






Olores comunitarios en comunidades de desarrollo de software: Revisión Sistemática de la Literatura

Community smells in software development communities: systematic literature review

Eydy del Carmen Suárez Brievas²   Jorge Enrique Bejarano Guarí¹  Nicole Daniela Urbano Palechor¹  César Jesús Pardo Calvache¹ 

¹ GTI Research Group, Universidad del Cauca, Popayán, Colombia.

² GISICO Research Group, Universidad Popular del Cesar, Valledupar, Colombia.

Resumen

Objetivo: Este estudio presenta los hallazgos iniciales de una revisión sistemática de la literatura, cuyo objetivo fue identificar posibles antipatrones, conocidos como olores comunitarios, que, si no se abordan eficazmente, pueden generar problemas sociales subyacentes en los equipos de desarrollo de software.

Metodología: El estudio se realizó mediante una revisión sistemática de la literatura que proporcionó un análisis detallado y estructurado de la investigación relacionada con los olores comunitarios en equipos de desarrollo de software. El artículo se centra en tres aspectos principales: (i) tendencias demográficas en el tema, (ii) tipos de investigación y (iii) definiciones de olores comunitarios proporcionadas por diferentes autores. La revisión siguió el protocolo propuesto por Kitchenham, lo que permitió la identificación, el análisis y la evaluación rigurosa de la literatura disponible.

Resultados: Se seleccionaron, analizaron y evaluaron un total de 44 estudios primarios, lo que permitió la identificación de 35 olores comunitarios distintos. Entre los más destacados se encuentran el silo organizacional, la nube negra, el lobo solitario y el cuello de botella, junto con sus respectivas causas y efectos adversos.

Conclusiones: La revisión sistemática de la literatura permitió identificar el estado actual de la investigación sobre olores comunitarios y los principales antipatrones asociados a su aparición. Estos problemas pueden dificultar la interacción y el trabajo en equipo entre los miembros, lo que resulta en fallos, retrasos, software incompleto o defectuoso, y la aparición de riesgos psicosociales.

Palabras clave: Antipatrones; Comunicación; Cooperación; Coordinación; Comunidades de software; Deuda social; Olores comunitarios..

Abstract

Objective: This study presents the initial findings of a Systematic Literature Review aimed at identifying potential anti-patterns, known as community smells, which, if not effectively addressed, can generate underlying social issues within software development teams.

Methodology: The study was conducted through a Systematic Literature Review that provided a detailed and structured analysis of research related to community smells in software development teams. The article focuses on three main aspects: (i) demographic trends within the topic, (ii) types of research, and (iii) definitions of community smells as provided by different authors. The review followed the protocol proposed by Kitchenham, enabling the identification, analysis, and rigorous evaluation of the available literature.

Results: A total of 44 primary studies were selected, analyzed, and evaluated, leading to the identification of 35 distinct community smells. Among the most prominent are organizational silo, black cloud, lone wolf, and bottleneck, along with their respective causes and adverse effects.

Conclusions: The Systematic Literature Review allowed for the identification of the current state of research on community smells and the main anti-patterns associated with their emergence. These issues can hinder interaction and teamwork among members, resulting in failures, delays, incomplete or defective software artifacts, and the emergence of psychosocial risks.

Keywords: Anti-patterns; Communication; Cooperation; Coordination; Software communities; Social debt; Community smells.

¿Cómo citar?

Suárez EC, Bejarano JE, Urbano ND, Pardo CJ. Olores comunitarios en comunidades de desarrollo de software: Revisión Sistemática de la Literatura. Ingeniería y Competitividad, 2025, 27;(2):e-30414826

<https://doi.org/10.25100/iyv.v27i2.14826>

Recibido: 27-03-25

Evaluado: 5-05-25

Aceptado: 24-02-25

Online: 18-06-25

Correspondence

eydysuarez@unicesar.edu.co



Contribución a la literatura

¿Por qué se realizó?

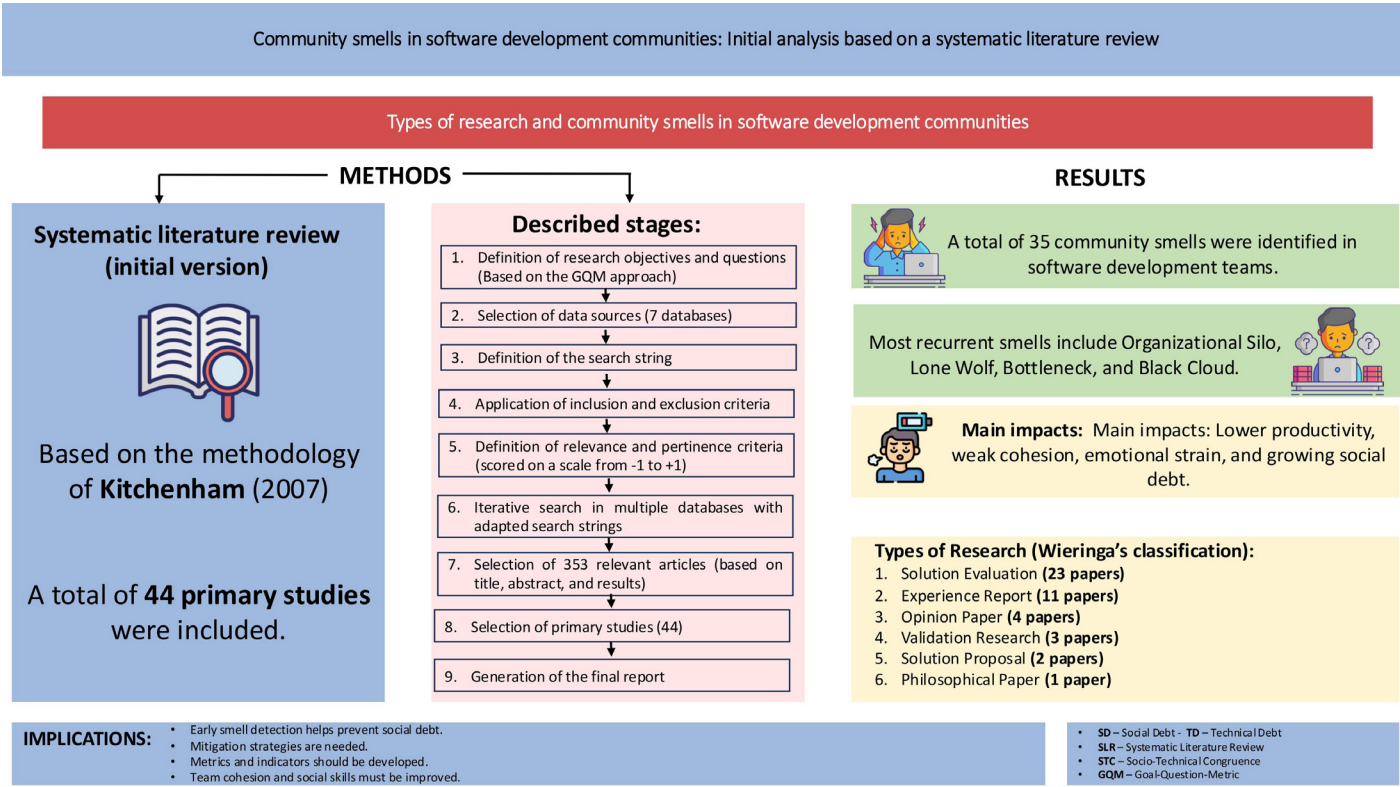
El desarrollo de software va más allá de los aspectos técnicos; es una actividad sociotécnica que exige un equilibrio entre habilidades sociales y técnicas. En este contexto, el factor humano es fundamental para el éxito del proyecto. La principal motivación de este artículo es identificar y analizar los factores que afectan negativamente el bienestar emocional y social de los miembros de las comunidades de desarrollo de software, centrándose en el fenómeno conocido como “olores comunitarios”. Además, este estudio busca evaluar la importancia que se otorga a estos temas dentro de la industria, identificar las investigaciones más significativas en el campo y mapear los países que han liderado el estudio de este tema.

¿Cuáles fueron los resultados más relevantes?

Los hallazgos más significativos del estudio revelan la existencia de 35 olores comunitarios en el contexto del desarrollo de software. La investigación demuestra que la gestión inadecuada de estos olores impacta negativamente no solo a las personas y los equipos, sino también al rendimiento organizacional y a la calidad técnica de los productos desarrollados. El análisis indica además que si no se abordan eficazmente las causas subyacentes de estos olores, puede producirse una acumulación de deuda social, con efectos adversos en la productividad, la eficiencia y el bienestar general de los miembros del equipo.

¿Qué aportan estos resultados?

Además, el estudio identificó y clasificó las causas generales y específicas, los efectos asociados y las estrategias para gestionar y mitigar estos olores comunitarios. Estos hallazgos proporcionan una sólida base conceptual y práctica para comprender y prevenir los problemas psicosociales en las comunidades de desarrollo de software, contribuyendo así a una mejor gestión de equipos y a una mayor calidad en los procesos de ingeniería de software.



Introducción

El proceso de desarrollo de software comprende varias etapas, tareas y actividades que deben abordarse a lo largo de un proyecto de software, desde su desarrollo inicial hasta su eventual retiro. En consecuencia, el trabajo en equipo es esencial, ya que se requiere una comunicación constante para programar y organizar tareas, resolver problemas y monitorear el progreso del proyecto dentro del cronograma planificado (1). También es fundamental que los miembros del equipo apliquen sus conocimientos técnicos para garantizar la entrega de un producto de calidad: software que sea funcional, eficiente, seguro y fácil de mantener. Para ello, es necesario integrar habilidades sociales y técnicas que permitan al equipo adaptarse a los nuevos retos y tecnologías, al tiempo que se promueve la mejora continua dentro de los proyectos (2).

Es importante destacar que el equilibrio entre las habilidades sociales y técnicas contribuye al éxito del proyecto, como se refleja en la calidad del producto final y en el cultivo de un ambiente de trabajo productivo y organizado, especialmente uno basado en la comunicación efectiva (3). Por el contrario, la falta de dicho equilibrio puede dar lugar a (i) desacuerdos o interpretaciones erróneas con respecto a las tareas asignadas, (ii) cambios en los roles después de que el proyecto haya comenzado, y (iii) modificaciones en las actividades sin aclaración o retroalimentación previa. Estos problemas pueden generar desconfianza, estrés, irritación y ruptura de la comunicación, lo que aumenta el riesgo psicosocial (4,5).

Por esta razón, si una organización no invierte en la gestión de estos aspectos humanos, puede surgir un fenómeno conocido como deuda social. La deuda social puede describirse como la acumulación de consecuencias negativas derivadas de la mala comunicación dentro del equipo de trabajo, lo que lleva a interpretaciones erróneas de las ideas, errores, retrasos en los proyectos y colaboración insuficiente (6-10). La deuda social (DS) se produce cuando no se fomenta la comunicación y el apoyo en el entorno laboral, lo que da lugar a que los departamentos, áreas o incluso miembros individuales del equipo trabajen de forma aislada sin relacionarse con otros miembros de la organización. Esto conduce a la incapacidad de compartir eficazmente los recursos, la información y los objetivos. También se manifiesta en individuos que prefieren trabajar solos y aislados, tomando decisiones sin consultar a otros o ejecutando actividades asignadas sin considerar los puntos de vista u opiniones de los demás.

Todos estos factores se reflejan en la productividad del equipo, la resistencia al cambio y a los desafíos, los obstáculos para el flujo y accesibilidad de la información y la falta de estímulo para el aprendizaje mutuo (6-9). Tales condiciones se manifiestan como olores de la comunidad, antipatrones sociotécnicos y una fuente de SD, que afectan a las comunidades de desarrollo de software (SDC), sus procesos, resultados y la organización en general. Los olores de la comunidad pueden entenderse como síntomas o indicadores de situaciones problemáticas o efectos negativos que afectan a los miembros, procesos y/o artefactos del equipo de desarrollo, como documentación poco clara o faltante, código ruidoso, manuales incompletos y uso inadecuado de herramientas. Estos problemas requieren un esfuerzo adicional para comprender o mantener los artefactos afectados, lo que reduce la calidad del software y aumenta los costos de producción.

Además, los olores de la comunidad se han asociado con varios fenómenos negativos, como una alta rotación de empleados y malas decisiones arquitectónicas que a menudo van más allá de los problemas técnicos. Estos olores de la comunidad también pueden conducir a la agitación organizacional o incluso disminuir la estabilidad de los proyectos de software y/o precipitar el fracaso del proyecto (11-16). Además, es pertinente señalar que predecir los olores de la comunidad dentro de una comunidad antes de que se manifiesten durante la línea de tiempo del proyecto de software puede ser crítico para anticipar condiciones organizativas y sociotécnicas subóptimas antes de que se vuelvan inmanejables (15). La falta de atención a los olores comunitarios fomenta la formación de comunidades disfuncionales, lo que produce consecuencias no solo a nivel organizacional sino también en la creación de ambientes laborales desfavorables. Dichas condiciones pueden afectar negativamente la salud mental del personal, desencadenando emociones negativas como irritabilidad, miedo, desánimo, agotamiento y disminución de la motivación. También contribuyen al aislamiento social y a la disminución de la competencia emocional, factores que perjudican la cooperación y el rendimiento del equipo, reduciendo en última instancia la productividad y la eficiencia. En conjunto, estos efectos tienen un impacto perjudicial en el bienestar general y la calidad de vida de los empleados (12,17-21).

Es importante destacar que esta situación puede resultar en el establecimiento de un ambiente laboral desfavorable y mal gestionado, generando consecuencias como: (i) un aumento de los problemas de salud mental, incluyendo depresión, ansiedad y burnout, que atentan contra el bienestar del equipo; (ii) condiciones físicas como hipertensión, enfermedades cardiovasculares y trastornos musculoesqueléticos, como lo demuestra el aumento de los costos médicos y el ausentismo; (iii) bajo desempeño laboral, reflejado en un aumento de errores y menor productividad y calidad en la ejecución de tareas; (iv) insatisfacción laboral, como lo indica el bajo compromiso y la alta rotación; (v) disminución de la innovación y la creatividad debido al estrés y la desmotivación, lo que lleva al estancamiento entre los miembros del equipo y en el desarrollo del proyecto; y (vi) intensificación de los conflictos interpersonales, que se manifiestan como una mala comunicación emocional y profesional entre los miembros del equipo, lo que da lugar a desacuerdos (5,22-24). Dado lo anterior y con el fin de proporcionar una evaluación estructurada y crítica de la deuda social y los problemas de olfato de la comunidad en los equipos de desarrollo de software, se realizó un análisis de la literatura. Los resultados identificaron 35 olores comunitarios, los cuales se detallan en el anexo, Tabla 10.

La Revisión Sistemática de la Literatura (SLR) preliminar está organizada de la siguiente manera: en la sección II se presenta un análisis de los estudios relacionados con el tema; En la sección III se describe la metodología de investigación utilizada para el estudio; En la sección IV se detallan los resultados de las preguntas de investigación; En la sección V se presentan las conclusiones y la labor futura; y por último, en la sección VI se presentan las referencias bibliográficas.

Revisión de la literatura

El SLR proporciona una síntesis de estudios relevantes sobre los temas de SD y olores comunitarios, incluida la metodología, los resultados, el trabajo futuro y las conclusiones. A continuación se presentan algunas de las investigaciones identificadas que enriquecen y aportan significativamente al tema y a su fundamentación conceptual.

