1.** Research objectives.

ID	Objetivos de Investigación
<b>OB1</b>	Categorizar los estudios e investigaciones considerando criterios demográficos, con el propósito de evaluar el interés científico de la comunidad académica respecto al tema de Deuda de Procesos, causas, efectos y procedimientos empleados para gestionar y controlar su presencia en el contexto de desarrollo de software.
<b>OB2</b>	Analizar la diversidad de estudios identificados y relacionados con la deuda de procesos, basándose en el esquema de clasificación propuesto por Wieringa et al. [6] y mediante la definición de criterios de calidad, seleccionar los artículos primarios considerando factores como claridad, suficiencia, relevancia y contribución investigativa.
<b>OB3</b>	Recopilar información actualizada, relevante y validada sobre las soluciones propuestas en la literatura para gestionar la deuda de procesos en el desarrollo de software, esto permitirá identificar las tendencias actuales, así como anticipar retos y desafíos futuros relacionados con este tema.

## 2.2. Especificación de las preguntas de investigación

En esta RSL, la construcción de las preguntas de investigación siguió el enfoque orientado a objetivos (GO) propuesto por Kitchenham et al. [5]. Este enfoque permitió formular preguntas específicas alineadas con los objetivos de la investigación. Tras una revisión exhaustiva de la literatura, se identificaron las lagunas y trabajos futuros relacionados con la deuda de procesos en el desarrollo de software. Las preguntas se agruparon en cuatro categorías:

- **Tipos de solución:** PI1 se enfoca en identificar, clasificar y comprender las ventajas y retos de las soluciones propuestas en la bibliografía.
- **Síntesis de información actualizada:** PI3 y PI4 se centran en sintetizar y evaluar la información existente, clasificar el tipo de investigación y evaluar la madurez del campo.

**Tabla 2.** Preguntas de investigación.

**Table 2.** Research questions.

Nº	Categoría	Preguntas de Investigación	Motivación	OB.
<b>PI1</b>	Tipo de Solución	¿Cuáles son los tipos de soluciones propuestas en la literatura que permitan analizar y/o gestionar la deuda de procesos en el desarrollo de software?	Identificar el tipo de contribución en una o varias de las siguientes categorías: (i) definición conceptual, (ii) causas, efectos, impactos y limitaciones, (iii) métodos o técnicas de evaluación, (iv) herramientas tecnológicas, (v) validación en la industria, (vi) metodologías de documentación y (vii) otros.	OB1
<b>PI2</b>	Síntesis de la Información actualizada y relevante	¿Cuál es el nivel de madurez y/o desarrollo actual en la literatura con respecto a los conceptos y enfoques de gestión de la deuda de procesos en el desarrollo de software?	Evaluar el estado actual de la literatura en términos de madurez, desarrollo de los conceptos y enfoques relacionados con la gestión de la deuda de procesos e identificar áreas de oportunidad y posibles direcciones futuras de investigación.	OB3
<b>PI3</b>		¿Cuáles son los beneficios, desafíos, causas, impacto, efectos y/o consecuencias asociadas con la investigación sobre la deuda	Determinar los beneficios, desafíos, causas, impacto, efectos y/o consecuencias que conlleva para las empresas productoras de software la reducción de la deuda de procesos en los proyectos que desarrollan.	OB3

		de procesos en el desarrollo de software?		
--	--	---	--	--

Acrónimos utilizados: OB: Objetivo

### 2.3. Criterios de evaluación de la calidad de las preguntas de investigación

Para acceder al contenido completo puede visitar: <https://shorturl.at/H2COw>.

### 2.4. Diseño de la RSL

La búsqueda manual comenzó con la revisión de literatura gris, proporcionados por expertos en Ingeniería de Software, para obtener un primer acercamiento a los conceptos y términos relacionados con la DP en el desarrollo de software, así como identificar los problemas y retos actuales. A partir de las palabras clave obtenidas, se definió la cadena de búsqueda básica: "process AND software AND 'process debt'", aplicada a las bases de datos IEEE Xplore, Science Direct, Scopus, Google Scholar, Springer Link, ACM y Web of Science. Se aplicaron los siguientes filtros: (i) artículos en inglés, (ii) artículos en áreas como Informática, Ingeniería de Software y Desarrollo de Software, y (iii) artículos de revistas y congresos. La búsqueda se limitó a publicaciones entre 2013 y 2024, finalmente, la cadena de búsqueda se adaptó a las configuraciones específicas de cada base de datos, ver Tabla 7.

### 2.5. Criterios de inclusión y exclusión

Los criterios de inclusión (en adelante CI) y exclusión (en adelante CE) incidió en el fortalecimiento de la calidad académica de los estudios, lo cual fue fundamental para que el proceso de selección de los estudios fuera riguroso, coherente y transparente. En la Tabla 4 se detallan los criterios utilizados.

**Tabla 4.** Criterios de inclusión y exclusión.

**Table 4.** Inclusion and exclusion criteria.

<b>Id</b>	<b>Criterio de Inclusión</b>	<b>Id</b>	<b>Criterio de Exclusión</b>
CI1	Estudios cuyo núcleo temático esté relacionado con la deuda de procesos en el desarrollo de software.	CE1	Estudios que no proporcionen una definición clara y operativa de la deuda de procesos.
CI2	Investigaciones publicadas en idioma inglés y español.	CE2	Artículos que no incluyan una validación empírica de sus propuestas o teorías.
CI3	Estudios publicados en revistas indexadas, actas de congresos, y otros medios científicos sometidos a revisión por pares.	CE3	Estudios que traten el tema de la deuda de procesos de manera superficial, sin un análisis profundo o detallado.
CI4	Investigaciones recientes (de los últimos 10 años) que aborden la deuda de procesos en el contexto del desarrollo de software.	CE4	Artículos que no estén accesibles por restricciones de acceso o falta de disponibilidad.
CI5	Estudios que describan la metodología de manera clara y detallada, permitiendo la replicación y evaluación de los resultados.	CE5	Artículos duplicados.
CI6	Literatura gris.		

### 2.6. Evaluación de la relevancia y pertinencia de los estudios primarios

Para acceder al contenido completo puede visitar: <https://shorturl.at/H2COw>.

### 2.7. Estrategia de extracción de datos y métodos de síntesis

Para acceder al contenido completo puede visitar: <https://shorturl.at/H2COw>.