En un estudio de Espinosa et al. (17), los autores indican que la SD surge cuando se toman atajos en el desarrollo de software, lo que puede generar problemas a largo plazo. Estos atajos, aunque se adoptan para ahorrar tiempo o dinero, en última instancia pueden resultar en costos futuros más altos. En este contexto, el DS describe las consecuencias adversas de las políticas y procedimientos que afectan a los individuos y las comunidades; estos impactos pueden ir desde la pérdida de ingresos hasta la alteración del equilibrio mental de los profesionales dentro de los COSUDE. Es probable que las comunidades afectadas por el SD generen costos adicionales del proyecto y se las conoce como comunidades “subóptimas” (7). Por otro lado, Tamburri et al. (25) definen el DS como la ausencia de acciones, condiciones o servicios esenciales que afectan el bienestar de los miembros de la comunidad. De manera similar, varios autores (26-30) definen la SD como el costo adicional imprevisto en el que se puede incurrir en un proyecto de software. Estos costos adicionales se producen debido a prácticas deficientes en comunicación, cooperación, coordinación y colaboración entre los miembros de una COSUDE, o debido a políticas estrictas, procesos obsoletos o herramientas establecidas por la organización que no promueven el desarrollo de habilidades blandas entre los miembros; estas condiciones generan causas y/o efectos negativos en la comunidad. Dreesen y cols. (29) también indican que el DS se refiere a las consecuencias futuras de las decisiones relacionadas con las personas y sus interrelaciones. Además, un estudio realizado por (31) subraya que la ingeniería de software es, por naturaleza, una actividad social que involucra a desarrolladores, gerentes y partes interesadas de diferentes partes del mundo. Esto trae varios desafíos, especialmente en términos de colaboración y comunicación, como conflictos de personalidad y barreras lingüísticas y culturales, como se indica en (6-9).

Adicionalmente, es relevante destacar lo mencionado en (32), donde se identifica que, a causa del DS, pueden presentarse condiciones de trabajo inadecuadas y deficientes, manifestándose como factores de riesgo psicosocial. Estos factores tienen el potencial de impactar negativamente la salud emocional, social, cognitiva y física de los trabajadores, generando estrés, frustración, sentimientos de derrota, trastornos del sueño, sobrecarga laboral y, por lo tanto, la necesidad u obligación de trabajar horas extras (5,23,24) para alcanzar los objetivos definidos en el proyecto de software. Se ha demostrado que el STC puede ser utilizado como un indicador temprano para la identificación de SD y, en consecuencia, de olores comunitarios. La adecuada alineación entre las dependencias técnicas y las interacciones sociales, particularmente en los procesos de comunicación, coordinación y cooperación, ayuda a mitigar los efectos derivados de decisiones subóptimas que comprometen la dinámica organizacional. Estas decisiones pueden generar tensiones interpersonales, pérdida de cohesión y acumulación de MS (10,14,28,33).

Otro estudio, realizado por Almarimi et al. (34), explica el término comunidad como condiciones sociales y organizativas subóptimas dentro de una comunidad en desarrollo. Es probable que estas condiciones conduzcan a sobrecostos en un proyecto de software y, posiblemente, a la aparición de SD, lo que afectará directamente a los profesionales de SDC y conducirá a la aparición de nuevos olores comunitarios.

En otro estudio, Palomba et al. (35) afirman que los olores comunitarios representan condiciones subóptimas que surgen en la SDC, como la falta de comunicación entre los subequipos, lo que puede conducir a la presencia de SD y aumentar los costos generales del proyecto. Para anticipar la manifestación de estos olores comunitarios, se utilizan métricas asociadas a STC.

Además, el análisis de Catolino et al. (36), define los olores comunitarios como estructuras organizativas subóptimas capaces de generar SD. Estos olores son frecuentes tanto en proyectos de código abierto como de código propietario y son percibidos como perjudiciales por los profesionales; incluso pueden conducir a la introducción de TD en el código fuente. Es importante destacar la contribución de Palomba et al. (37), quienes afirman que la identificación de los olores de la comunidad dentro de una COSUDE antes de su manifestación durante el desarrollo de un proyecto de software implica reconocer los primeros signos de problemas organizativos o sociotécnicos que podrían afectar el éxito del proyecto. Al identificar estas señales, se pueden implementar acciones preventivas para evitar que los obstáculos se vuelvan inmanejables a medida que avanza el proyecto.

Con base en lo anterior, y a partir de la evaluación de los resultados de los estudios primarios identificados en el SLR, se evidenció que existen propuestas relacionadas con el manejo, detección y reducción de olores comunitarios en la COSUDE. Algunos de estos estudios sugieren que para eliminar o reducir los olores comunitarios, se requiere un esfuerzo por parte de los miembros del equipo de trabajo a través de la recomendación de medidas de apoyo individuales para cada miembro, de acuerdo con sus motivaciones, estilos de trabajo y estatus (16). Además, se identificaron varios olores comunitarios dentro de SDC. Los resultados muestran que los olores más frecuentes son el silo organizativo, la nube negra, el lobo solitario y el silencio de radio. A partir de estos hallazgos, se observó que ninguna investigación evalúa la documentación, las causas y/o los efectos negativos que los olores comunitarios pueden generar en una comunidad en desarrollo. Una vez definidos los conceptos de SD y olores comunitarios, a continuación se presentan las SLR más relevantes sobre el tema.

Espinosa y cols. (17) llevaron a cabo un SLR en el que identificaron 25 estudios primarios, basados en búsquedas en bases de datos reconocidas como Scopus, IEEE Xplore, Google Scholar, Science Direct, ACM, Springer y Web of Science. Los hallazgos de esta revisión destacaron los efectos perjudiciales de la SD en el desempeño de los equipos de desarrollo, impactando negativamente en la toma de decisiones colaborativa, la interpretación de la información, la capacidad de reconocer desafíos y la formulación de estrategias de afrontamiento. El estudio también mostró que el SD puede fomentar tensiones interpersonales, disminuir el compromiso individual, generar ambigüedad en la asignación de roles y aumentar la frustración entre los miembros del equipo, siendo estos efectos consecuencia de una mala comunicación y liderazgo. Como parte de los resultados, se identificaron 30 olores comunitarios vinculados a las causas del MS, lo que permitió una comprensión más profunda de su naturaleza e implicaciones organizacionales.

En un enfoque complementario, Saeeda et al. (21) llevó a cabo una revisión sistemática multivariante centrada en investigaciones recientes sobre las ETD. Para ello, analizaron 110 fuentes, tanto científicas como de literatura gris, y seleccionaron 40 estudios primarios utilizando la estrategia PICO. El estudio destaca que muchos fracasos en el desarrollo de software tienen un origen social o humano, lo que pone de relieve una brecha significativa en la investigación de las ETD, a pesar del creciente interés en la TD. La revisión identifica cinco tipos de ETD: de proceso, social, relacionado con las personas, organizacional y cultural, cada uno asociado con componentes del enfoque sociotécnico. Entre las causas más comunes se encuentran la mala comunicación, el

mal liderazgo, los procesos ineficientes y los entornos organizacionales tóxicos. Finalmente, el estudio propone herramientas, estrategias de mitigación y un marco conceptual para facilitar su comprensión y gestión en diferentes contextos organizacionales.

Por otro lado, Raza et al. (38) desarrolló un SLR centrado en el análisis en profundidad del concepto de STC y su pertinencia en el contexto del desarrollo de programas informáticos. La STC se centra en la alineación de tres factores clave: la comunicación, la coordinación y la cooperación, que son esenciales para las interacciones sociales dentro de las comunidades en desarrollo. La revisión consideró 46 estudios primarios publicados entre 2008 y 2019, seleccionados mediante búsquedas en bases de datos como IEEE, ACM, Scopus y Web of Science. El estudio buscó estructurar y sintetizar el conocimiento disponible sobre STC, abarcando sus componentes, técnicas de medición, factores influyentes, su impacto en el rendimiento del equipo y la calidad del producto, así como las consecuencias derivadas de su ausencia. Entre los resultados, se destacan dos enfoques principales para medir la STC: uno basado en matrices y otro en el análisis de redes sociales. Una mayor congruencia se relaciona con mejores resultados, mientras que su falta puede provocar retrasos, errores y fracasos. Además, se detectaron vacíos en la literatura, como la aplicación limitada de estas métricas en fases distintas al desarrollo y la poca atención a los factores contextuales.

Del mismo modo, Tahsin et al. (39) abordó el estudio de los olores de la comunidad en ingeniería de software utilizando una SLR. Estos olores representan patrones de interacción social disfuncionales que, si no se manejan con prontitud, pueden afectar negativamente la comunicación, la colaboración y el rendimiento del equipo. La revisión se basó en búsquedas en cuatro bases de datos científicas: IEEE Xplore, ACM Digital Library, Scopus y Springer Link, que identificaron 44 estudios primarios publicados hasta 2022. El análisis permitió clasificar 14 tipos de olores comunitarios, siendo los más comunes los silos organizacionales, el lobo solitario, el cuello de botella, la nube negra y el eslabón perdido. Las causas más comunes fueron las deficiencias de comunicación, las estructuras organizativas inadecuadas, la falta de confianza entre los miembros del equipo y la ambigüedad en la asignación de roles. En cuanto a los efectos, se destacó la disminución de la productividad colectiva, la aparición de conflictos internos y la acumulación de SD y TD. Para su detección se utilizaron enfoques como métricas sociotécnicas, SNA y técnicas de aprendizaje automático. Sin embargo, existe una falta de herramientas estandarizadas y ampliamente adoptadas en contextos industriales, lo que pone de manifiesto la necesidad de integrar los mecanismos de detección automática con intervenciones organizativas destinadas a fortalecer la cultura, la comunicación y la estructura del equipo, así como a profundizar la relación entre los olores comunitarios y otras formas de ETD.

Finalmente, Raza et al. (40), realizó un análisis cienciométrico para examinar la evolución, la estructura del conocimiento y las tendencias emergentes del concepto de STC en el desarrollo de software. El trabajo se basó en 306 artículos académicos recopilados entre 2000 y 2020 a través de las bases de datos Scopus y Web of Science. El objetivo principal fue mapear el conocimiento existente sobre CTS, identificar sus principales líneas de investigación, autores e instituciones influyentes, y clasificar los temas más relevantes abordados en la literatura. Para ello, se utilizaron cuatro técnicas cienciométricas: análisis de co-palabras, co-autoría, co-citación y agrupamiento de documentos con análisis de línea de tiempo. Como resultado, se identificaron seis ejes temáticos:

estructura de la comunidad, STC, investigación actual, estructura del producto, desarrolladores clave, y las últimas dos décadas, se evidenció su creciente relevancia para abordar temas de coordinación en equipos distribuidos, y la necesidad de avanzar hacia modelos integradores que combinen métricas técnicas y sociales. Esta investigación proporciona una visión integral del estado del arte y establece una base sólida para futuras investigaciones, al tiempo que mitiga la subjetividad de los estudios anteriores al combinar el análisis cuantitativo y la revisión crítica estructurada.

En la tabla 1 a continuación se presenta el análisis comparativo de los estudios relacionados. Los estudios revisados abordan los olores comunitarios desde perspectivas complementarias y con diferentes niveles de profundidad teórica y metodológica. La obra de Espinosa et al. (17) Realiza una revisión sistemática exhaustiva, identificando 30 olores comunitarios y analizando sus propiedades, causas, efectos y estrategias de mitigación. Su enfoque es amplio, formal y estructurado, respaldado por modelos como el Marco de Etapas del Olfato Comunitario y representaciones gráficas (por ejemplo, diagramas de Sankey). En contraste, el presente artículo ofrece una exploración inicial con la identificación de 35 olores comunitarios, pero no incorpora un análisis demográfico, sino que se centra en un tratamiento sistemático y conceptual del fenómeno. Por su parte, el estudio de Saeeda et al. (21) aborda diversas formas de ETD, incluidas las dimensiones sociales, organizacionales, culturales, de proceso y de personas, basándose tanto en la literatura científica como en la gris. Aunque considera el concepto de SD, su tratamiento es más general y difuso, sin centrarse específicamente en los olores comunitarios. A diferencia del presente trabajo, no proporciona una clasificación detallada de los olores de la comunidad ni un análisis específico de sus efectos, sino más bien una visión panorámica de las fuentes y estrategias asociadas con las ETD en su conjunto.

Tabla 1. Análisis comparativo de los estudios primarios

Aspectos	Espinosa et al. (17)	Saeeda et al. (21)	Raza et al. (38)	Tahsin y cols. (39)	Raza et al. (40)	El presente estudio
Año	2024	2024	2022	2023	2021	2024
Periodo	2013-2022	2014-2022	2008-2019	2015-2022	2000-2020	2017-2024
# Estudios primarios	25	40	46	21	306	48
Blanco	Definir el concepto de olores comunitarios, describir sus propiedades e identificar su relación como fuente de SD.	Analice las ETD identificando sus causas, efectos y estrategias de mitigación para reducir su impacto en el desarrollo de software.	Identificar estudios sobre la evolución de la STC y su impacto en el desarrollo de software.	Revisar sistemáticamente los olores de la comunidad en ingeniería de software, su clasificación, causas, efectos y mitigación.	Analizar el desarrollo histórico y conceptual de la STC a través de técnicas cuantitativas y una revisión crítica de los estudios más significativos en el área.	Se llevó a cabo una revisión de la literatura para proporcionar una evaluación estructurada y crítica de la SD y los olores de la comunidad en los equipos de desarrollo de software.
Análisis	Se discuten las propiedades, causas, efectos y enfoques de mitigación de los olores comunitarios.	Identifica las causas más comunes de las enfermedades tropicales desatendidas, entre las que se incluyen la mala comunicación, el liderazgo ineficaz, los procesos inadecuados y las dinámicas organizativas tóxicas.	Se identifican técnicas para medir la CTS, como el enfoque matricial y el análisis de redes sociales, y se destacan sus impactos positivos en la coordinación del equipo, la calidad del software y el rendimiento del desarrollo.	Los olores comunitarios están relacionados con problemas de coordinación, comunicación y colaboración, que pueden generar SD. Además, se mencionan técnicas para su detección, como el análisis de redes sociales y la minería de repositorios.	Identifica seis temas principales de investigación, incluyendo la estructura de la comunidad, la coordinación, el GSD y el OSS. Muestra los autores, países e instituciones más influyentes.	Se identificaron treinta y cinco olores comunitarios, siendo los más frecuentes el silo organizacional, la nube negra, el lobo solitario, el cuello de botella y el silencio de radio.

Preguntas de investigación (RQ)	Se formularon ocho cuestionarios centrados en los olores comunitarios, abordando su definición, causas, efectos y estrategias de mitigación.	Se proponen tres RQ para analizar las causas, los efectos y las estrategias de mitigación de las ETD en el contexto del desarrollo de software.	Se formularon siete Rec.D. centrados en la STC, abordando sus componentes, su evolución, el impacto de su ausencia, las fuentes de datos, las técnicas de medición y los factores que influyen en su implementación.	El estudio formula cinco RQs que guían un análisis cualitativo de la clasificación taxonómica, causas, consecuencias, estrategias de mitigación y técnicas de detección de olores comunitarios.	El artículo no formula RQ explícitos ni estructura su análisis en torno a ellos.	En esta primera revisión, el estudio responde a cuatro de las seis preguntas sobre la marcha, centrándose en la evolución del concepto, su definición y la identificación de los olores comunitarios.
Dominios	Se identifican treinta olores comunitarios originados por decisiones sociotécnicas subóptimas.	Las ETD se clasifican de acuerdo con el marco hexagonal sociotécnico, que abarca a las personas, los procesos, la cultura, la organización y los aspectos sociales.	Aborda el dominio STC explorando su impacto positivo en la coordinación del equipo y la calidad del software.	Este estudio relaciona directamente los olores comunitarios con los dominios de la comunicación, la colaboración, la coordinación, la estructura social y la CTS.	En este artículo no se definen ni clasifican dominios específicos asociados a SD. Se centra en el análisis cuantitativo y temático del concepto de CTS.	Comunicación, coordinación, colaboración, estructura social y congruencia sociotécnica.

Fuente: Elaboración propia

En comparación, el estudio Raza et al. (38) refieren centrarse en la CTS como mecanismo para mejorar la coordinación y prevenir problemas sociales. Sin embargo, no aborda directamente los olores de la comunidad o el SD, y su análisis se limita al impacto de la STC en la dinámica de equipo, lo que marca una diferencia importante con respecto al enfoque del presente artículo. Por otro lado, Tahsin et al. (39) presentan una revisión sistemática rigurosa que vincula explícitamente los olores de la comunidad con el MS, clasificando las causas y efectos desde una perspectiva global. A diferencia del presente estudio, su enfoque es más teórico y general, sin ahondar en los contextos organizacionales ni en las consecuencias psicosociales de los olores comunitarios. Finalmente, Raza et al. (40) ofrecen una revisión cuantitativa y conceptual centrada en la evolución del concepto de CTS. A diferencia del presente trabajo, no identifica dominios ni analiza los olores de la comunidad; su objetivo se orienta a trazar la trayectoria investigativa del concepto, sin considerar sus efectos sociales ni su relación con el DS.

Metodología

El propósito de este artículo es esclarecer el significado de los términos “olores comunitarios” y “deuda social”, así como las causas que los producen, sus características y sus efectos en la CDS. Esto se logró mediante la revisión y clasificación de 44 artículos primarios, los cuales fueron identificados mediante la aplicación de criterios de inclusión (CI) y criterios de exclusión (CE). Para llevar a cabo este estudio, se realizó una SLR basada en el procedimiento sugerido por Kitchenham et al. (41).

Además, se utilizó el enfoque propuesto por Basili y Caldiera (42), Goal-Question-Metric (GQM), para establecer los objetivos y las preguntas de investigación. Este enfoque sugiere tres niveles: (i) nivel conceptual, que establece objetivos para identificar el propósito del SLR, determinando su alcance y beneficios esperados; (ii) nivel operativo, en el que las preguntas se diseñan con base en los objetivos del nivel conceptual, estas preguntas son fundamentales para definir, organizar y describir las principales características de los ítems identificados y seleccionados, asegurando que estén relacionados con el tema principal de investigación; y (iii) nivel cuantitativo, en el que se establecen métricas para cuantificar los resultados, aunque en este estudio no se aplicará este nivel. En la Figura 1 se muestran las acciones llevadas a cabo durante el desarrollo del SLR.

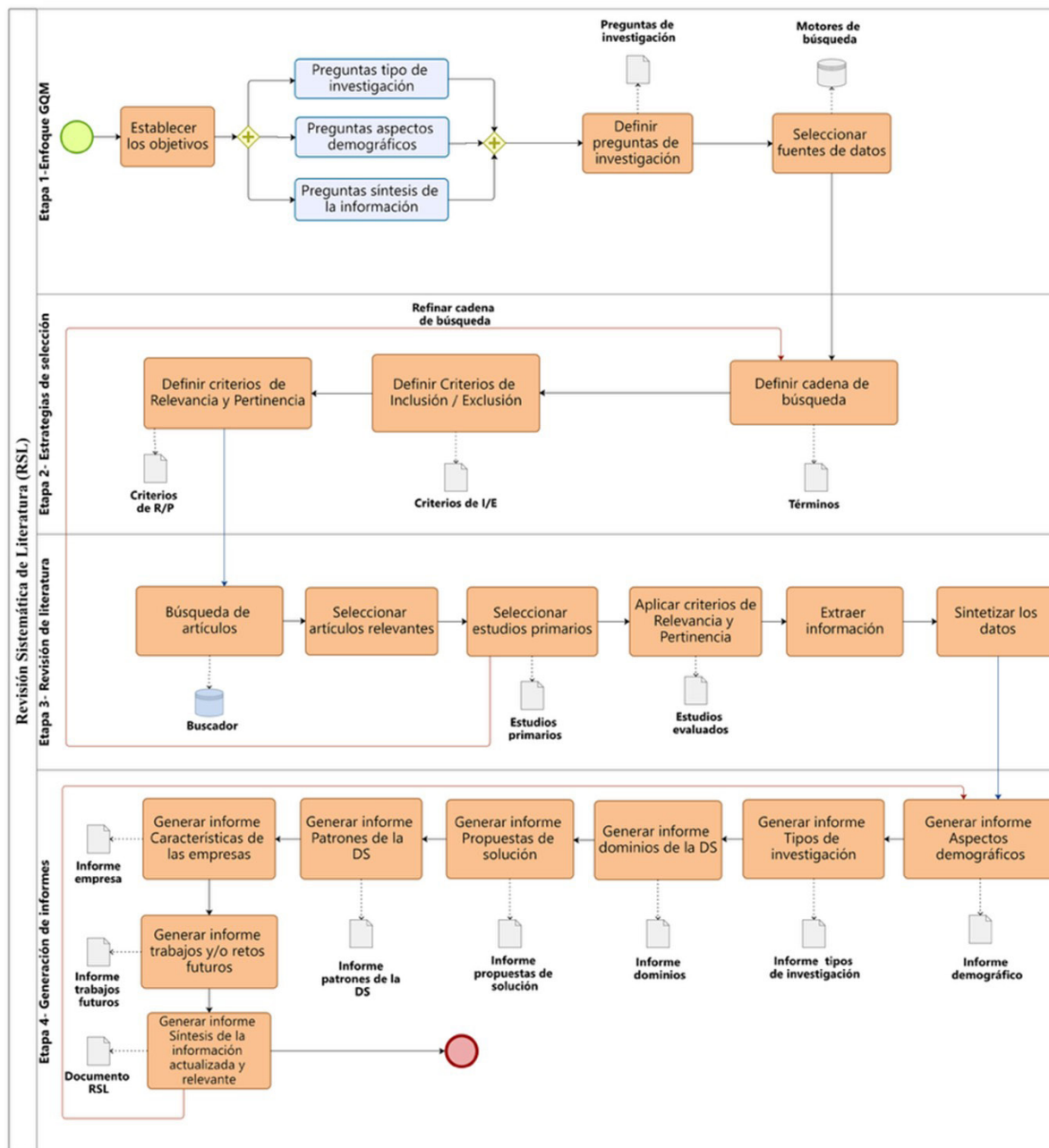


Figura 1. Protocolo RSL (41)

A continuación se ofrece una explicación detallada del desarrollo de cada una de las etapas enumeradas en la (Fig.1), y se proporciona el marco estructurado utilizado para llevar a cabo el SLR.

Etapa 1 – Enfoque GQM: Las actividades desarrolladas en esta etapa se detallan a continuación:

Establecer los objetivos de investigación: Permiten clarificar los aspectos específicos a investigar, facilitando el enfoque en el tema. Para este estudio se establecieron cuatro objetivos de investigación: (i) analizar los estudios sobre SD y olores comunitarios en CSD, (ii) evaluar la calidad y los métodos de investigación para identificar brechas de conocimiento, (iii) sistematizar el estado actual de las propuestas académicas y profesionales en el área, y (iv) clasificar las tendencias y trabajos futuros.

Definir preguntas de investigación: Este SLR abordó cuatro de los once RQ que se formularon originalmente para el estudio, cada uno vinculado a un objetivo de investigación: dos dirigidos a identificar aspectos demográficos (RQ1-RQ2), uno para clasificar el tipo de investigación (RQ3), y uno para sintetizar información actualizada y relevante (RQ4) sobre soluciones relacionadas con la definición, causas, efectos y consecuencias de los olores comunitarios en el CSD. Estas preguntas, enumeradas en la tabla 2, buscan: (a) identificar las conferencias y revistas científicas donde se publicaron los estudios primarios, (b) reconocer a los líderes, países e instituciones prominentes en el estudio de los olores comunitarios y el SD, (c) organizar los tipos de investigación de acuerdo con el esquema de clasificación establecido por Wieringa et al. (43), que abarca categorías como investigación, propuesta de solución, validación, artículos filosóficos, documentos de opinión o artículos de experiencia personal, (d) definir claramente el concepto, los tipos y la evolución de los olores comunitarios, y (e) analizar su acumulación y efectos en el CSD.

Tabla 2. Preguntas de investigación

Pregunta de identificación	Categoría	Pregunta	Motivación
RQ1	Aspectos demográficos	¿Cuál es la distribución temporal de la producción científica sobre los olores comunitarios en la CDS?	Esta pregunta permite identificar tendencias y la evolución del interés académico en el tema, facilitando el análisis de la madurez del campo.
RQ2		¿Cuáles son las revistas y conferencias de mayor impacto que han publicado investigaciones sobre los olores de la comunidad en el contexto del desarrollo de software?	Conocer los canales de publicación más relevantes ayuda a delimitar las fuentes de referencia clave y reconocer la legitimidad académica del tema.

RQ3	Tipo de investigación	¿Cuáles son los tipos de investigación que predominan en los estudios de olores comunitarios en CSD?	Clasificar los diferentes tipos de investigación evidenciados de acuerdo con el esquema de clasificación propuesto por Wieringa et al. (44,45), investigación, validación, propuesta de solución, artículos filosóficos, artículos de opinión o artículos de experiencia personal.
RQ4	Resumen de la información actualizada y relevante	¿Cómo conceptualizan y definen los diferentes autores el término olores comunitarios en el contexto de la CDS?	Definir el concepto de comunidad y sus tipos en el contexto del desarrollo de software permite comprender su evolución, los dominios en los que se produce y su manifestación dentro de las comunidades de desarrollo, facilitando así una mayor claridad en su análisis e interpretación.

Fuente: Elaboración propia

Selección de fuentes de datos: El propósito de esta fase es asegurar la inclusión de información de alta calidad, garantizando que los estudios seleccionados aborden directamente las preguntas de investigación. Por esta razón, se identificaron y seleccionaron las siguientes bases de datos: Google Scholar, ACM, Web of Science, Scopus, ScienceDirect, SpringerLink e IEEE Xplore.

Etapa 2 – Estrategias de selección. Esta etapa incluye:

Definición de la cadena de búsqueda, con el objetivo de recopilar estudios relevantes y evitar sesgos. Para ello, se utilizó una combinación de los operadores lógicos “AND” y “OR”, para controlar la inclusión de términos relevantes en los resultados. La tabla 3 Presenta la cadena de búsqueda básica ejecutada para clasificar los artículos primarios, considerando el período de 2017 a 2024, y ajustando los filtros de acuerdo a cada buscador. La elección de esta ventana de tiempo se basó en el aumento significativo de las publicaciones sobre SD a partir de 2017. Es importante destacar que el primer artículo sobre este tema se publicó en 2013, titulado “¿Qué es la deuda social en ingeniería de software?” (46) , Este artículo se centró en el concepto de SD, algunos de sus efectos y los aspectos técnicos afectados por su acumulación. En 2015, se publicaron dos artículos adicionales centrados en el impacto de la SD en las empresas y el desarrollo de un marco llamado DAHLIA, bajo los títulos “Deuda social en ingeniería de software: perspectivas de la industria” (25) y “Cuando la arquitectura de software lleva a la deuda social” (47). Posteriormente, en 2016 se publicaron nuevos resultados y actualizaciones del marco, que se consideraron como artículos relevantes en este trabajo. Por lo tanto, en función de los cambios, publicaciones y actualizaciones, se consideró relevante la ventana de tiempo seleccionada.

Tabla 3. Cadena de búsqueda

Cadena de búsqueda.
("deuda social" O "olor a comunidad") Y ("desarrollo ágil de software" O "enfoque ágil" O "desarrollo ágil").

Fuente: Elaboración propia

Definir criterios de inclusión (IC) y exclusión (EC): Durante la etapa de ejecución, uno de los investigadores definió los criterios necesarios para incluir o excluir estudios (CI y CE), para acotar la búsqueda y asegurar la pertinencia de los resultados. Estos resultados fueron posteriormente cuantificados. A continuación, el segundo investigador revisó estos criterios y proporcionó comentarios para mejorarlos. Gracias a estos lineamientos, fue posible identificar estudios relevantes que abordaron los objetivos y preguntas de investigación. La tabla 4 detalla los criterios aplicados durante el SLR. Su implementación optimizó el proceso de evaluación, permitiendo la selección de 353 estudios considerados relevantes para la investigación. Esta selección se basó en una revisión cuidadosa del título, el resumen y los resultados de cada artículo, priorizando aquellos que abordaban principalmente el SD y los olores de la comunidad.

Tabla 4. Criterios de inclusión y exclusión

Id IC	Criterios de inclusión	Identificación de la CE	Criterios de exclusión
IC1	Estudios cuyo tema principal son los olores de la comunidad o SD en el desarrollo de software.	EC1	Estudios duplicados (considerando el primer estudio identificado).
IC2	Estudios cuya temática esté relacionada con los olores de la comunidad o el SD en el desarrollo de software.	EC2	Estudios donde el tema de investigación se aborda de manera superficial.
IC3	Estudios que traten temas relacionados con los olores comunitarios o el SD.	EC3	Estudios fuera del rango en años comprendidos entre 2017 y 2024.
IC4	Estudios publicados en revistas, congresos o conferencias de alto impacto revisados por pares.	EC4	Estudios de tipo discusión o estudios disponibles solo como presentaciones o resúmenes.

IC5	Autores que más estudios han realizado en este campo.	EC5	Estudios que son libros o capítulos de libros.
-----	---	-----	--

Fuente: Elaboración propia

Definir criterios de pertinencia y pertinencia: Para evaluar la relevancia y pertinencia de los estudios primarios relacionados con los olores comunitarios o el SD en la DCS, se eligió el enfoque de Kitchenham et al. (41), s y se definieron y aplicaron 11 criterios de evaluación utilizando una escala de tres valores (-1, 0, +1): una puntuación de +1 indicó que el estudio abordó explícitamente la pregunta evaluada; 0,5, que lo abordó parcialmente; y 0, que no lo consideró. Esta evaluación permitió clasificar los artículos de acuerdo con su contribución, sin que ello implique su eliminación del análisis. Los criterios aplicados se detallan en el Anexo. Tabla 8.

Etapa 3 – Revisión de la literatura. En esta etapa se consideraron los siguientes pasos:

Búsqueda de artículos: para identificar los artículos relevantes se llevaron a cabo 7 iteraciones, una para cada base de datos: Google Scholar, ACM, Web of Science, Scopus, ScienceDirect, SpringerLink e IEEE Xplore. Este proceso implicó adaptar la cadena de búsqueda básica en cada motor de búsqueda. Se obtuvieron un total de 803 artículos como resultados de búsqueda, presentados en la tabla 5.

Tabla 5. Resultados de la búsqueda

Identificación	Buscar fuente	Artículos encontrados	Artículos relevantes	Artículos primarios
1	Google Académico	349	14	10
2	Biblioteca Digital	31	7	7
3	ACM	183	182	1
4	Web de la Ciencia	129	104	8
5	Scopus	50	26	7
6	Ciencia Directa	48	13	5
7	Enlace Springer	13	7	6
Total	IEEEExplore	803	353	44

Fuente: Elaboración propia

Selección de artículos relevantes: durante esta fase se aplicaron el CI y el CE. El EC incluyó la eliminación de artículos, libros o capítulos de libros duplicados, así como estudios que abordaban superficialmente el SD y los olores de la comunidad. Para ello, se revisó el título, resumen y resultados de cada artículo. Por otro lado, los criterios de inclusión consideraron estudios cuya temática principal fue la SD o los olores de la comunidad en el contexto del desarrollo de software. Como resultado de este proceso, se identificaron 353 artículos relevantes, como se presenta en la Tabla 5.