### 3. EJECUCIÓN DE LA RSL

Se realizaron siete iteraciones, una para cada base de datos utilizada en la búsqueda: Google Scholar, Scopus, Science Direct, SpringerLink, IEEEXplore, Web of Science y ACM, además de la literatura gris. La cadena de búsqueda se adaptó a las características de cada base de datos (ver Tabla 7). La Tabla 8 muestra el esquema de clasificación de los estudios encontrados y los relevantes. Los estudios primarios seleccionados se presentan en la Tabla 9, junto con su título, para que el lector pueda realizar una búsqueda más detallada.

**Tabla 7.** Adaptación de la cadena de búsqueda en las bases de datos.

**Table 7.** Adaptation of the search string in the databases.

ID	Cadena de Búsqueda	Bases de Datos
CB1	process AND software AND “process debt”	Google Scholar
CB2	(ALL ( process ) AND ALL ( variability ) OR ALL ( divergence ) OR ALL ( inefficient ) OR ALL ( suboptimal ) OR ALL ( sub-optimal ) OR ALL ( ineffective ) OR ALL ( dependencies ) OR ALL ( software ) AND ALL ( “Process Debt” ) )	Scopus
CB3	process AND (variability OR divergence OR inefficient OR suboptimal OR ineffective OR dependencies) AND software AND “process debt”	Science direct
CB4	process AND (variability OR divergence OR inefficient OR suboptimal OR ineffective OR dependencies) AND software AND “process debt”	Springer link
CB5	(“Full Text & Metadata”: “process debt”) AND (“Full Text & Metadata”:software) AND (“Full Text & Metadata”:process)	IEEE Xplore
CB6	process (All Fields) and software (All Fields) and process debt (All Fields)	Web Of Science,
CB7	[All: process] AND [[All: variability] OR [All: divergence] OR [All: inefficient] OR [All: suboptimal] OR [All: sub-optimal] OR [All: ineffective] OR [All: dependencies]] AND [All: software] AND [All: “process debt”]	ACM

Acrónimos utilizados: CB: Cadena de Búsqueda.

#### 3.1. Evaluación de la relevancia y pertinencia

Para acceder al contenido completo puede visitar: <https://shorturl.at/H2COw>.

#### 3.2. Frecuencia y tipos de publicaciones realizadas sobre la deuda de procesos

Para acceder al contenido completo puede visitar: <https://shorturl.at/H2COw>.

### 4. RESULTADOS

La Tabla 11 muestra los resultados obtenidos para responder a las preguntas de investigación. Los resultados se presentan organizados en tres perspectivas clave: (i) tipos de solución, (ii) tipo de investigación y (iii) síntesis de la información actualizada y relevante.

**Tabla 11.** Preguntas de investigación abordadas por cada estudio primario.

**Table 11.** Research questions addressed by each primary study.

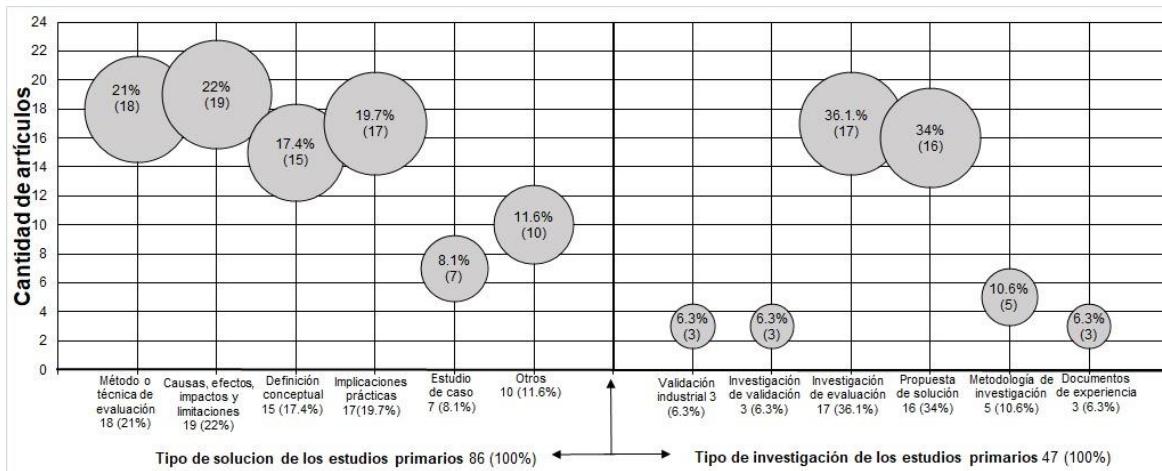
Id	Artículos																			
	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	A11	A12	A13	A14	A15	A16	A17	A18	A19	A20
PI1	X	X	X	X	X	X	-	X	X	X	X	-	X	-	X	X	X	X	X	X
PI2	X	X	X	X	X	-	-	-	X	-	-	-	X	-	-	-	-	-	-	-
PI3	X	X	X	X	x	X	-	-	-	-	-	-	X	-	-	X	-	-	-	-

#### 4.1. ¿Cuáles son los tipos de soluciones propuestas en la literatura que permitan analizar y/o gestionar la deuda de procesos en el desarrollo de software? (PI1)

Algunas soluciones como la adopción de buenas prácticas, documentación clara, control de calidad y planificación eficaz [14]. Para gestionarla, se recomienda realizar actividades como el seguimiento de la planificación del proyecto, la refactorización de procesos, el análisis de riesgos y la clarificación de requisitos. La falta de comunicación y los procesos mal definidos, son causas clave de la deuda de procesos [15]. Saeeda et al. [17] sugieren tres pasos fundamentales: (i) entender el proceso y su relación con la deuda de las personas, (ii) asignar tiempo para el seguimiento y la comunicación, y (iii) medir el proceso, evaluando equipos y clientes. La automatización de procesos, como la integración y entrega continuas, es vista como una solución clave para evitar la acumulación de deuda [20], aunque la falta de revisión de código puede generar costos a largo plazo [21].