Selección de artículos primarios: en esta etapa se llevó a cabo la revisión de los artículos relevantes. Durante este proceso, se evaluó la contribución de cada artículo a las preguntas de investigación

formuladas; además, se identificaron nuevas palabras clave, lo que permitió ajustes en la cadena de búsqueda. Luego de una evaluación exhaustiva, se seleccionaron 44 artículos como estudios primarios, los cuales se destacan en la tabla 6.

Tabla 6. Resumen de la lista de artículos primarios resultantes de la búsqueda.

Identificación	Artículo	NC	Año	Ref
PS1	La influencia de la deuda técnica en la moral de los desarrolladores de software	61	2020	(48)
PS2	El uso de incentivos para promover la gestión técnica de la deuda	14	2022	(49)
PS3	Seguimiento técnico de la deuda: Estado actual de la práctica: una encuesta y un estudio de caso múltiple en 15 grandes organizaciones	91	2018	(50)
PS4	Aprender a detectar los olores de la comunidad en proyectos de software de código abierto	41	2020	(51)
PS5	Predicción de la aparición de olores comunitarios mediante métricas sociotécnicas: un enfoque de aprendizaje automático	42	2021	(52)
PS6	Impactos de los patrones de la comunidad de software en el proceso y el producto: un estudio empírico	14	2022	(53)
PS7	Refactorización de los olores de la comunidad en la naturaleza: el manual de campo del profesional	39	2020	(54)
PS8	Sobre la detección de olores comunitarios mediante una cadena clasificadora de conjuntos basada en programación genética	37	2020	(55)
PS9	Diversidad de género y mujeres en equipos de software: ¿Cómo afectan a los olores de la comunidad?	142	2019	(56)
PS10	Empalme de patrones y olores de la comunidad: un estudio preliminar	23	2020	(57)
PS11	Rompiendo una barrera a la vez: cómo las mujeres desarrolladoras se las arreglan en una industria dominada por hombres	24	2021	(58)

PS12	Entendiendo la Variabilidad de los Olores de la Comunidad: Un Enfoque Estadístico	25	2021	(59)
PS13	Explorando los olores de la comunidad en código abierto: un enfoque automatizado	92	2021	(60)
PS14	Arquitectura de Software Deuda Social: Gestión del Factor de Incomunicabilidad	34	2019	(61)
PS15	¿Cómo influyen los olores comunitarios en los olores codificados?	25	2018	(62)
PS16	Predicción de la ocurrencia de olores de la comunidad en desarrolladores individuales por sentimientos	18	2021	(63)
PS17	Diversidad de género y olores comunitarios: perspectivas desde las trincheras	43	2020	(64)
PS18	Más allá de los aspectos técnicos: ¿Cómo influyen los olores comunitarios en la intensidad de los code smells?	132	2021	(65)
PS19	Un estudio empírico sobre el efecto de los olores de la comunidad en la predicción de insectos	14	2021	(66)
PS20	La comunidad de empalme y los olores de la arquitectura de software en los equipos ágiles: un estudio industrial	15	2019	(67)
PS21	Entendiendo la Deuda Social en la Ingeniería de Software	4	2021	(68)
PS22	"El segundo vicio es mentir, el primero es endeudarse". Antecedentes y Prácticas Mitigadoras de la Deuda Social: un Estudio Exploratorio en Equipos de Desarrollo de Software Distribuido	7	2021	(69)
PS23	Entendiendo la Relación entre el Olor de la Comunidad del Eslabón Perdido y los Cambios que Inducen a la Reparación	4	2021	(70)
PS24	Diversidad de género y olores comunitarios: un estudio de doble replicación en equipos de software brasileños	10	2022	(71)

PS25	¿Las buenas cercas hacen buenos vecinos? Sobre el impacto de la dispersión cultural y geográfica en los olores comunitarios	16	2022	(72)
PS26	En busca de la congruencia sociotécnica: un estudio longitudinal a gran escala	23	2021	(73)
PS27	Sobre la relación entre la raza y el origen étnico perceptibles del desarrollador y la evaluación de las contribuciones en OSS	34	2022	(74)
PS28	Incluir a todos, en todas partes: comprender las oportunidades y los desafíos de la inclusión geográfica de género en el OSS	50	2022	(75)
PS29	Un estudio preliminar sobre la asignación de problemas de GitHub a los comentaristas de problemas y la relación con los olores sociales	2	2022	(76)
PS30	Un estudio sobre las correlaciones entre los olores arquitectónicos y los patrones de diseño	12	2021	(77)
PS31	Decisiones de diseño arquitectónico que incurren en deuda técnica: un estudio de caso industrial	19	2021	(78)
PS32	Women in Agile: El impacto del apoyo organizacional para el avance de las mujeres en la calidad y el rendimiento del trabajo en equipo en los equipos de desarrollo de software ágiles	22	2021	(79)
PS33	Comunicación entre equipos en ingeniería de software a gran escala en co-ubicaciones: un estudio de caso	17	2022	(80)
PS34	Deuda técnica, social y de proceso en la metodología ágil a gran escala: un estudio de caso exploratorio	34	2019	(81)
PS35	Descubriendo patrones de comunidad en código abierto: un enfoque sistemático y su evaluación	86	2019	(82)
PS36	Dimensiones de la Consistencia en GSD: Factores Sociales, Estructuras e Interacciones	6	2020	(83)

PS37	Refactorización de recomendaciones basadas en la optimización de la congruencia socio-técnica	4	2020	(84)
PS38	Comprender la participación de los desarrolladores en el olor de la comunidad de eslabones perdidos: un estudio exploratorio sobre proyectos apache	7	2020	(85)
PS39	¿Estamos trabajando bien con los demás? Cómo impactan los sistemas multiequipo en la calidad del software	6	2018	(86)
PS40	Descubriendo la deuda social con el marco CAFFEA: Un antídoto contra la deuda arquitectónica	13	2017	(87)
PS41	El trabajo emocional de los ingenieros de software	27	2017	(88)
PS42	Analítica de deuda social para mejorar la gestión de las tareas de evolución del software	7	2017	(89)
PS43	Seguridad Psicológica, Liderazgo y Deuda No Técnica en el Desarrollo de Software Ágil a Gran Escala	1	2023	(90)
PS44	Localización de olores de la comunidad en los procesos de desarrollo de software utilizando centralidades de red de orden superior	35	2023	(91)

Siglas. PS: Identificador primario del estudio. NC: Número de citas según Google Scholar. Ref: Referencia. PS: Estudios Primarios.

Fuente: Elaboración propia

Aplicar criterios de pertinencia: Una vez definidos los criterios se aplicaron a los estudios primarios sumando los valores individuales, tal y como se establece en el Anexo. tabla 9 y a los estudios enumerados en la Tabla 6. El dos por ciento (PD18) obtuvo la puntuación más alta (10 puntos), mientras que el 7 por ciento obtuvo 0 puntos, lo que refleja contribuciones parciales con limitaciones en la cobertura de los temas. Por otro lado, el 4% (PS6, PS9) registró la puntuación más baja (-2 puntos). Es importante destacar que una puntuación baja no implica una baja calidad del estudio, sino solo el cumplimiento parcial de los criterios aplicados en esta revisión sistemática.

Extracto de información: Este estudio recopiló los elementos más relevantes de cada artículo revisado, incluyendo título, resumen, año y tipo de publicación, base de datos, tipo de investigación, conclusiones, resultados y contribuciones. También se consideraron los objetivos y las preguntas de investigación formuladas.

Sintetizar los datos: en esta fase se integró, resumió y presentó de manera coherente y estructurada la información obtenida de los estudios revisados. A continuación, se elaboró un resumen en el que se destacaron los aspectos clave y las principales conclusiones derivadas de la literatura analizada.

Etapas 4 – Generación de informes. En esta etapa, las preguntas de investigación fueron respondidas a través de la elaboración de informes, los cuales fueron presentados considerando las tres categorías establecidas para las preguntas de investigación: aspectos demográficos, tipos de investigación y síntesis de información actualizada y relevante (ver tabla 2). Para facilitar la lectura de este artículo en apartados posteriores y la identificación de los estudios primarios, junto con el resto de referencias utilizadas, se utilizará el identificador asignado a cada estudio primario (EP), tal y como se presenta en la tabla 6.

Resultados y discusión

Después de completar el procedimiento de búsqueda y analizar los estudios primarios identificados, se proporcionan respuestas a las siguientes preguntas de investigación.

Aspectos demográficos

En esta sección, el objetivo principal es proporcionar respuestas relevantes a las preguntas de investigación RQ1 – RQ2, enumeradas en la tabla 1 y con base al análisis realizado.

PI1. ¿Cuál es la distribución temporal de la producción científica en las comunidades de desarrollo de software?

La clasificación y el análisis de los estudios primarios seleccionados permitieron identificar la frecuencia y el tipo de publicaciones relacionadas con los temas de olores comunitarios y SD. (Fig. 2) muestra que el mayor número de publicaciones se registró en 2021, representando el 32% del total, con 9 artículos publicados en revistas científicas y 5 artículos en congresos en ese año. En 2022 se registró el 18% de las publicaciones, con 5 artículos de revistas y 3 artículos de congresos. Por su parte, 2020 aportó otro 18%, distribuido en 3 ponencias de congresos, 3 artículos de revistas y 2 artículos de taller. En 2019, la producción alcanzó el 11%, con 2 artículos de revista, 2 artículos de conferencias y 1 artículo de taller. Por otro lado, el año 2023 aportó un 7%, distribuido en 2 artículos de revista y 1 artículo de taller. Los años 2017 y 2018 aportaron cada uno el 7% de las publicaciones: en 2017, hubo 3 publicaciones de taller, mientras que en 2018, hubo 2 artículos de revista y 1 artículo de taller. Finalmente, hasta enero de 2024, no se identificó ninguna publicación.

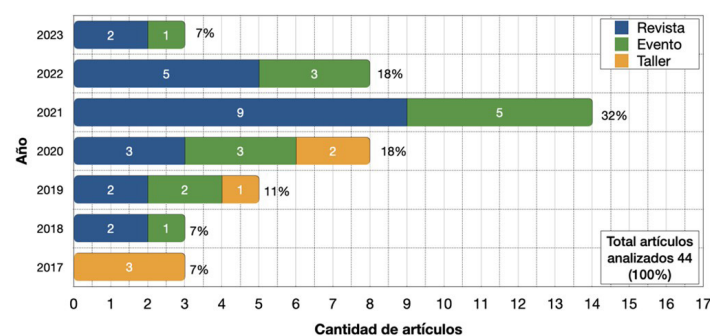


Figura 2. Tipo y número de publicaciones por año

RQ2. ¿Cuáles son las revistas y conferencias de mayor impacto que han publicado investigaciones sobre los olores de la comunidad en el contexto del desarrollo de software?

El notable aumento en el número de publicaciones en 2021 indica que estos temas cobraron mayor relevancia entre los investigadores y la comunidad científica, lo que motivó la difusión de hallazgos, trabajos futuros y/o la identificación de brechas de investigación. Publicar en revistas científicas y participar en eventos académicos son estrategias clave para compartir conocimientos, recibir retroalimentación para mejorar los estudios y aumentar la visibilidad de la investigación tanto dentro de la comunidad científica como entre el público interesado.

Se publicaron un total de 23 artículos en revistas especializadas en Ingeniería de Software, cubriendo una variedad de temas relacionados con esta disciplina, incluyendo el uso de métodos ágiles en equipos distribuidos globalmente, la gestión de la deuda técnica, la inteligencia artificial y el aprendizaje automático en Ingeniería de Software, y la gestión de proyectos de software, así como herramientas, entornos y modelos de desarrollo. El resto de los artículos fueron presentados en eventos científicos en Ingeniería de Software, abordando temas de gran importancia como la responsabilidad ética de los profesionales y los dilemas éticos en la práctica de este campo, la privacidad de datos, la equidad e inclusión, la ciberseguridad, y su impacto en el empleo y la economía. También se discutieron investigaciones y proyectos que utilizan la Ingeniería de Software para abordar desafíos sociales y mejorar la calidad de vida.

Como se muestra en la (Fig. 3), las revistas "Journal of Systems and Software" representan el 30,45% de las publicaciones, lo que equivale a 7 artículos, y "IEEE Transactions on Software Engineering" con el 21,7%, correspondiente a 5 publicaciones, son las revistas con mayor número de artículos científicos publicados sobre SD y community smells. Por otro lado, el evento con mayor número de presentaciones es "Proceedings of the Annual Hawaii International Conference on System Sciences" con un 18,75%, correspondiente a 3 artículos publicados, como se ilustra en la Figura 4. El taller con mayor número de publicaciones es "CEUR Workshop Proceedings", que representa el 50% del total de artículos publicados en los talleres, lo que equivale a 3 artículos publicados, como se muestra en la Figura 5.

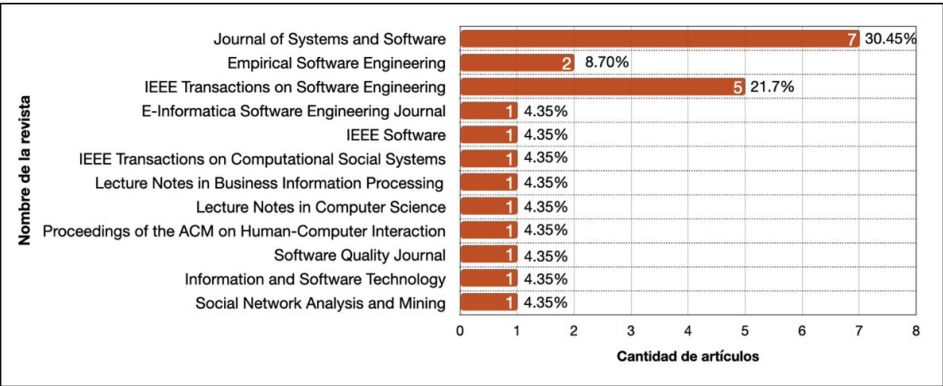


Figura 3. Número de artículos por revista

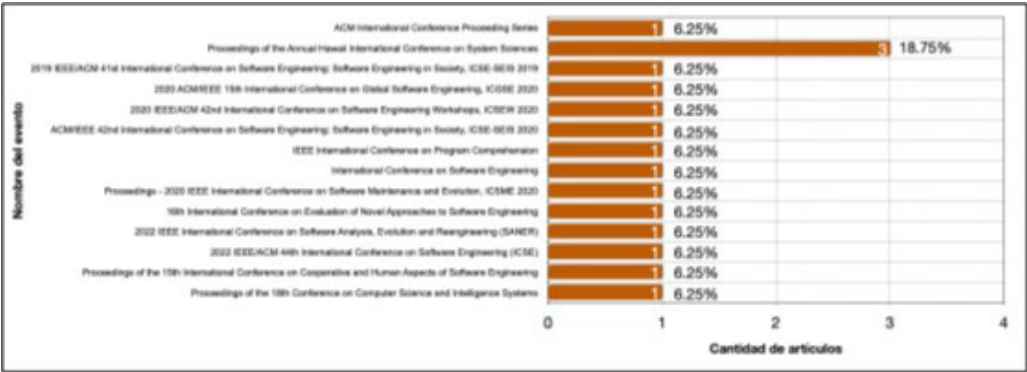


Figura 4. Número de ponencias por evento

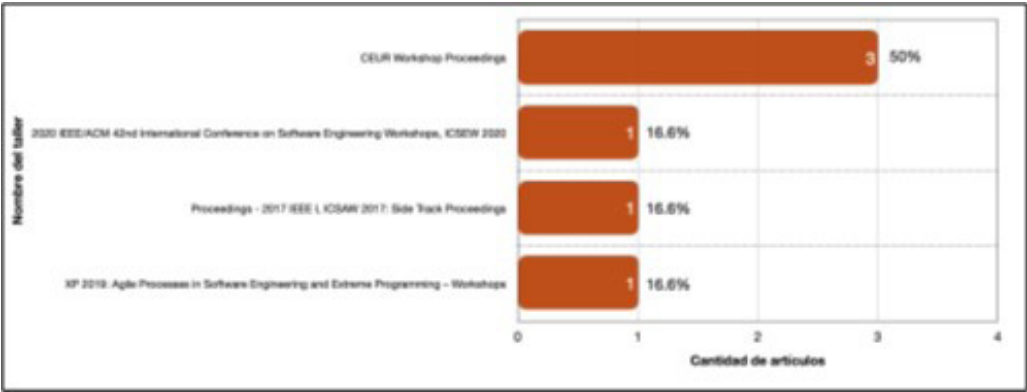


Figura 5. Número de ponencias por taller

Tipos de investigación

Esta sección responde a la siguiente pregunta de investigación:

PI3. ¿Cuáles son los tipos de investigación que predominan en los estudios sobre los olores comunitarios en las comunidades de desarrollo de software?

Para clasificar los 44 estudios primarios identificados, se utilizaron los seis tipos de investigación propuestos por Wieringa et al. (92) se utilizaron como referencia: (i) validación, (ii) evaluación, (iii) propuesta de solución, (iv) artículos filosóficos, (v) artículos de opinión y (vi) artículos de experiencia personal. De acuerdo con esta clasificación y con base en la (Fig. 6), se identificaron 2 estudios (5%) de validación . Esta categoría incluye la investigación que examina las características de una propuesta de solución que aún no se ha implementado. Los temas abordados en estos estudios incluyen diferentes dimensiones de la Ingeniería de Software, como la diversidad de género, los olores comunitarios, la congruencia sociotécnica y la dispersión cultural y geográfica.

En cuanto al tipo de investigación evaluativa, se encontró que el 45% (correspondiente a 20 estudios) pertenecen a esta categoría. Estos trabajos se clasificaron como investigación de evaluación porque generaron nuevos conocimientos a través de metodologías de investigación detalladas y novedosas. Estos artículos incluyen estudios de caso cuyo propósito es recopilar

datos cuantitativos y/o cualitativos sobre los factores y causas que impulsan y mitigan los olores comunitarios en CSD. Asimismo, se identificaron estudios que se apoyan en técnicas de recolección de datos como encuestas, entrevistas y observación para determinar la ocurrencia de olores comunitarios en equipos de trabajo de PS14 a PS16.

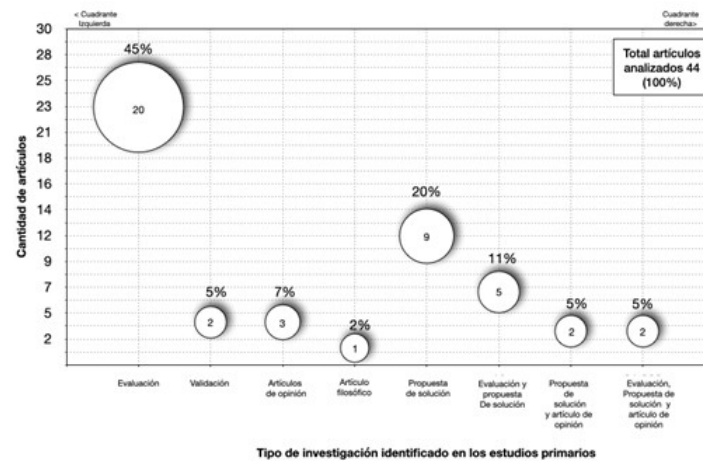


Figura 6. Tipos de investigación

Por otro lado, el 20% de los estudios revisados, representados por 9 artículos, fueron clasificados como propuestas de solución. Esta categoría se basa en la originalidad de las propuestas, su aplicabilidad como solución y las mejoras significativas que ofrecen en comparación con los enfoques anteriores. Entre las propuestas de solución identificadas en estos artículos, varias abordaron el uso de técnicas de procesamiento del lenguaje natural (PNL) para detectar la presencia de olores comunitarios mediante la identificación de emociones o actitudes expresadas por los profesionales en el texto, como la frustración, la ira, la desconfianza, el estrés, la ansiedad, la angustia y la irritabilidad, entre otras. Estas soluciones también incluyen métodos estadísticos para identificar sentimientos en el código, a través de comentarios o registros de incidentes de miembros de una comunidad de desarrollo de PS16. Además, se examina la relación entre los olores comunitarios y los olores codificados. En la PS15, el análisis de regresión se utiliza para examinar la relación entre los olores de la comunidad y la duración de los olores de código, mientras que en la PS17 se emplea un modelo de regresión para analizar la relación entre los olores de la comunidad y la intensidad de los olores de código. Además, se desarrolló una herramienta automatizada llamada CodeFace4Smells, que permite la identificación de 4 olores comunitarios: lobo solitario, silo organizacional, nube negra y cuello de botella. Esta herramienta refleja problemas sociotécnicos que han demostrado ser perjudiciales tanto para los campos de la Ingeniería de Software como para la investigación organizacional. La herramienta también utiliza una heurística para identificar la coordinación entre dos desarrolladores (PS4 – PS6, PS13 y PS23). Además, 1 artículo (2%) fue clasificado como filosófico, ya que presenta la perspectiva del autor sobre los resultados de investigaciones previas, evaluando su validez o cuestionándola. Asimismo, tres artículos (7%) fueron clasificados como artículos de opinión, con base en el punto de vista, experiencia y aportes del autor al debate sobre el tema. Es importante destacar que dos artículos (5%) fueron clasificados simultáneamente como artículos de evaluación, propuesta de solución y artículos de opinión. Otros dos artículos (5%) se clasificaron como propuestas de solución y artículos de opinión, y 5 artículos (11%) se incluyeron en las categorías de evaluación y propuesta

de solución. Finalmente, no se identificaron artículos en los estudios primarios que se ajusten a la categoría de investigación de la experiencia personal.

La clasificación de los 44 estudios primarios según los seis tipos de investigación propuestos por Wieringa et al. (92) revelaron una notable diversidad en los enfoques utilizados. La mayoría de los estudios se organizaron en las categorías de evaluación (45%) y propuesta de solución (20%), reflejando un marcado interés en generar nuevos conocimientos e introducir innovaciones dentro de la Ingeniería de Software. También es evidente que la comunidad científica ha priorizado la generación de conocimiento a través de la aplicación de metodologías detalladas y novedosas, lo que contribuye a la validación y mejora de teorías y prácticas relevantes para la disciplina, con especial énfasis en el desarrollo de software y la importancia del talento humano por sobre procesos y herramientas. Además, algunos estudios adoptan un enfoque multidimensional, abordando diferentes aspectos del problema investigado y proponiendo soluciones integrales.

Resumen de la información actualizada y relevante

Esta sección responde a la siguiente pregunta de investigación:

PI4. ¿Cómo conceptualizan y definen los diferentes autores el término “olor a comunidad” en el contexto de las comunidades de desarrollo de software?

El SLR es un enfoque riguroso que facilita la clarificación de conceptos relacionados con el tema objeto de estudio, en este caso, centrado en el SD y los olores comunitarios. Lo que sigue es una síntesis de los conceptos y aportes aportados por diferentes autores sobre el tema. De acuerdo con las normas PS11, PS12, PS13 y PS14, el concepto de olores comunitarios puede considerarse como antipatrones sociotécnicos y una fuente de deuda social que afecta los procesos de desarrollo, los resultados y la organización de la CDS. También es importante destacar la definición de olores comunitarios establecida por Palomba y Tamburri en PS5, donde se sugiere que “... Los olores comunitarios están relacionados con todo tipo de fenómenos desagradables, por ejemplo: rotación de empleados, malas decisiones arquitectónicas que muchas veces van más allá de causar problemas técnicos en el código; también pueden causar confusión organizacional o incluso una disminución en la estabilidad organizacional de los proyectos de software y/o el fracaso de los proyectos...”. Estos olores también se asocian con los comportamientos de los miembros de la comunidad de desarrollo de software, así como con las políticas y la desalineación entre los objetivos y/o procesos de la organización.

Además, otros autores asocian los olores de la comunidad con las causas que conducen a diferentes tipos de deuda, como la deuda de proceso en PS34 y la deuda técnica en PS3 y PS31. Por otro lado, la SD se manifiesta en problemas técnicos reflejados en artefactos de software como documentación incomprensible y componentes de software incompletos (PS14, PS43). En la Tabla 7 se definen 35 olores comunitarios identificados en la literatura analizada. Si no se gestionan adecuadamente, estos olores pueden provocar la aparición de SD en los equipos de desarrollo de software, ya que se consideran posibles causas de SD.

También se observó que los olores comunitarios con mayor presencia en CSD son: silo organizacional (mencionado en 20 estudios primarios), seguido de nube negra y silencio de radio

(cada uno identificado en 12 estudios primarios). En particular, algunos estudios primarios, como PS7, PS18, PS26 y PS13, mencionan el silencio de radio y el cuello de botella como la misma definición; sin embargo, en este SLR, se tratan como dos olores comunitarios distintos. Además, es importante destacar la observación por parte de PS5 respecto a "... que predecir los olores de la comunidad dentro de una comunidad antes de su manifestación en los plazos de los proyectos de software puede ser fundamental para anticipar condiciones organizativas y sociotécnicas subóptimas antes de que se vuelvan inmanejables...". La Tabla 7 a continuación enumera 10 de los 35 olores comunitarios identificados. La lista completa se puede encontrar en el Anexo A: Tabla 10.

Tabla 7. La comunidad huele en el desarrollo de software

Id. de CS	Nombre CS	Definición de olor comunitario	Referencia
1	Arquitectura por Ósmosis (AO)	Ocurre cuando las decisiones arquitectónicas se toman en base a información incompleta o mal administrada. Algunos miembros de la comunidad utilizan canales mal definidos o estándares diferentes	AP26
		para documentar los cambios, lo que resulta en pérdida de tiempo, dificultad para rastrear decisiones clave y una arquitectura inestable.	
2	Campana arquitectónica (AH)	Ocurre cuando los equipos de arquitectura trabajan de forma remota o con poca interacción con los desarrolladores. Esta falta de contacto directo dificulta la cooperación e impide la correcta resolución de los problemas de implementación derivados de las decisiones arquitectónicas.	AP26
3	Silo organizacional (OS)	Se refiere a la existencia de áreas dentro de la comunidad de desarrolladores que se comunican solo a través de uno o dos representantes, limitando el flujo de información y la integración efectiva del equipo.	Código Postal: No.13 , No.16- No.19, No.23, No.29, No.32, No.35
4	Nube Negra (BC)	Surge cuando la organización no ofrece condiciones que favorezcan la comunicación e interacción entre los miembros de la comunidad. Esto dificulta el intercambio de conocimientos y experiencias durante el desarrollo de software.	AP5- AP7, Contáctenos: No.9 - No.10, No.12-No.13, No.17-No.18, No.24

5	Lobo solitario (LW)	Se manifiesta cuando ciertos miembros actúan de manera independiente, sin tener en cuenta a sus pares ni coordinar esfuerzos. Este comportamiento puede dar lugar a decisiones técnicas no autorizadas, errores de código y retrasos en el proyecto.	Patrón de trabajo: No.7, No.19, No.10, No.12, No.16 - No.18, No.21
6	Cuello de botella (BN)	Ocurre cuando la misma persona interviene en todas las interacciones entre dos o más subcomunidades, dificultando la comunicación fluida. Esto genera retrasos en la producción y una disminución de la eficiencia del equipo.	Contáctenos: No. 2 - No. 2 - No. 2
7	Silencio radial (RS)	Ocurre cuando los procedimientos dentro de la comunidad son demasiado formales y estructurados, causando retrasos en la implementación de cambios y pérdidas de tiempo debido a procesos complejos y poco ágiles.	Contáctenos: No.1, No.14, No.7, No.8, No.8, No.9, No.12, No.13, No.17, No.24, No.25 - No.32
8	Prima-donnas (PD)	Aparece cuando algunos miembros se niegan a aceptar los cambios propuestos por otros, como resultado de una estructura de colaboración ineficiente. Esta resistencia afecta negativamente al flujo de trabajo y a la cohesión del equipo.	AP8, AP16
9	Compartir villanía (SV)	Describe entornos en los que el intercambio de información o experiencias no se considera una prioridad. Esto suele deberse a la falta de espacio, incentivos o apoyo organizativo para fomentar el aprendizaje colaborativo.	AP1, AP8
10	Escaramuza organizativa (OSk)	Ocurre cuando las diferencias entre las culturas organizacionales dificultan la coordinación y gestión de proyectos. Estas diferencias pueden afectar a la productividad y causar retrasos significativos.	AP8, (17)

Siglas. Id: Identificador de olor de la comunidad. CS: Olor de la comunidad. **Fuente:** Elaboración propia.

Discusión

Esta sección presenta los resultados obtenidos de la recopilación de datos de SLR, con el objetivo de proporcionar una visión completa del estado actual de SD y los olores de la comunidad en el desarrollo de software. La discusión se organiza en tres secciones principales: (A) Observaciones

principales, donde se examinan los factores clave que contribuyen a la SD y los olores de la comunidad, se identifican las áreas de investigación futuras, se presentan los beneficios y las posibles soluciones, y se analiza el estado actual de la literatura; y (B) limitaciones de la revisión y análisis de los resultados.

Principales observaciones

El SLR realizado permitió recopilar, analizar, comprender y profundizar los conceptos de comunidad y SD en el contexto de los equipos de desarrollo de software. A continuación se presentan los principales hallazgos:

En las comunidades de desarrollo de software, los profesionales requieren un alto nivel de habilidades interpersonales y blandas. La ausencia o deficiencia de estas habilidades es un factor de riesgo que favorece la aparición de olores comunitarios, reflejando problemas subyacentes en la dinámica laboral. Estos olores de la comunidad pueden afectar negativamente la productividad, la calidad del software y el clima organizacional, impactando directamente en la salud y la sostenibilidad de las comunidades de desarrollo a las que pertenecen los profesionales.

Entre las habilidades blandas fundamentales para los profesionales de las comunidades de desarrollo de software se encuentran la comunicación, la colaboración y la coordinación. Las deficiencias en estas competencias pueden generar malentendidos, afectar la calidad técnica del software y deteriorar la productividad y la cohesión del equipo. En este contexto, los olores comunitarios emergen como indicadores de problemas subyacentes en la dinámica interpersonal. La identificación de estos olores comunitarios permite reconocer situaciones o profesionales que pueden estar generando tensiones, permitiendo la implementación de estrategias preventivas que apoyen el bienestar social y emocional de los equipos. En lugar de ser causas de exclusión, estos hallazgos deben verse como oportunidades de mejora, reconociendo que las intervenciones adecuadas pueden fortalecer tanto a los individuos como al colectivo, fomentando entornos de trabajo más saludables y sostenibles.

Para abordar problemas como los olores comunitarios, se utiliza el concepto de congruencia sociotécnica, entendida como la alineación efectiva entre los aspectos sociales y técnicos dentro de una organización. Esta congruencia implica que los procesos, las tecnologías y las relaciones laborales deben diseñarse de manera integrada, promoviendo tanto la eficiencia organizacional como el bienestar y la satisfacción de los empleados. Es esencial garantizar que las interacciones entre los miembros del equipo sean fluidas y propicias para lograr los objetivos del proyecto.

Además, la diversidad de género es un factor relevante para mitigar los olores comunitarios. Como se ha demostrado en la literatura revisada, los entornos de desarrollo que integran equipos diversos manteniendo la competencia profesional fomentan la creación de entornos propicios para el uso efectivo de las habilidades blandas, como la comunicación, la cooperación y la coordinación entre los miembros de la comunidad.