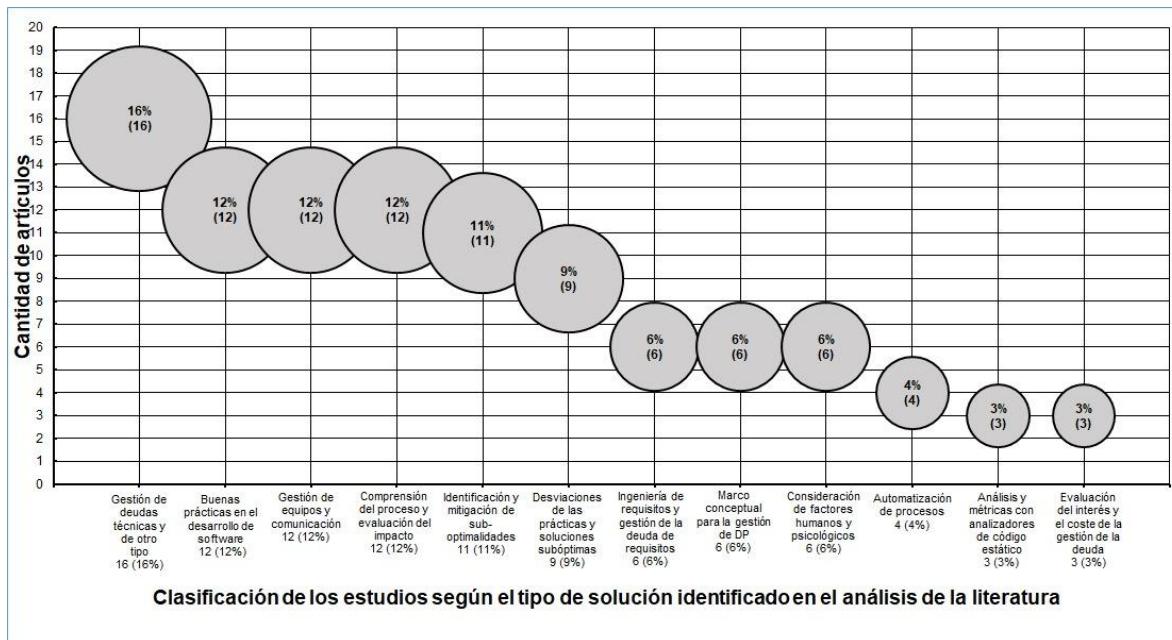
La Figura 3 presenta la clasificación de las soluciones según Wieringa et al. [6], mostrando que un 11,6% de los estudios abordan soluciones como herramientas tecnológicas, validación industrial, investigación de validación, artículos de opinión y de experiencia (A5, A9, A12, A13, A15, A1, A17, A11, A3, A19) [1]. El 21% de los estudios se enfoca en el método o técnica de evaluación, lo que es esencial para comprobar la efectividad de las soluciones implantadas [3]. Un 22% se dedica al análisis de las causas, efectos, impactos y limitaciones de la deuda de procesos, proporcionando un marco completo para entender su origen y consecuencias a corto y largo plazo [15]. La definición conceptual, que representa un 17,4% (A1, A2, A3, A5, A7, A8, A9, A10, A12, A13, A14, A15, A18, A19, A20), es crucial para estandarizar el lenguaje y los criterios de análisis, facilitando la comunicación y comparación entre estudios [26]. Las implicaciones prácticas, que representan un 19,7% (A1, A2, A3, A5, A6, A7, A8, A9, A10, A11, A12, A13, A14, A15, A17, A18, A19), son fundamentales para trasladar el conocimiento teórico a la práctica en empresas de software [14]. Los estudios de caso, que representan un 8,1% (A1, A2, A11, A13, A13, A17, A18, A19), proporcionan ejemplos concretos de cómo la deuda de procesos afecta proyectos específicos [28].

En la Figura 4 se muestra que la categoría de "Gestión de la deuda técnica y otros tipos" destaca con un 16% (A1, A3, A4, A5, A6, A7, A8, A10, A11, A12, A13, A14, A16, A17, A18, A20), seguida por "Buenas prácticas en el desarrollo de software", "Gestión de equipos y comunicación" y "Comprensión del proceso y evaluación del impacto", cada una representando un 12%. Esto refleja una tendencia hacia la gestión técnica, mientras que las buenas prácticas y la comunicación eficaz son esenciales para mitigar la deuda de procesos.



**Figura 3.** Clasificación de los artículos de acuerdo con las categorías definidas por Wieringa *et al.* [6].  
**Figure 3.** Classification of articles according to the categories defined by Wieringa et al [6].

El análisis temporal de los datos de la Figura 4 muestra que la categoría «Identificación y mitigación de suboptimidades» ha tenido un crecimiento constante desde 2016, alcanzando un 11% (A1, A2, A3, A5, A8, A9, A12, A14, A15, A18, A19). Esto refleja un aumento en la conciencia sobre la importancia de abordar no solo las deudas técnicas, sino también las suboptimidades en los procesos de desarrollo. En cambio, la categoría «Desviaciones de prácticas y soluciones subóptimas» ha representado un 9% (A1, A5, A6, A9, A13, A14, A15, A18, A19), siendo más relevante en los últimos años, especialmente en 2023 y 2024, lo que indica un interés creciente sobre cómo estas desviaciones afectan la calidad y eficiencia del desarrollo de software. Las categorías de «Ingeniería de requisitos y gestión de la deuda de requisitos» (6%, A3, A7, A8, A10, A11, A17), «Marco conceptual para la gestión de la DP» (6%, A1, A5, A8, A13, A14, A19) y «Consideración de factores humanos y psicológicos» (6%, A3, A5, A6, A15, A16, A18) indican una creciente inclusión de enfoques holísticos en la investigación sobre la deuda de procesos.