Limitaciones de la revisión sistemática

Cada una de las etapas involucradas en la realización de este SLR, desde la formulación de los objetivos hasta la recopilación y el análisis de la información obtenida, puede haberse visto afectada por fallas en el muestreo, particularmente en lo que respecta a la precisión de la información recopilada. Algunas de estas limitaciones se presentan a continuación:

Sesgo debido a la ambigüedad conceptual y a la reformulación tardía de los objetivos y preguntas de la investigación: durante el desarrollo del SLR, se identificó una limitación metodológica, relacionada con la formulación inicial de los objetivos y preguntas de investigación, así como el amplio alcance del DS, que puede aplicarse a múltiples dominios. Esta amplitud generó ambigüedad en la delimitación conceptual e impactó la claridad en la selección y extracción de estudios relevantes. En particular, la falta de una definición precisa de la relación entre SD y los olores de la comunidad en contextos de desarrollo de software ágil dificulta la identificación de fuentes relevantes y un análisis coherente de los hallazgos. Como medida correctiva, se reformularon los objetivos y las preguntas de investigación, y se ajustó la cadena de búsqueda para fundamentar la investigación específicamente en el dominio de desarrollo de software. Este refinamiento aumentó la precisión y relevancia de los resultados, fortaleciendo la validez del análisis y la consistencia de los hallazgos obtenidos.

Un posible sesgo identificado en este SLR está relacionado con las restricciones aplicadas durante la selección de artículos, particularmente en términos de lenguaje y período de publicación. La búsqueda se limitó a estudios en inglés, ya que la mayoría de las investigaciones sobre SD en el desarrollo de software se publican en ese idioma; sin embargo, esta decisión puede haber excluido estudios relevantes en otros idiomas, reduciendo así la diversidad del análisis. Asimismo, se estableció como criterio temporal la inclusión de artículos publicados a partir de 2017, bajo la premisa de que los estudios previos fueron actualizados en trabajos más recientes. Aunque estas restricciones tenían como objetivo garantizar la puntualidad y la calidad de las fuentes, también implican un posible sesgo de omisión. Para mitigar este riesgo, los artículos fundacionales se verificaron a través de sus versiones actualizadas, y se dio preferencia a la literatura altamente citada y revisada por pares de bases de datos reconocidas.

Se identificó una limitación en el proceso de búsqueda de información que puede haber afectado la cobertura y claridad del tema bajo análisis. Aunque se consultaron varias bases de datos, cada una posee características distintas y no todas proporcionan el mismo tipo de contenido, lo que da lugar a hallazgos inconsistentes. Inicialmente, se utilizó una cadena de búsqueda general con los términos "deuda social" y "olores comunitarios", que recuperó numerosos artículos no relacionados directamente con la ingeniería de software. Esto dificultó el enfoque del estudio. Para mejorar la pertinencia y precisión de los resultados, la búsqueda se refinó incluyendo los términos "desarrollo de software" y "desarrollo de software ágil", lo que permitió un filtrado más eficaz de los estudios pertinentes y una información más precisa para el análisis.

Uno de los posibles sesgos identificados se relaciona con el proceso de clasificación de los artículos seleccionados. Aunque se aplicaron ciertos criterios, es posible que no hayan sido lo suficientemente rigurosos, lo que podría dar lugar a ambigüedad en la categorización y afectar la perspectiva

general del tema. Esta limitación podría haber influido en la interpretación de los resultados, particularmente al agrupar los artículos por tipos de contribución, enfoque o dominio. Para abordar este sesgo, se revisaron las categorías establecidas y se realizó una validación cruzada basada en descripciones explícitas de los artículos.

Conclusiones

Entre 2017 y 2024, se identificaron cuarenta y cuatro estudios primarios, lo que refleja un creciente interés en comprender los olores de la comunidad dentro de las comunidades de desarrollo de software. Estos estudios, que fueron principalmente exploratorios, permitieron identificar una amplia variedad de enfoques metodológicos y canales de publicación relevantes para este campo. Se centraron en las causas fundamentales de estos olores y su asociación con deficiencias en factores sociotécnicos, como la comunicación, la coordinación y la colaboración, que son clave en entornos ágiles y distribuidos.

Se identificaron y analizaron un total de treinta y cinco olores comunitarios, destacando la frecuente ocurrencia de silos organizacionales, cuellos de botella, nubes negras y lobos solitarios. Estos factores interrumpen el flujo de comunicación, crean puntos críticos de dependencia, fomentan el aislamiento de los miembros del equipo e impiden el acceso equitativo a la información. Tales olores se han relacionado con un aumento de la deuda social, técnica y de procesos, con consecuencias negativas para la calidad del software y el bienestar del equipo.

Un hallazgo clave es que los olores de la comunidad a menudo pasan desapercibidos, ya que tienden a normalizarse dentro de la cultura laboral y no suelen percibirse como una amenaza inmediata. Esta invisibilidad dificulta la detección temprana y la intervención oportuna. En consecuencia, se recomienda avanzar en el desarrollo de métricas e indicadores para su detección, así como diseñar estrategias de mitigación contextualizadas.

Se reconoce que la dimensión social de la deuda no puede separarse de la deuda técnica, ya que ambas están intrínsecamente interconectadas en los procesos de desarrollo. El bienestar emocional y social de los miembros del equipo se convierte así en un factor decisivo para mantener el equilibrio entre la productividad y la sostenibilidad a largo plazo. En este sentido, el fomento de entornos colaborativos, el reconocimiento del rendimiento sobresaliente, la mejora de las habilidades técnicas e interpersonales y el fomento de la participación activa de todos los miembros contribuyen significativamente a mitigar la incidencia de los olores de la comunidad.

Como trabajo futuro, se propone investigar más a fondo los treinta olores comunitarios restantes, identificando sus causas, efectos y métricas potenciales, y documentando sus características desde varias perspectivas conceptuales. Además, se considera fundamental reducir la heterogeneidad terminológica mediante el diseño de una taxonomía que sistematice los olores comunitarios y su relación con la dimensión social de la deuda en la ingeniería de software. Esta taxonomía constituirá una contribución tanto conceptual como práctica, avanzando en la gestión de estos fenómenos en equipos de desarrollo de software distribuido y colaborativo.

Reconocimientos

Los profesores Eydy Suárez Brieva y César Pardo Calvache desean agradecer a la Universidad Popular del Cesar y a la Universidad del Cauca, donde actualmente se desempeñan como profesores titulares, respectivamente. Asimismo, los estudiantes Jorge Enrique Bejarano Guar y Nicolle Daniela Urbano expresan su agradecimiento a la Universidad del Cauca, donde actualmente cursan la carrera de Ingeniería de Sistemas.

Declaración de contribución de autoría de CreditT

Conceptualización - Ideas: Eydy Suárez Brieva, Jorge Enrique Bejarano Guar, Nicole Daniela Urbano Palechor y César Jesús Pardo Calvache. Curación de datos: Eydy Suárez Brieva, Jorge Enrique Bejarano Guar y Nicole Daniela Urbano Palechor. Análisis formal: Eydy Suárez Brieva y Jorge Enrique Bejarano Guar. Investigación: Eydy Suárez Brieva, Jorge Enrique Bejarano Guar y Nicole Daniela Urbano Palechor. Metodología: Eydy Suárez Brieva y Jorge Enrique Bejarano Guar. Dirección de Proyecto: Eydy Suárez Brieva y César Jesús Pardo Calvache. Recursos: Eydy Suárez Brieva, Jorge Enrique Bejarano Guar y Nicole Daniela Urbano Palechor. Software: Eydy Suárez Brieva, Jorge Enrique Bejarano Guar y Nicole Daniela Urbano Palechor. Supervisión: Eydy Suárez Brieva y César Jesús Pardo Calvache. Validación: Eydy Suárez Brieva y César Jesús Pardo Calvache. Redacción - borrador original - Elaboración: Eydy Suárez Brieva y Jorge Enrique Bejarano Guar. Redacción - revisión y edición - Elaboración: Eydy Suárez Brieva, Jorge Enrique Bejarano Guar, Nicole Daniela Urbano Palechor y César Jesús Pardo Calvache.

Financiación: no declara. Conflicto de intereses: no declara. Aspecto ético: no declara

Referencias

1. Gómez Fuentes M del C, Cervantes Ojeda J, González Pérez P. Fundamentos de Ingeniería de Software 1st ed. UAM UC, editor. Ciudad de México: Universidad Autónoma Metropolitana; 2019, 1–277 p. Available from: <https://shorturl.at/gjGIL>
2. Yilmaz M, O'Connor R V., Clarke P. Effective Social Productivity Measurements during Software Development - An Empirical Study. International Journal of Software Engineering and Knowledge Engineering. 2016 Apr 1;26(3):457–90. <https://doi.org/10.1142/S0218194016500194>
3. Galster M, Tamburri DA, Kazman R. Towards Understanding the Social and Organizational Dimensions of Software Architecting. ACM SIGSOFT Software Engineering Notes. 2017;42(3):24–5. <https://doi.org/10.1145/3127360.3127374>
4. Sievi-Korte O, Fagerholm F, Systä K, Mikkonen T. Dimensions of Consistency in GSD: Social Factors, Structures and Interactions. Proceedings Lecture Notes in Computer Science. 2020;12562:315–30. https://doi.org/10.1007/978-3-030-64148-1_20
5. Brieva ES, Pardo C, Villarreal V. Deuda Social y Riesgos psicosociales en entornos de desarrollo de software ágil: Análisis preliminar de una Revisión Sistemática de la Literatura. In: 2024 IEEE VII Congreso Internacional en Inteligencia Ambiental, Ingeniería de Software y Salud Electrónica y Móvil (AmITIC). 2024. p. 1–8. <https://doi.org/10.1109/AmITIC62658.2024.10747594>
6. Vizcaíno A, García F, Piattini M. Visión General del Desarrollo Global de Software. International Journal of Information Systems and Software Engineering for Big Companies (IJISEBC) 2014;1(1):8–22. www.ijisebc.com

7. Jiménez M, Piattini M, Vizcaíno A. Challenges and improvements in distributed software development: A systematic review. *Advances in Software Engineering*. 2009; 1–14. <https://doi.org/10.1155/2009/710971>
8. Vizcaíno A, Valencia D, Soto JP, García-Mundo L, Piattini M. ¿Qué desafíos presenta el desarrollo global del software? Aprende jugando. In: XXI Jornadas de Ingeniería del Software y Bases de Datos. México; 2016. p. 605–8. https://investigadores.unison.mx/ws/portalfiles/portal/8020882/CEDI_2016_paper_47.pdf
9. García GD, Pardo Calvache CJ, Rodríguez FJA. Society 5.0 and Soft Skills in Agile Global Software Development. *Revista Iberoamericana de Tecnologías del Aprendizaje*. 2022 May 1;17(2):197–207. <https://doi.org/10.1109/RITA.2022.3166966>
10. Mens T. An ecosystemic and socio-technical view on software maintenance and evolution. In: *Proceedings - 2016 IEEE International Conference on Software Maintenance and Evolution, ICSME 2016*. Institute of Electrical and Electronics Engineers Inc.; 2017. p. 1–8. <https://doi.org/10.1109/ICSME.2016.19>
11. Gote C, Perri V, Zingg C, Casiraghi G, Arzig C, von Gernler A, et al. Locating community smells in software development processes using higher-order network centralities. *Soc Netw Anal Min*. 2023 Oct 1;13(1):1–28. <https://doi.org/10.1007/s13278-023-01120-w>
12. Olsson J, Risfelt E, Besker T, Martini A, Torkar R. Measuring affective states from technical debt: A psychoempirical software engineering experiment. *Empir Softw Eng*. 2021 Sep 1;26(5). <https://doi.org/10.1007/s10664-021-09998-w>
13. Tamburri DA, Milano D, Kazman R. The Architect's Role in Community Shepherdling. <https://doi.org/10.1109/MS.2016.144>
14. Tamburri DA, Palomba F, Kazman R. Exploring Community Smells in Open-Source: An automated approach. *IEEE Transactions on Software Engineering*. 2017;14(8):630–52. <https://doi.org/10.1109/TSE.2019.2901490>
15. Palomba F, Tamburri DA. Predicting the emergence of community smells using socio-technical metrics: A machine-learning approach. *Journal of Systems and Software*. 2021;171. <https://doi.org/10.1016/j.jss.2020.110847>
16. Huang Z, Shao Z, Fan G, Gao J, Zhou Z, Yang K, et al. Predicting Community Smells' Occurrence on individual developers by sentiments. In: ICPC, editor. *2021 IEEE/ACM 29th International Conference on Program Comprehension (ICPC)*. Madrid : IEEE; 2021. p. 230–41. <https://doi.org/10.1109/ICPC52881.2021.00030>
17. Caballero-Espinosa E, Carver JC, Stowers K. Community smells—The sources of social debt: A systematic literature review. *Inf Softw Technol*. 2023;153. <https://doi.org/10.1016/j.infsof.2022.107078>
18. Catolino G, Palomba F, Tamburri DA, Serebrenik A. Understanding Community Smells Variability: A Statistical Approach. *Proceedings - International Conference on Software Engineering*. 2021; 77–86. <https://doi.org/10.1109/ICSE-SEIS52602.2021.00017>
19. Almarimi N, Ouni A, Chouchen M, Mkaouer MW. Improving the detection of community smells through socio-technical and sentiment analysis. *Journal of Software: Evolution and Process*. 2022;35 <https://doi.org/10.1002/smr.2505>
20. Saeeda H, Ovais Ahmad M, Gustavsson T. Navigating social debt and its link with technical debt in large-scale agile software development projects. *Software Quality Journal*. 2024;32:1581–1613 <https://doi.org/10.1007/s11219-024-09688-y>

21. Saeeda H, Ovais Ahamd M, Gustavsson T. 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*. 2024;18(1):240101. <https://doi.org/10.37190/e-Inf240101>
22. Palomba F, Serebrenik A, Zaidman A. Social Debt Analytics for improving the management of Software evolution tasks. In: *CEUR Workshop Proceedings*, editor. Belgian Netherlands Software evolution Symposium. Belgium; 2023;19–21. <https://shorturl.at/ahoX7>
23. Moreno Jiménez, Bernardo. "Factores y riesgos laborales psicosociales: conceptualización, historia y cambios actuales." *Medicina y Seguridad del trabajo*. 2011; 57: 4-19.
24. Organización Internacional del Trabajo. La organización del trabajo y los riesgos psicosociales: una mirada de género. Organización Internacional del Trabajo. San José; 2013 https://scielo.isciii.es/scielo.php?pid=S0465-546X2011000500002&script=sci_arttext&lng=en
25. Tamburri DA, Kruchten P, Lago P, Vliet H van. Social debt in software engineering: insights from industry. *Journal of Internet Services and Applications*. 2015;6(1). <https://doi.org/10.1186/s13174-015-0024-6>
26. De Stefano M, Iannone E, Pecorelli F, Tamburri DA. Impacts of software community patterns on process and product: An empirical study. *Sci Comput Program*. 2022; 214:102731. <https://doi.org/10.1016/j.scico.2021.102731>
27. Lambiase S, Catolino G, Tamburri DA, Serebrenik A, Palomba F, Ferrucci F. Good fences make good neighbours? on the impact of cultural and geographical dispersion on community smells. In: *Proceedings of the 2022 ACM/IEEE 44th International Conference on Software Engineering: Software Engineering in Society*. New York, NY, USA: Association for Computing Machinery; 2022. p. 67–78. <https://doi.org/10.1145/3510458.3513015>
28. Tamburri DA. Software Architecture Social Debt: Managing the Incommunicability Factor. *IEEE Trans Comput Soc Syst*. 2019;6(1):20–37. <https://doi.org/10.1109/TCSS.2018.2886433>
29. Dreesen T, Hennel P, Rosenkranz C, Kude T. The second vice is lying, the first is running into debt. Antecedents and mitigating practices of social debt: An exploratory study in distributed software development teams. *Proceedings of the Annual Hawaii International Conference on System Sciences*. 2021; 6826–35. <https://doi.org/10.24251/HICSS.2021.818>
30. Mumtaz H, Paradis C, Palomba F, Tamburri DA, Kazman R, Blincoe K. A Preliminary Study on the Assignment of GitHub Issues to Issue Commenters and the Relationship with Social Smells. Vol. 1, *Proceedings - 15th International Conference on Cooperative and Human Aspects of Software Engineering, CHASE 2022*. Association for Computing Machinery. 2022; 61–65 <https://doi.org/10.1145/3528579.3529181>
31. Lambiase S, Catolino G, Tamburri DA, Serebrenik A, Palomba F, Ferrucci F. Good fences make good neighbours? 2022;67–78. <https://doi.org/10.1109/ICSE-SEIS55304.2022.9793992>
32. Pontificia Universidad Javeriana, Ministerio de la Protección Social. Batería de instrumentos para la evaluación de factores de riesgo psicosocial. 2010 <https://shorturl.at/kuE49>
33. R. S. Sangwan, K. W. Jablokow and J. F. DeFranco, "Asynchronous Collaboration: Bridging the Cognitive Distance in Global Software Development Projects," in *IEEE Transactions on Professional Communication*. 2020; 63(4):361–371 <https://doi.org/10.1109/TPC.2020.3029674>
34. Almarimi N, Ouni A, Mkaouer MW. Learning to detect community smells in open source software projects. *Knowl Based Syst*. 2020 Sep 27;204. <https://doi.org/10.1016/j.knosys.2020.106201>
35. Palomba F, Tamburri DA. Predicting the emergence of community smells using socio-technical metrics: A machine-learning approach. *Journal of Systems and Software*. 2021;171. <https://doi.org/10.1016/j.jss.2020.110847>