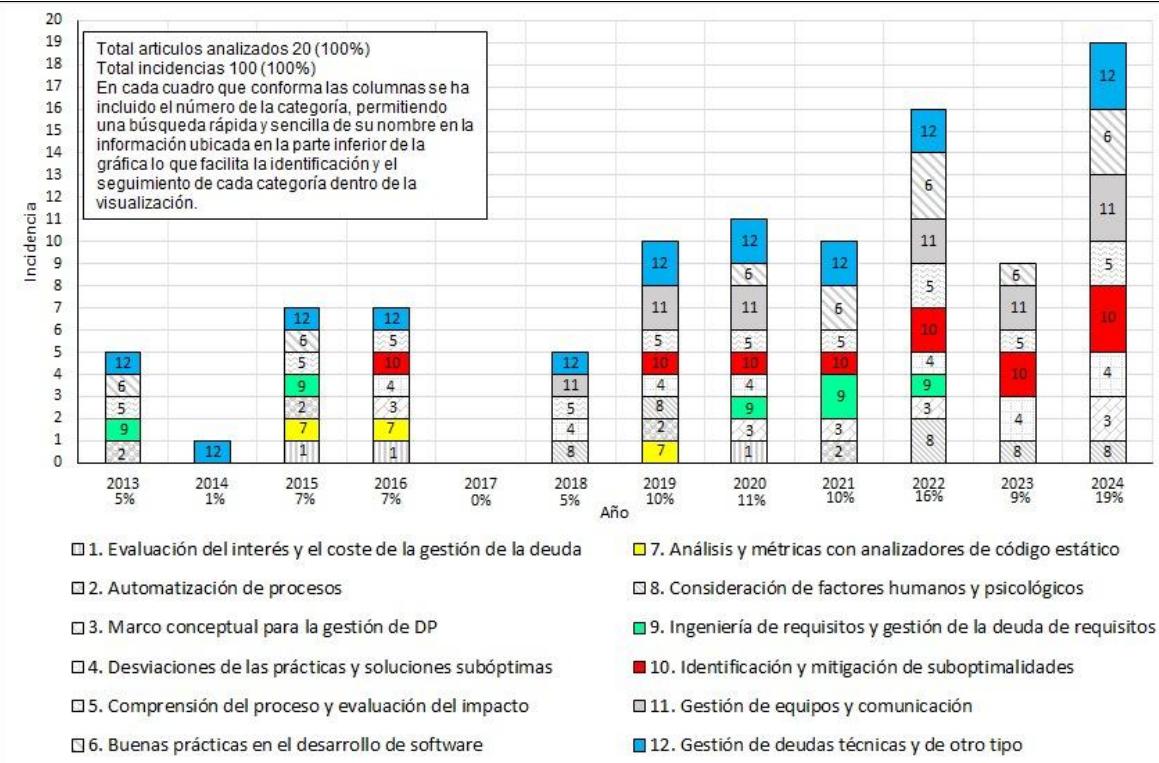
**Figura 4.** Clasificación de los estudios según el tipo de solución identificado en el análisis de la literatura.  
**Figure 4.** Classification of studies according to the type of solution identified in the literature analysis.

Por último, soluciones como «Automatización de procesos» 4% (A7, A8, A11, A18), «Ánalisis y métricas con analizadores estáticos de código» 3% (A7, A14, A18) y «Evaluación del interés y el coste de la gestión de la deuda» 3% (A1, A7, A14), han surgido en momentos puntuales, lo que refleja un interés menor. Esto podría indicar que, aunque estas soluciones no predominan en la literatura, están ganando atención y podrían ser áreas prometedoras para futuras investigaciones.

La Figura 5 muestra un aumento en el interés por la deuda de procesos, reflejando la evolución de las categorías de estudio a lo largo del tiempo. En los artículos seleccionados para esta RSL, no se encontró ninguno del año 2017, lo cual se destaca en las Figuras 2 y 5, evidenciando la ausencia de literatura relevante para ese año. Este vacío se debe a la aplicación de los Criterios de Inclusión (CI) y Criterios de Exclusión (CE) a los artículos, lo que permitió filtrar aquellos que no cumplían con los estándares del SLR. Se recomienda que futuros estudios exploren áreas poco investigadas, como la automatización de procesos y la consideración de factores humanos y psicológicos.

### Principales resultados de PI1

El desarrollo de software suele acumular deuda técnica y no técnica, lo que dificulta la eficacia y la calidad. Las estrategias para gestionar esta deuda incluyen documentación clara, control de calidad y comunicación eficaz. Al priorizar y abordar la deuda de forma sistemática, las organizaciones pueden reducir los riesgos y mejorar el rendimiento general. La investigación futura debe explorar soluciones emergentes como la automatización de procesos y los enfoques centrados en el ser humano, y los factores psicológicos, apuntan a futuras necesidades de investigación para apoyar el esfuerzo holístico de mitigación de la deuda.



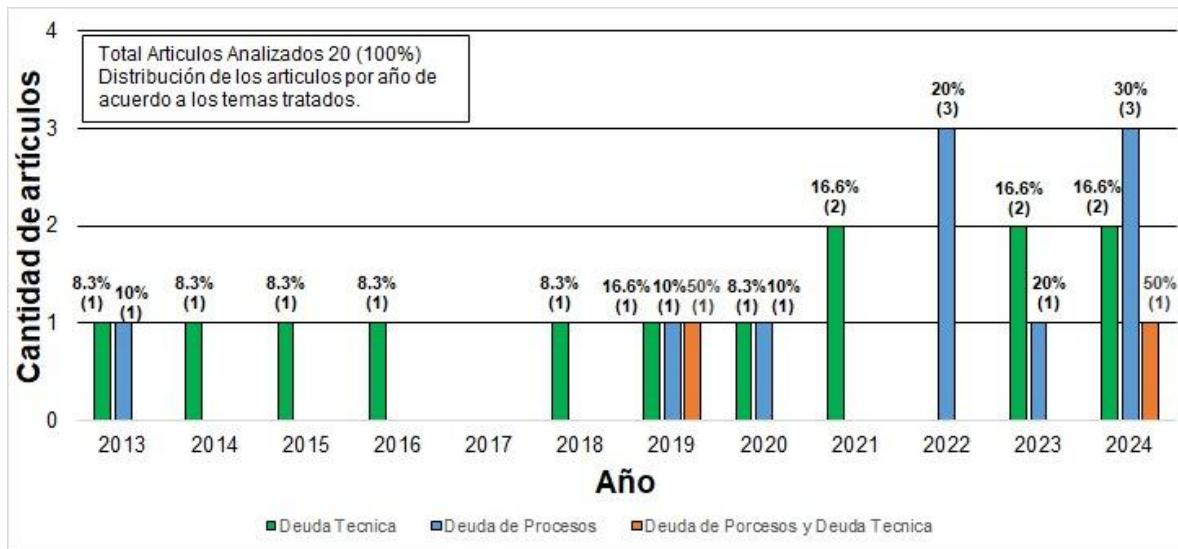
**Figura 5.** Distribución de temática por año.

**Figure 5.** Distribution of topics by year.

#### 4.2. ¿Cuál es el nivel de madurez y/o desarrollo actual en la literatura con respecto a los conceptos y enfoques de gestión de la deuda de procesos en el desarrollo de software? (PI2)

El nivel de madurez de la literatura sobre la deuda de procesos es variable, evidenciando tanto avances significativos como áreas en desarrollo, donde se encuentran aspectos como la investigación empírica, conceptos y enfoques, impactos de factores no técnicos y estrategias de mitigación. Algunos estudios han mejorado la comprensión y la gestión de la deuda de procesos, especialmente en entornos de desarrollo ágil a gran escala. Sin embargo, otros carecen de rigor, lo que dificulta la toma de decisiones informadas y subraya la necesidad de más investigación empírica, como se destaca en [17].

La Figura 6 muestra la distribución de los artículos primarios analizados en la ventana temporal consultada: 2013-2024, diferenciando tres categorías fundamentales: Deuda Técnica (TD), Deuda de Proceso (DP) y estudios que combinan ambos tipos de deuda.



**Figura 6.** Distribución de los artículos por año de acuerdo al enfoque o temáticas abordadas.  
**Figure 6.** Distribution of articles by year according to the approach or topics addressed.

**La deuda técnica (TD):** Representada por la barra verde, ha mantenido una cuota constante del 8,3% anual entre 2013 y 2020, excepto en 2017 y 2021 (ver [22], [19]). Este patrón sugiere que, aunque la deuda técnica es relevante, la atención en los estudios de deuda de proceso se ha centrado principalmente en su impacto indirecto sobre la deuda técnica. El aumento significativo en 2021 (16,6%) podría estar relacionado con la mayor complejidad de los sistemas actuales.

**Deuda de proceso (DP):** Representada por la barra azul, muestra un aumento a partir de 2019, alcanzando el 30% en 2024. Como se menciona en [1], [2], [3], [28] este crecimiento refleja que las organizaciones han empezado a reconocer la importancia de los procesos en el desarrollo de software y cómo su mala gestión puede generar o amplificar otras formas de deuda, incluida la deuda técnica. **Ánálisis combinado de la deuda técnica y la deuda de procesos (DT y DP):** Representada por la barra naranja, muestra una interconexión del 50% en 2024. Esto indica que los investigadores están reconociendo que ambas deudas no son independientes, sino que están estrechamente relacionadas. Los errores de proceso pueden generar problemas técnicos, y las limitaciones técnicas pueden afectar la eficiencia del proceso, lo que resalta la necesidad de un enfoque holístico para la gestión de la deuda (ver [4], [27]).

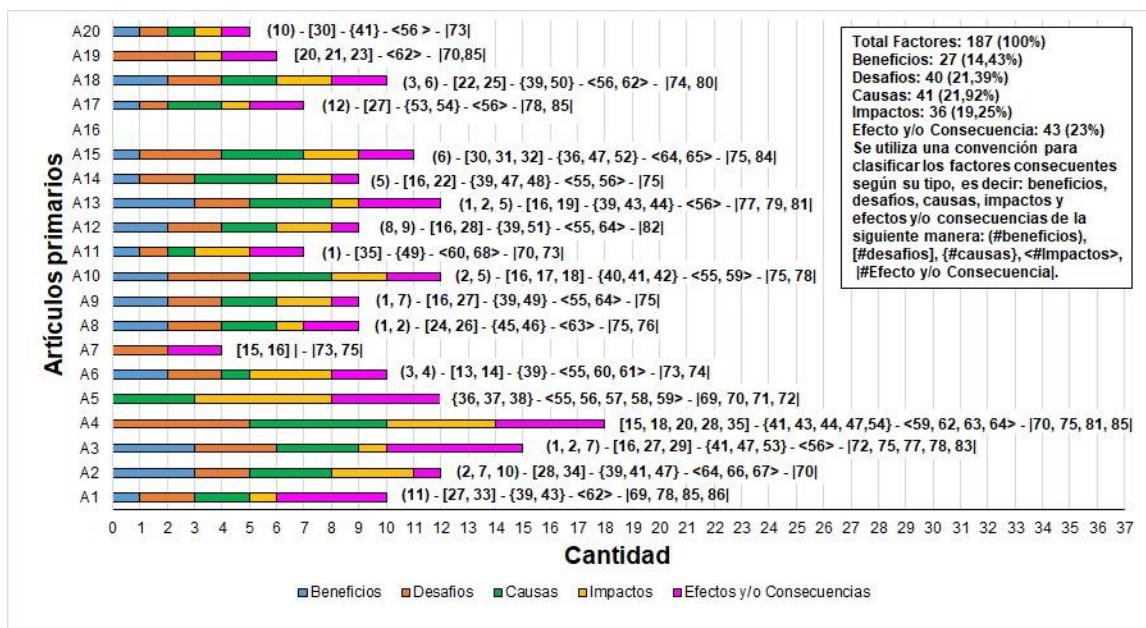
Cabe destacar que los estudios sobre la DP y la DT no solo mencionan esta categoría, sino que también incluyen referencias a otros tipos de deuda, como la deuda de documentación y la deuda social, entre otras.

#### Principales resultados de PI2

La literatura sobre la gestión de la deuda de procesos sigue emergiendo, centrándose cada vez más en su interrelación con la deuda técnica. Si bien se han logrado avances significativos en la comprensión y el tratamiento de la deuda técnica, el campo de la gestión de la deuda de procesos está menos desarrollado. Se necesita más investigación empírica, marcos estructurados y un enfoque holístico para gestionar eficazmente tanto la deuda técnica como la de procesos. Estudios recientes han puesto de relieve la creciente importancia de la deuda de procesos en el desarrollo de software, lo que sugiere un cambio hacia una comprensión más global de la gestión de la deuda.

#### 4.3. ¿Cuáles son los beneficios, desafíos, causas, impactos, efectos y/o consecuencias asociadas con la investigación sobre la deuda de procesos en el desarrollo de software? (PI3)

La deuda de procesos (DP) está vinculada a la deuda técnica (DT) y social, con causas que dificultan la comunicación y colaboración dentro de las organizaciones, como la falta de comunicación efectiva, silos organizacionales y el fenómeno del “lobo solitario”. Además, los cuellos de botella en las interacciones entre líderes y miembros del equipo generan exceso de comunicación y retrasos [14]. Otros fenómenos incluyen sobreingeniería, falta de documentación, prioridades cambiantes y cultura de la culpa [2], entre otros. Los anti-patrones identificados en la literatura, como conocimiento no compartido y falta de motivación [3], están relacionados con la DP y DT. Las consecuencias de la DP incluyen retrasos, mala calidad a corto plazo, y mayores costos y plazos a largo plazo [1]. Para mitigar la DP, se recomienda medir y automatizar procesos, realizar evaluaciones continuas y desarrollar modelos conceptuales. La Tabla 12 ofrece una visión global de los aspectos asociados a la DP en el desarrollo de software, destacando que los retos y causas predominan en la investigación. El artículo [18] destaca la brecha entre habilidades técnicas y blandas, especialmente en entornos ágiles.