36. Catolino G, Palomba F, Tamburri DA, Serebrenik A, Ferrucci F. Refactoring Community Smells in the Wild: The Practitioner's Field Manual. In: Proceedings - 2020 ACM/IEEE 42nd International Conference on Software Engineering: Software Engineering in Society, ICSE-SEIS 2020. 2020; 25–34. <https://doi.org/10.1145/3377815.3381380>
37. Palomba F, Tamburri DA. Predicting the emergence of community smells using socio-technical metrics: A machine-learning approach. Journal of Systems and Software. 2021;171:110847. <https://doi.org/10.1016/j.jss.2020.110847>
38. Raza B, Ahmad R, Nasir MH, Fauzi SS, Raza MA. Assessing the impact of socio-technical congruence in software development: a systematic literature review. KJS 2021;49(1). <https://journalskuwait.org/kjs/index.php/KJS/article/view/9240>
39. Tahsin N, Rauf A, Manzoor A, Anwer MU. Community smells—The sources of social debt: A systematic literature review. Systematic Literature Review and Meta-analysis Research. 2023;3(4). Available from: <https://doi.org/10.54480/slr.m.v3i4.51>
40. Raza B, Ahmad R, Nasir MHNBM, Fauzi SSM. Socio-Technical Congruence as an Emerging Concept in Software Development: A Scientometric Analysis and Critical Literature Review. IEEE Access. 2021;9:129051–77. <https://doi.org/10.1109/ACCESS.2021.3113637>
41. Kitchenham B, Charters S. Guidelines for performing Systematic Literature Reviews in Software Engineering Version 2.3. Vol. 45, Durham University. 2007 <https://acortar.link/dzpbgk>
42. Van Solingen R, Basili V, Caldiera G, Rombach HD. Goal Question Metric (GQM) Approach. Encyclopedia of Software Engineering. 2002. <https://doi.org/10.1002/0471028959.sof142>
43. Wieringa R, Maiden N, Mead N, Rolland C. Requirements engineering paper classification and evaluation criteria: A proposal and a discussion. Requir Eng. 2006;11(1):102–7. <https://doi.org/10.1007/s00766-005-0021-6>
44. Kitchenham BA, Charters S. Guidelines for performing Systematic Literature Reviews in Software Engineering (Software Engineering Group, Department of Computer Science, Keele Technical Report EBSE 2007- 001 Keele University and Durham University Joint Report. 2007;(January).
45. Wieringa R, Maiden N, Mead N, Rolland C. Requirements engineering paper classification and evaluation criteria: A proposal and a discussion. Requir Eng. 2006 11(1):102–7. <https://doi.org/10.1007/s00766-005-0021-6>
46. Tamburri DA, Kruchten P, Lago P, Van Vliet H. What is social debt in software engineering? In: 2013 6th International Workshop on Cooperative and Human Aspects of Software Engineering, CHASE 2013 - Proceedings. 2013; 93–6. <https://doi.org/10.1109/CHASE.2013.6614739>
47. Tamburri DA, Nitto E Di. When Software Architecting Leads to Social Debt.2005. <http://tinyurl.com/ljqgay9>
48. Besker T, Ghanbari H, Martini A, Bosch J. The influence of Technical Debt on software developer morale. Journal of Systems and Software. 2020;167. <https://doi.org/10.1016/j.jss.2020.110586>
49. Besker T, Martini A, Bosch J. The use of incentives to promote technical debt management. Inf Softw Technol. 2022;142: 106740 <https://doi.org/10.1016/j.infsof.2021.106740>
50. Martini A, Besker T, Bosch J. Technical Debt tracking: Current state of practice: A survey and multiple case study in 15 large organizations. Sci Comput Program. 2018;163:42–61. <https://doi.org/10.1016/j.scico.2018.03.007>
51. Almarimi N, Ouni A, Mkaouer MW. Learning to detect community smells in open source software projects. Knowl Based Syst. 2020 ;204. <https://doi.org/10.1016/j.knosys.2020.106201>

52. Palomba F, Tamburri DA. Predicting the emergence of community smells using socio-technical metrics: A machine-learning approach. *Journal of Systems and Software*. 2021;171:110847 <https://doi.org/10.1016/j.jss.2020.110847>
53. De Stefano M, Iannone E, Pecorelli F, Tamburri DA. Impacts of software community patterns on process and product: An empirical study. *Sci Comput Program*. 2022; 214: 102731 <https://doi.org/10.1016/j.scico.2021.102731>
54. Gemma Catolino, Fabio Palomba, Damian A. Tamburri, Alexander Serebrenik, Filomena Ferrucci. Refactoring Community Smells in the Wild: The Practitioner's Field Manual. *ICSE*. 2020;25–34. <https://doi.org/10.1145/3377815.3381380>
55. Almarimi N, Ouni A, Chouchen M, Saidani I, Mkaouer MW. On the detection of community smells using genetic programming-based ensemble classifier chain. In: *Proceedings - 2020 ACM/IEEE 15th International Conference on Global Software Engineering, ICGSE 2020*. Association for Computing Machinery, Inc. 2020; 43–54. <https://doi.org/10.1145/3372787.3390439>
56. Catolino G, Palomba F, Tamburri DA, Serebrenik A, Ferrucci F. Gender diversity and women in software teams: How do they affect community smells? 2019 IEEE/ACM 41st International Conference on Software Engineering: Software Engineering in Society (ICSE-SEIS). IEEE, 2019. <https://doi.org/10.1109/ICSE-SEIS.2019.00010>
57. De Stefano M, Pecorelli F, Tamburri DA, Palomba F, De Lucia A. Splicing Community Patterns and Smells: A Preliminary Study. 42nd International Conference on Software Engineering Workshops. 2020: 703–10. <https://doi.org/10.1145/3387940.3392204>
58. Canedo ED, Mendes F, Cerqueira A, Okimoto M, Pinto G, Bonifacio R. Breaking one barrier at a time: How women developers cope in a men-dominated industry. In: *ACM International Conference Proceeding Series*. Association for Computing Machinery. 2021;378–87. <https://doi.org/10.1145/3474624.3474638>
59. Catolino G, Palomba F, Tamburri DA, Serebrenik A. Understanding Community Smells Variability: A Statistical Approach. In: *Proceedings - International Conference on Software Engineering*. IEEE Computer Society. 2021;77–86. <https://doi.org/10.1109/ICSE-SEIS52602.2021.00017>
60. Tamburri DA, Palomba F, Kazman R. Exploring Community Smells in Open-Source: An Automated Approach. *IEEE Transactions on software Engineering*. 2019; 47(3): 630-652. <https://doi.org/10.1109/TSE.2019.2901490>
61. Tamburri DA. Software Architecture Social Debt: Managing the Incommunicability Factor. *IEEE Trans Comput Soc Syst*. 2019;6(1):20–37. <https://doi.org/10.1109/TCSS.2018.2886433>
62. Palomba F, Tamburri DA, Serebrenik A, Zaidman A, Fontana FA, Oliveto R. How do community smells influence code smells? In: *Proceedings - International Conference on Software Engineering*. 2018; 240–1. <https://doi.org/10.1145/3183440.3194950>
63. Huang Z, Shao Z, Fan G, Gao J, Zhou Z, Yang K, et al. Predicting Community Smells' Occurrence on Individual Developers by Sentiments. 2021;12. <https://doi.org/10.1109/ICPC52881.2021.00030>
64. Catolino G, Palomba F, Tamburri DA, Serebrenik A, Ferrucci F. Gender Diversity and Community Smells: Insights from the Trenches. *IEEE Softw*. 2020;37(1):10–6. <https://doi.org/10.1109/MS.2019.2944594>
65. Palomba F, Andrew Tamburri D, Arcelli Fontana F, Oliveto R, Zaidman A, Serebrenik A. Beyond Technical Aspects: How Do Community Smells Influence the Intensity of Code Smells? *IEEE Transactions on Software Engineering*. 2021;47(1):108–29. <https://doi.org/10.1109/TSE.2018.2883603>

66. Eken B, Palma F, Ay   B, Ay   T. An empirical study on the effect of community smells on bug prediction. *Software Quality Journal*. 2021;29(1):159–94. <https://doi.org/10.1007/s11219-020-09538-7>
67. Tamburri TU DA, datamburri -jads, Rick Kazman tuenl, Van den Heuvel Universiteit van Tilburg -JADS WJAMvdHeuvel WJ. Splicing Community and Software Architecture Smells in Agile Teams: An industrial Study. 2019. <https://doi.org/10.24251/HICSS.2019.843>
68. Tamburri DA, Kruchten P, Lago P, Vliet H van. Social debt in software engineering: insights from industry. *Journal of Internet Services and Applications*. 2015;6(1). <https://doi.org/10.1186/s13174-015-0024-6>
69. Dreesen T, Hennel P, Rosenkranz C, Kude T. "The second vice is lying, the first is running into debt." Antecedents and mitigating practices of social debt: An exploratory study in distributed software development teams. 2021;6826–35. <https://doi.org/10.24251/HICSS.2021.818>
70. Ahammed T, Asad M, Sakib K. Understanding the Relationship between Missing Link Community Smell and Fix-inducing Changes. 2021;469–75. <https://doi.org/10.5220/0010500604690475>
71. Sarmiento C, Massoni T, Serebrenik A, Catolino G, Tamburri D, Palomba F. Gender Diversity and Community Smells: A Double-Replication Study on Brazilian Software Teams. 2022 IEEE International Conference on Software Analysis, Evolution and Reengineering, 2022; 273–83. <https://doi.org/10.1109/SANER53432.2022.00043>
72. Lambiase S, Catolino G, Tamburri DA, Serebrenik A, Palomba F, Ferrucci F. Good fences make good neighbours? In Association for Computing Machinery (ACM); 2022; 67–78. <https://doi.org/10.1109/ICSE-SEIS55304.2022.9793992>
73. Mauerer W, Joblin M, Tamburri DA, Paradis C, Kazman R, Apel S. In Search of Socio-Technical Congruence: A Large-Scale Longitudinal Study. *IEEE Transactions on Software Engineering*. 2022;48(8):3159–84. <https://doi.org/10.1109/TSE.2021.3082074>
74. Nadri R, Rodriguez-Perez G, Nagappan M. On the Relationship Between the Developer's Perceptible Race and Ethnicity and the Evaluation of Contributions in OSS. *IEEE Transactions on Software Engineering*. 2022;48(8):2955–68. <https://doi.org/10.1109/TSE.2021.3073773>
75. Prana GAA, Ford D, Rastogi A, Lo D, Purandare R, Nagappan N. Including Everyone, Everywhere: Understanding Opportunities and Challenges of Geographic Gender-Inclusion in OSS. *IEEE Transactions on Software Engineering*. 2021; <https://doi.org/10.1109/TSE.2021.3092813>
76. Mumtaz H, Paradis C, Palomba F, Tamburri DA, Kazman R, Blincoe K. A preliminary study on the assignment of GitHub issues to issue commenters and the relationship with social smells. 15th International Conference on Cooperative and Human Aspects of Software Engineering. 2022; 61–5. <https://doi.org/10.1145/3528579.3529181>
77. Pigazzini I, Fontana FA, Walter B. A study on correlations between architectural smells and design patterns. *Journal of Systems and Software*. 2021;178:110984. <https://doi.org/10.1016/j.jss.2021.110984>
78. Soliman M, Avgeriou P, Li Y. Architectural design decisions that incur technical debt — An industrial case study. *Information-and-Software-Technology*. 2021;139. <https://doi.org/10.1016/j.infsof.2021.106669>
79. Aksekili AY, Stettina CJ. Women in Agile: The Impact of Organizational Support for Women's Advancement on Teamwork Quality and Performance in Agile Software Development Teams. *Lecture Notes in Business Information Processing*. 2021;408:3–23. https://doi.org/10.1007/978-3-030-67084-9_1

80. Bjarnason E, Gislason Bern B, Svedberg L. Inter-team communication in large-scale co-located software engineering: a case study. *Empir Softw Eng*. 2022;27(2). <https://doi.org/10.1007/s10664-021-10027-z>
81. Martini A, Stray V, Moe NB. Technical-, Social- and Process Debt in Large-Scale Agile: An Exploratory Case-Study. In: *Lecture Notes in Business Information Processing*. Springer Verlag. 2019; 112–9. https://doi.org/10.1007/978-3-030-30126-2_14
82. Tamburri DA, Palomba F, Serebrenik A, Zaidman A. Discovering community patterns in open-source: a systematic approach and its evaluation. Vol. 24, *Empirical Software Engineering*. Empirical Software Engineering; 2019;1369–1417 <https://doi.org/10.1007/s10664-018-9659-9>
83. Sievi-Korte O, Fagerholm F, Systä K, Mikkonen T. Dimensions of Consistency in GSD: Social Factors, Structures and Interactions. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. 2020;12562:315–30. https://doi.org/10.1007/978-3-030-64148-1_20
84. Stefano M De, Pecorelli F, Tamburri DA, Palomba F, Lucia A De. Refactoring Recommendations Based on the Optimization of Socio-Technical Congruence. In: *2020 IEEE International Conference on Software Maintenance and Evolution, ICSME*. 2020;794–6. <https://doi.org/10.1109/ICSME46990.2020.00094>
85. Ahammed T, Asad M, Sakib K. Understanding the Involvement of Developers in Missing Link Community Smell: An Exploratory Study on Apache Projects. *International Conference on Evaluation of Novel Approaches to Software Engineering, ENASE*. 2020; 469–75. <https://doi.org/10.5220/0010500604690475>
86. Lavallée M, Robillard PN. Are we working well with others? How the multi team systems impact software quality. *E-Informatica Software Engineering Journal*. 2018;12(1):117–31. <https://doi.org/10.5277/e-inf180105>
87. Martini A, Bosch J. Revealing social debt with the CAFFEA framework: An antidote to architectural debt. *Proceedings - 2017 IEEE International Conference on Software Architecture Workshops*. 2017;179–81. <https://doi.org/10.1109/ICSAW.2017.42>
88. Serebrenik A. Emotional labor of software engineers. 2017;1-6. <https://tinyurl.com/46n66wmX>
89. Palomba F, Serebrenik A, Zaidman A. Social Debt Analytics for Improving the Management of Software Evolution Tasks. 16th Edition of the BELgian-NEtherlands Software EVOLution Symposium.2017; 18-21 <https://tinyurl.com/mr3edacy>
90. M. O. Ahmad. Psychological Safety, Leadership and Non-Technical Debt in Large-Scale Agile Software Development. In: *Proceedings of the 18th Conference on Computer Science and Intelligence Systems*. IEEE; 2023; 327–34. <https://doi.org/10.15439/2023F8595>
91. C. Gote. Locating community smells in software development processes using higher-order network centralities. *Soc Netw Anal Min*. 2023;13:1–28. <https://doi.org/10.1007/s13278-023-01120-w>
92. Wieringa R, Maiden N, Mead N, Rolland C. Requirements engineering paper classification and evaluation criteria: A proposal and a discussion. *Requir Eng*. 2006;11(1):102–7. <https://doi.org/10.1007/s00766-005-0021-6>