**Figura 7. Distribución de factores relacionados con la deuda de procesos en la literatura.**

**Figure 7. Distribution of factors related to process debt in the literature.**

Principales resultados de PI3	
Las consecuencias de la deuda de procesos pueden manifestarse de diversas formas, como retrasos en las entregas, mala calidad de los productos y aumento de los costes. Varios antipatrones, como la falta de comunicación, los silos organizativos y la sobrecarga de trabajo, contribuyen a la acumulación de deuda de procesos. Abordar la deuda de procesos requiere un enfoque proactivo, que incluya la medición y automatización de los procesos, la realización de evaluaciones continuas y la formación de los equipos. Sin embargo, retos como la resistencia al cambio y la dificultad para priorizar las mejoras de los procesos pueden obstaculizar la gestión eficaz de la deuda. Para mitigar los efectos negativos de la deuda de procesos, es esencial adoptar un enfoque más holístico que tenga en cuenta la interacción entre las competencias técnicas y las interpersonales.	

**Tabla 12.** Elementos identificados en la deuda de procesos de software.**Table 12.** Elements identified in the software process debt.

Beneficios			<b>Id</b>	<b>Causa</b>	<b>#Ap</b>	<b>Referencia</b>	
<b>Id</b>	<b>Beneficio</b>	<b>#Ap</b>	<b>Referencia</b>	44	Sincronización deficiente	2	A13, A4
1	Mejora en la calidad del software	6	A8, A13, A9, A3, A2, A11	45	Despliegue apresurado	1	A8
2	Reducción de costos a largo plazo	5	A8, A10, A13, A3, A2	46	Integraciones oportunistas	1	A8
3	Prevención de la deuda técnica	2	A6, A18	47	Ineficiencia en la comunicación	5	A2, A14, A3, A15, A4
4	Identificación de causas	1	A6	48	Falta de capacitación	1	A14
5	Mejora en la productividad	3	A10, A13, A14	49	Decisiones de diseño subóptimas	2	A9, A11
6	Mejora en la comunicación	2	A18, A15	50	Falta de pruebas	1	A18
7	Optimización de procesos	3	A9, A3, A2	51	Gestión inadecuada	1	A12
8	Mejora de la Eficiencia	1	A12	52	Decisiones sociotécnicas inadecuadas	1	A15
9	Satisfacción del Cliente	1	A12	53	Planificación inadecuada	2	A3, A17
10	Organización del conocimiento	1	A20	54	Factores humanos	2	A17, A4
11	Mejora en la priorización de procesos	1	A1	<b>Impactos</b>			
12	Estrategias de gestión	1	A17	<b>Id</b>	<b>Impacto</b>	<b>#Ap</b>	<b>Referencia</b>
<b>Desafíos</b>			55	Aumento de costos	7	A5, A6, A7, A9, A10, A12, A14	
<b>Id</b>	<b>Desafío</b>	<b>#Ap</b>	<b>Referencia</b>	56	Aumento de Tiempo, extensión de los tiempos de entrega.	6	A3, A5, A13, A14, A17, A18
13	Complejidad en la prevención	1	A6	57	Dificultad en la comunicación	1	A5
14	Perspectivas diferentes	1	A6	58	Dificultad en la colaboración	1	A5
15	Equilibrio entre entrega rápida y calidad del software	2	A7, A4	59	Ineficiencia en el desarrollo de software	4	A5, A10, A20, A4
16	Resistencia al cambio	7	A7, A9, A10, A13, A14, A3, A12	60	Dificultad en la planificación	2	A6, A11
17	Planificación inadecuada	1	A10	61	Moral del equipo	1	A6
18	Falta de experiencia del equipo	2	A10, A4	62	Calidad del software	4	A1, A18, A19, A4
19	Falta de conciencia	1	A13	63	Dificultades en el mantenimiento	2	A8, A4
20	Falta de revisión adecuada	2	A19, A4	64	Disminución de la calidad del software	5	A9, A12, A2, A15, A4
21	Revisões dormidas	1	A19	65	Disminución en la productividad del equipo	1	A15
22	No identificar la Deuda de procesos	2	A14, A18	66	Limitaciones en la aplicación de prácticas modernas	1	A2
23	Grandes conjuntos de cambios	1	A19	67	Aumento del esfuerzo operativo	1	A2
24	Falta de Documentación	1	A8	68	Riesgos mal gestionados	1	A11
25	Cambio de Mentalidad	1	A18	<b>Efectos y/o Consecuencias</b>			
26	Decisiones Impulsivas	1	A8	<b>Id</b>	<b>Efecto/consecuencia</b>	<b>#Ap</b>	<b>Referencia</b>
27	Dificultad de identificación y medición	4	A9, A3, A1, A17	69	Dificultad en la planificación y priorización de tareas	2	A1, A5
28	Identificación de la Deuda	5	A12, A2, A4	70	Retrabajo	5	A2, A5, A11, A19, A4
29	Falta de cultura del software	1	A3	71	Confusión en el personal	1	A5
30	Falta de Comunicación	2	A15, A20	72	Moral negativa del equipo	2	A5, A3
31	Procesos Ineficientes	1	A15	73	Acumulación de Deuda	4	A6, A7, A11, A20
32	Aislamiento en el Trabajo	1	A15	74	Gestión inadecuada	2	A6, A18
33	Falta de prácticas de priorización	1	A1	75	Baja calidad del código	8	A7, A8, A9, A10, A14, A3, A15, A4
34	integración efectiva de estrategias de mitigación	1	A2	76	Riesgos en la sostenibilidad del sistema	1	A8
35	Dificultad en la gestión	2	A11, A4	77	Aumento de errores en el código	2	A13, A3
<b>Causas</b>			78	Retrasos en el desarrollo	4	A1, A10, A3, A17	
<b>Id</b>	<b>Causa</b>	<b>#Ap</b>	<b>Referencia</b>	79	Vulnerabilidades en el código	1	A13
36	Procesos defectuosos	2	A5, A15	80	Bajo rendimiento del sistema	2	A13, A18
37	No seguir el proceso definido	1	A5	81	Baja escalabilidad del sistema	2	A13, A4
38	Fallas de infraestructura	1	A5	82	Aumento de la deuda técnica	1	A12
39	Falta de documentación	8	A1, A2, A6, A13, A14, A9, A18, A12	83	Aumento de la carga de trabajo	1	A3
40	Requisitos mal definidos	1	A10	84	Aumento de la complejidad en la gestión de proyecto	1	A15
41	Presión por entrega rápida	5	A2, A10, A3, A20, A4	85	Insatisfacción de las partes interesadas y del equipo de desarrollo	4	A1, A17, A19, A4
42	Prácticas de desarrollo inadecuadas	1	A10	86	Deterioro de la calidad del producto final	1	A1
43	Desalineación de roles y responsabilidades	3	A1, A13, A4				

## 5. DISCUSIÓN

### 5.1. Consideraciones principales

La RSL sobre la DP en el desarrollo de software destaca varios retos y beneficios en la implementación de enfoques ágiles para gestionarla. Uno de los principales desafíos es la complejidad técnica, relacionada estrechamente con la deuda técnica, que se presenta al integrar nuevas herramientas y técnicas en infraestructuras rígidas y poco flexibles. A pesar de estos retos, los beneficios incluyen la optimización de procesos, mejora en la calidad del software y mayor sostenibilidad a largo plazo. La adopción de prácticas ágiles y herramientas automatizadas se propone como solución, aunque debe adaptarse al contexto organizacional para ser eficaz. Los resultados evidencian el cumplimiento de los objetivos: se categorizaron los estudios (OB1), se analizó la diversidad y madurez de enfoques (OB2) y se recopilaron soluciones y desafíos (OB3). Sin embargo, muchas propuestas carecen de validación empírica, lo que resalta la necesidad de estudios de caso y encuestas que fortalezcan su contraste con la práctica. Asimismo, la ausencia de literatura en 2017 podría deberse a un enfoque hacia la deuda técnica. La literatura sugiere que la adecuada gestión de la DP puede ser un diferenciador clave para las empresas de software, recomendando la creación de métricas estandarizadas y la mejora continua como estrategias para prevenir la acumulación de DP en el futuro.

### 5.2. Limitaciones de la revisión

Para acceder al contenido completo puede visitar: <https://shorturl.at/H2C0w>.

### 5.3. Relevancia para la investigación y la práctica

La relevancia de la RSL radica en su valiosa contribución para comprender el estado actual de la gestión de la deuda de procesos en el desarrollo de software. Los resultados obtenidos ofrecen implicaciones significativas tanto para la investigación como para la práctica. En la práctica, estos resultados pueden guiar a los profesionales hacia la adopción de mejores prácticas y herramientas para gestionar eficazmente la deuda de procesos, impulsando así el desarrollo de estrategias más eficaces para mitigar los impactos negativos asociados a esta deuda.

### 5.4. Conclusiones y trabajo futuro

La RSL destaca diversas oportunidades para investigaciones futuras relacionadas con la deuda de procesos. (i) se propone analizar cómo la moral y diversidad del equipo (género, etnia, habilidades, etc.) afectan la calidad del software y la acumulación de deuda de procesos, explorando su impacto en la dinámica, creatividad y resolución de problemas, así como en la eficiencia y el producto final; (ii) es esencial desarrollar herramientas y modelos que automatizan procesos, identifiquen y mitiguen la deuda en los requisitos, y evalúen la incertidumbre en los artefactos de calidad; (iii) se requiere validar y ampliar los marcos conceptuales actuales mediante estudios empíricos, ajustando las prácticas de gestión a distintos contextos organizativos; (iv) se destaca la importancia de fomentar la colaboración entre equipos y disciplinas, diseñando estrategias coordinadas y analizando cómo las interacciones entre diferentes tipos de deuda afectan su acumulación y el desempeño organizacional; (v) se enfatiza la relevancia de continuar investigando este tema emergente para optimizar los procesos en el desarrollo de software. Finalmente, consideramos crucial que los equipos de desarrollo reconozcan la importancia de gestionar la deuda de procesos, ya que esto puede ahorrar tiempo, esfuerzo y recursos, y una buena gestión de la deuda de procesos puede fortalecer las relaciones entre los miembros del equipo y aumentar las probabilidades de éxito del proyecto. Se recomienda además alinear las prácticas industriales con los modelos teóricos, apoyándose en marcos como así como: CMMI, ISO/IEC 33014, LeSS, SAFe, Nexus, entre otros, que faciliten la transferencia del conocimiento académico a la práctica.

## 6. AGRADECIMIENTOS

El profesor César Pardo agradece a la Universidad del Cauca, donde actualmente se desempeña como profesor titular.

## 7. REFERENCIAS

- [1] Martini, A., Besker, T., & Bosch, J. (2020). Process Debt: A First Exploration. *2020 27th Asia-Pacific Software Engineering Conference (APSEC)*, 316–325. <https://doi.org/10.1109/APSEC51365.2020.00040>
- [2] Martini, Antonio and Stray, Viktoria and Besker, Terese and Brede Moe, Nils and Bosch, Jan, Process Debt: Definition, Risks and Management. 2024 Available at SSRN: <https://ssrn.com/abstract=4328073> or <http://dx.doi.org/10.2139/ssrn.4328073>
- [3] Ahmad, M. O., & Gustavsson, T. (2022). The Pandora's box of social, process, and people debts in software engineering. *Journal of Software: Evolution and Process*, 36. <https://doi.org/10.1002/smrv.2516>
- [4] Saeeda, H., Ahmad, M. O., & Gustavsson, T. (2024). A Multivocal Literature Review on Non-Technical Debt in Software Development: An Insight into Process, Social, People, Organizational, and Culture Debt. *E-Informatica Software Engineering Journal*, 18(1), 240101. <https://doi.org/10.37190/e-Inf240101>
- [5] Kitchenham and S. Charters, "Guidelines for performing Systematic Literature Reviews in Software Engineering," 2007. Accessed: May 28, 2022. [Online]. Available: <https://www.researchgate.net/publication/302924724>.
- [6] Wieringa, R., Maiden, N., Mead, N., & Rolland, C. (2006). Requirements engineering paper classification and evaluation criteria: a proposal and a discussion. *Requirements Engineering*, 11(1), 102–107. <https://doi.org/10.1007/s00766-005-0021-6>
- [7] Goldschmidt, G., & Matthews, B. (2022). Formulating design research questions: A framework. *Design Studies*, 78, 101062. <https://doi.org/https://doi.org/10.1016/j.destud.2021.101062>
- [8] Kitchenham, B., Pearl Brereton, O., Budgen, D., Turner, M., Bailey, J., & Linkman, S. (2009). Systematic literature reviews in software engineering – A systematic literature review. *Information and Software Technology*, 51(1), 7–15. <https://doi.org/10.1016/j.infsof.2008.09.009>
- [9] Dybå, T., & Dingsøyr, T. (2008). Strength of evidence in systematic reviews in software engineering. *Proceedings of the Second ACM-IEEE International Symposium on Empirical Software Engineering and Measurement*, 178–187. <https://doi.org/10.1145/1414004.1414034>
- [10] Yang, L., Zhang, H., Shen, H., Huang, X., Zhou, X., Rong, G., & Shao, D. (2021). Quality Assessment in Systematic Literature Reviews: A Software Engineering Perspective. *Information and Software Technology*, 130, 106397. <https://doi.org/https://doi.org/10.1016/j.infsof.2020.106397>
- [11] Castillo, E., & Vásquez, M. L. (2003). El rigor metodológico en la investigación cualitativa. *Colombia Médica*, 34(3), 164–167.
- [12] Harzing, A.-W. (2010). *The publish or perish book*. Tarma Software Research Pty Limited Melbourne, Australia.
- [13] Wohlin, C. (2014). Guidelines for snowballing in systematic literature studies and a replication in software engineering. *Proceedings of the 18th International Conference on Evaluation and Assessment in Software Engineering*, 1–10.
- [14] Freire, S., Rios, N., Mendonça, M., Falessi, D., Seaman, C., Izurieta, C., & Spínola, R. O. (2020). Actions and impediments for technical debt prevention: results from a global family of industrial surveys. *Proceedings of the 35th Annual ACM Symposium on Applied Computing*, 1548–1555. <https://doi.org/10.1145/3341105.3373912>
- [15] Rios, N., Oliveira Spinola, R., de Mendonça Neto, M. G., & Seaman, C. (2018). A Study of Factors that Lead Development Teams to Incur Technical Debt in Software Projects. *2018 44th*

- Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*, 429–436. <https://doi.org/10.1109/SEAA.2018.00076>
- [16] Saeeda, H., Ahmad, M. O., & Gustavsson, T. (2024). Exploring Process Debt in Large-Scale Agile Software Development For Secure Telecom Solutions. *Proceedings of the 7th ACM/IEEE International Conference on Technical Debt*, 11–20. <https://doi.org/10.1145/3644384.3644470>
- [17] Saeeda, H., Ahmad, M. O., & Gustavsson, T. (2023). Multivocal Literature Review on Non-Technical Debt in Software Development : An Exploratory Study. *Proceedings of the International Conference on Evaluation of Novel Approaches to Software Engineering, ENASE* : , 2023-April, 89–101. <https://doi.org/10.5220/0011772300003464>
- [18] García, G. D., Pardo Calvache, C. J., & Rodríguez, F. J. Á. (2022). Society 5.0 and Soft Skills in Agile Global Software Development. *IEEE Revista Iberoamericana de Tecnologias Del Aprendizaje*, 17(2), 197–207. <https://doi.org/10.1109/RITA.2022.3166966>
- [19] Berenguer, C., Borges, A., Freire, S., Rios, N., Tausan, N., Ramac, R., Pérez, B., Castellanos, C., Correal, D., Pacheco, A., López, G., Falessi, D., Seaman, C., Mandic, V., Izurieta, C., & Spínola, R. (2021). Technical Debt is not Only about Code and We Need to be Aware about It. *Proceedings of the XX Brazilian Symposium on Software Quality*. <https://doi.org/10.1145/3493244.3493285>
- [20] Dargó, Z. (2019). *Technical Debt Management: Definition of a Technical Debt Reduction Software Engineering Methodology for SMEs*. <https://api.semanticscholar.org/CorpusID:202233439>
- [21] Doğan, E., & Tüzün, E. (2022). Towards a taxonomy of code review smells. *Information and Software Technology*, 142, 106737. <https://doi.org/https://doi.org/10.1016/j.infsof.2021.106737>
- [22] Kruchten, P., Nord, R. L., Ozkaya, I., & Falessi, D. (2013). Technical debt: towards a crisper definition report on the 4th international workshop on managing technical debt. *SIGSOFT Softw. Eng. Notes*, 38(5), 51–54. <https://doi.org/10.1145/2507288.2507326>
- [23] Alves, N. S. R., Ribeiro, L. F., Caires, V., Mendes, T. S., & Spínola, R. O. (2014). Towards an Ontology of Terms on Technical Debt. *2014 Sixth International Workshop on Managing Technical Debt*, 1–7. <https://doi.org/10.1109/MTD.2014.9>
- [24] Martini, A., Stray, V., & Moe, N. (2019). *Technical-, Social- and Process Debt in Large-Scale Agile: An Exploratory Case-Study* (pp. 112–119). [https://doi.org/10.1007/978-3-030-30126-2\\_14](https://doi.org/10.1007/978-3-030-30126-2_14)
- [25] Poliakov, D. (2015). *A systematic mapping study on technical debt definition*. <https://api.semanticscholar.org/CorpusID:15260409>
- [26] Rosser, L. A., & Norton, J. H. (2021). A Systems Perspective on Technical Debt. *2021 IEEE Aerospace Conference (50100)*, 1–10. <https://doi.org/10.1109/AERO50100.2021.9438359>
- [27] Zabardast, E., Gonzalez-Huerta, J., Gorschek, T., Šmite, D., Alégroth, E., & Fagerholm, F. (2023). A taxonomy of assets for the development of software-intensive products and services. *Journal of Systems and Software*, 202, 111701. <https://doi.org/https://doi.org/10.1016/j.jss.2023.111701>
- [28] Ahmad, M. O., & Al-Baik, O. (2024). Beyond Technical Debt Unravelling Organisational Debt Concept. *Proceedings of the 39th ACM/SIGAPP Symposium on Applied Computing*, 802–809. <https://doi.org/10.1145/3605098.3635913>
- [29] Alves, N. S. R., Mendes, T. S., de Mendonça, M. G., Spínola, R. O., Shull, F., & Seaman, C. (2016). Identification and management of technical debt: A systematic mapping study. *Information and Software Technology*, 70, 100–121. <https://doi.org/https://doi.org/10.1016/j.infsof.2015.10.008>
- [30] Lahti, J. R., Tuovinen, A.-P., Mikkonen, T., & Capilla, R. (2022). ScrumBut as an Indicator of Process Debt. *2022 48th Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*, 318–321. <https://doi.org/10.1109/SEAA56994.2022.00057>