Anexos

Tabla 8. Criterios de relevancia y pertinencia

Id criterio	Criterio	Puntuación para las respuestas		
		+1	0	-1
C1	El estudio tiene como tema principal la investigación de los OC o DS en el desarrollo de software.	Si	Parcialmente	No
C2	El estudio presenta una descripción precisa del problema de investigación planteado.	Si	Parcialmente	No
C3	El estudio tiene un procedimiento de investigación estructurado y claro.	Si	Parcialmente	No
C4	El estudio facilita una clara definición sobre la DS.	Si	Parcialmente	No
C5	El estudio propone un conjunto de elementos a considerar para evaluar los OC o DS en el desarrollo de software.	Si	Parcialmente	No
C6	El estudio presenta los resultados obtenidos de manera clara.	Si	Parcialmente	No
C7	El estudio expone de manera clara los aportes enfocados a la industria y/o academia.	Si	Parcialmente	No
C8	El estudio describe claramente la discusión de las limitaciones del proceso de investigación realizado y del análisis de los resultados obtenidos.	Si	Parcialmente	No
C9	El estudio presenta los trabajos futuros y brechas de investigación de manera clara.	Si	Parcialmente	No
C10	El estudio ha sido publicado en una revista, conferencia o congreso relevante. Se utilizó la clasificación de cuartiles propuestos por Scimago (https://www.scimagojr.com/) para clasificar las revistas y el ranking del Computing Research & Education (CORE, http://portal.core.edu.au/conf-ranks/) para congresos y conferencias.	Muy relevante (cuartil Q1 para revistas y A+ para congresos y conferencias)	Relevante (cuartiles Q2 y Q3 para revistas y A o B para congresos y conferencias)	No relevante (cuartil Q4 para revistas y C o no clasificados para congresos y conferencias)
C11	El estudio ha sido citado por otros autores (según el índice de citas de Google Scholar).	Ha sido citado por más de diez autores	Entre uno y diez autores	No ha sido citado hasta el momento

Tabla 9. Puntuación obtenida por artículo criterios de relevancia y pertinencia

Artículo	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	Total
AP1	+1	+1	0	0	-1	0	0	-1	0	0	+1	1
AP2	+1	+1	0	0	0	+1	+1	0	0	+1	-1	4
AP3	0	0	+1	0	-1	+1	+1	-1	+1	0	+1	3
AP4	0	0	+1	-1	0	0	0	+1	0	0	+1	2
AP5	+1	0	+1	0	0	+1	+1	0	+1	+1	+1	7
AP6	0	0	0	-1	-1	0	0	0	0	+1	-1	-2
AP7	0	1	0	0	0	1	0	0	0	-1	+1	2
AP8	1	1	1	0	0	1	0	-1	0	-1	+1	3
AP9	-1	0	0	-1	-1	+1	0	0	0	-1	+1	-2
AP10	+1	+1	0	+1	0	0	0	+1	+1	-1	+1	5
AP11	+1	+1	0	-1	-1	+1	+1	0	0	-1	-1	0
AP12	0	+1	+1	+1	0	+1	+1	+1	+1	0	0	7
AP13	+1	+1	0	+1	0	+1	+1	0	0	1	+1	7

AP14	+1	0	+1	+1	0	0	+1	0	+1	1	+1	7
AP15	+1	+1	0	+1	0	+1	0	+1	0	-1	+1	5
AP16	0	+1	0	0	0	+1	0	+1	0	-1	0	2
AP17	+1	0	0	-1	+1	+1	+1	+1	+1	1	+1	7
AP18	+1	+1	+1	+1	+1	+1	0	+1	+1	1	+1	10
AP19	0	+1	+1	+1	0	+1	+1	0	+1	+1	-1	6
AP20	+1	+1	+1	0	+1	+1	0	0	0	+1	0	6
AP21	0	+1	+1	0	+1	+1	+1	+1	+1	-1	-1	5
AP22	+1	+1	+1	+1	+1	+1	0	0	+1	-1	0	6
AP23	+1	+1	0	0	+1	+1	+1	+1	0	-1	0	5
AP24	+1	+1	+1	+1	+1	+1	+1	+1	0	0	0	8
AP25	+1	+1	+1	0	+1	+1	+1	+1	0	-1	-1	5
AP26	+1	+1	+1	0	+1	+1	+1	0	+1	+1	0	8
AP27	+1	+1	+1	+1	+1	+1	+1	+1	0	0	0	8
AP28	+1	+1	+1	0	-1	+1	+1	+1	0	+1	0	6
AP29	0	+1	+1	0	+1	+1	+1	+1	0	+1	0	7
AP30	+1	0	+1	+1	0	+1	+1	+1	+1	+1	-1	7
AP31	+1	+1	+1	-1	+1	+1	+1	+1	0	+1	0	7
AP32	0	+1	0	-1	-1	+1	+1	+1	-1	-1	-1	-1
AP33	-1	1	0	-1	-1	1	0	1	1	-1	0	0
AP34	-1	1	0	-1	-1	1	0	1	1	-1	0	0
AP35	0	1	0	-1	0	1	1	0	1	-1	1	3
AP36	-1	1	1	-1	0	1	1	1	1	-1	0	3
AP37	0	1	1	-1	0	1	1	1	1	-1	1	5
AP38	-1	1	1	-1	-1	1	1	1	1	-1	1	3
AP39	-1	1	1	-1	-1	1	1	1	1	-1	0	2
AP40	0	1	1	-1	1	1	1	1	1	-1	0	5
AP41	-1	1	1	-1	1	1	-1	1	1	-1	0	2
AP42	1	1	1	1	1	1	1	1	1	-1	0	8
AP43	1	1	1	1	1	1	1	1	1	-1	0	8
AP44	1	1	1	1	1	1	1	1	1	-1	1	9

Tabla 10. Olores comunitarios identificados en el desarrollo de software

Id OC	Nombre OC	Definición del OC	Ref.
1	Arquitectura por osmosis. (AO)	Se presenta cuando las decisiones arquitectónicas se toman con base en información incompleta o mal gestionada. Algunos miembros de la comunidad utilizan canales poco definidos o estándares distintos para documentar los cambios, lo que genera pérdida de tiempo, dificultades para rastrear decisiones clave y una arquitectura inestable.	A26
2	Capucha arquitectónica (CA)	Ocurre cuando los equipos encargados de la arquitectura trabajan de forma remota o con escasa interacción con los desarrolladores. Esta falta de contacto directo dificulta la cooperación e impide resolver adecuadamente problemas de implementación derivados de decisiones arquitectónicas.	A26

3	Silo organizacional (SO)	Hace referencia a la existencia de áreas dentro de la comunidad de desarrolladores que se comunican únicamente a través de uno o dos representantes, limitando el flujo de información y la integración efectiva del equipo.	A4, A5, A6, A7, A8, A9, A10, A12, A13, A16, A17, A18, A19, A23, A24, A25, A26, A29, A32, A35,
4	Nube negra (NG)	Surge cuando la organización no ofrece condiciones que favorezcan la comunicación y la interacción entre los miembros de la comunidad. Esto dificulta el intercambio de conocimientos y experiencias durante el desarrollo del software.	A5, A6, A7, A9, A10, A12, A13, A17, A18, A24, A25, A26
5	Lobo solitario (LS)	Se manifiesta cuando ciertos miembros actúan de manera independiente, sin considerar a sus compañeros ni coordinar esfuerzos. Este tipo de comportamiento puede llevar a decisiones técnicas no autorizadas, errores en el código y retrasos en el proyecto.	A7, A9, A10, A12, A16, A17, A18, A21, A26
6	Cuello de botella (CB)	Ocurre cuando una misma persona interviene en todas las interacciones entre dos o más subcomunidades, dificultando la comunicación fluida. Esto genera demoras en la producción y una caída en la eficiencia del equipo.	A6, A7, A10, A13, A16, A18
7	Silencio radial (SR)	Se presenta cuando los procedimientos dentro de la comunidad son demasiado formales y estructurados, lo que provoca retrasos en la implementación de cambios y pérdida de tiempo debido a trámites complejos y poco ágiles.	A1, A4, A7, A8, A9, A12, A13, A17, A24, A25, A26, A32
8	Prima-donnas (PD)	Aparece cuando algunos miembros se niegan a aceptar cambios propuestos por otros, como resultado de una estructura de colaboración ineficiente. Esta resistencia afecta negativamente el flujo de trabajo y la cohesión del equipo.	A8, A16, A26
9	Compartir villanía (CV)	Describe entornos donde compartir información o experiencias no se considera una actividad prioritaria. Esto suele deberse a la falta de espacios, incentivos o apoyo organizacional para fomentar el aprendizaje colaborativo.	A1, A8, A26

10	Escaramuza Organizacional (EO)	Se da cuando las diferencias entre culturas organizacionales dificultan la coordinación y gestión de proyectos. Estas diferencias pueden impactar la productividad y provocar retrasos significativos.	A8, A26
11	Desafío a la solución (DES)	Se refiere a tensiones que surgen cuando grupos dentro de una misma comunidad, con valores o antecedentes técnicos similares, defienden visiones opuestas sobre decisiones técnicas o estratégicas, generando conflictos en reuniones clave.	A8, A14, A26
12	Factor camión (FC)	Se evidencia cuando gran parte de la información relevante del proyecto está centralizada en uno o dos desarrolladores, lo que representa un riesgo para la continuidad del trabajo y limita la autonomía del equipo.	A8, A9, A18, A25
13	Desarrollador maloliente (DM)	Hace referencia a desarrolladores impactados por olores comunitarios de tipo organizacional, como el LS, el SO y CB. Estos olores reflejan entornos con baja comunicación, colaboración limitada y estructuras poco integradas, que afectan la participación y efectividad del desarrollador dentro del equipo.	A16, A41
14	Abandonador maloliente (AM)	Hace referencia a aquellos desarrolladores con presencia de OC que participaron en etapas previas del proyecto y luego abandonaron la comunidad. Su salida puede dejar vacíos de conocimiento y afectar la continuidad del desarrollo.	A16
15	Eslabón perdido (EP)	Ocurre cuando los colaboradores realizan modificaciones al código fuente de forma aislada, sin comunicarse con sus compañeros. Esta práctica genera inconsistencias, retrabajo y dificulta la integración de los cambios.	A1, A26, A32, A41
16	Cognición de clase (CC)	Se refiere a situaciones donde la refactorización del código complica su comprensión, tanto para los desarrolladores actuales como para los recién incorporados, quienes deben invertir tiempo adicional en entender las nuevas estructuras y ubicaciones.	A18, A26
17	Código rojo (CR)	Este olor se refiere a clases tan complejas que solo uno o dos desarrolladores pueden comprenderlas o mantenerlas, generando dependencia crítica y riesgo técnico.	A26

18	Distancia cognitiva (DC)	Ocurre cuando los miembros del equipo sienten barreras físicas, técnicas, sociales o culturales respecto a sus compañeros. Esta percepción dificulta la colaboración y puede generar conflictos o problemas de coordinación.	A14, A20, A26, A36, A38
19	Desarrollo de libros de cocina (DLC)	Se presenta cuando los desarrolladores se aferran a enfoques tradicionales y rechazan adoptar nuevas metodologías como las ágiles, lo que limita la innovación y genera productos que no cumplen con las expectativas del cliente.	A20, A26
20	Choque DevOps (CHDO)	Describe enfrentamientos entre equipos distribuidos de desarrollo y operaciones, especialmente cuando están sujetos a obligaciones contractuales distintas. Esto ralentiza el desarrollo y afecta la eficiencia operativa.	A20, A26
21	Desenganche (DES)	Sucede cuando los desarrolladores consideran que el software está listo y lo entregan a operaciones, aunque no esté completo. Esta falta de validación genera errores y omisiones en las funcionalidades.	A20, A26
22	Dispersión (DISP)	"... una solución o refactorización que provoca la fragmentación de un grupo existente, trabajando o siendo parte de una red de colaboración. Los reordenamientos de funcionalidad también conducen a un trabajo desordenado. Finalmente, coordinar y llevar a cabo actividades de mantenimiento es más desafiante".	A26
23	Disenso (DISE)	Se presenta cuando el equipo no logra acuerdos sobre cómo resolver ciertos problemas, lo que provoca que los olores de código identificados permanezcan sin corregirse.	A18, A26
24	Hipercomunidad (HC)	En comunidades muy cohesionadas, puede surgir una influencia excesiva de opiniones colectivas que anula el pensamiento crítico. Esto puede extenderse a subcomunidades y afectar la calidad del software.	A20, A26
25	Exceso de informalidad (EI)	La ausencia de normas claras para gestionar la información lleva a una sobrecarga de datos, desorganización y falta de responsabilidad entre los miembros de la comunidad.	A20, A26
26	Isomorfismo institucional (ISOI)	Se refiere a la similitud entre procesos o estructuras de distintas subcomunidades, ya sea por imitación o por operar bajo las mismas restricciones. Esta condición puede generar estancamiento, falta de innovación y dificultades de comunicación.	A26, A38

27	Arquitectura invisible (ARI)	Ocurre cuando las decisiones arquitectónicas y los acuerdos de reuniones no se registran de manera clara ni uniforme. Esto provoca desconocimiento dentro del equipo, errores de implementación y conflictos entre compañeros.	A26
28	Arquitectura solitaria (ARSO)	Se presenta cuando desarrolladores sin formación en arquitectura asumen decisiones críticas debido a la escasa disponibilidad de arquitectos. Esto genera incompatibilidades técnicas y falta de alineación en el desarrollo.	A20, A26
29	Aprovechamiento de novatos (APNO)	Los recién llegados no reciben orientación adecuada y deben averiguar por sí mismos qué hacer y para quién trabajar. Esto genera presión adicional, desinterés por parte de empleados antiguos y afecta el bienestar psicológico del equipo.	A20, A26
30	Arquitectura ofuscada (AROFU)	Ocurre cuando múltiples subcomunidades carecen de una visión organizacional y técnica compartida. Esto dificulta la integración armónica de nuevos desarrolladores, especialmente en proyectos que combinan sistemas heredados y modernos.	A26
31	Distancia de poder (DISPO)	Se refiere a cómo los miembros con menor jerarquía perciben y aceptan el comportamiento, las decisiones y la asignación de tareas de los líderes. Esta percepción puede afectar la cohesión, la motivación y la confianza dentro del equipo.	A6, A20, A25, A26
32	Miembros mojigatos (MIMO)	Algunos miembros actúan con una cautela extrema y apego estricto a normas éticas o técnicas, lo que dificulta la toma de decisiones arriesgadas. Esto reduce la innovación, limita la adaptabilidad del equipo y afecta el progreso del proyecto.	A20, A26
33	Desaprendizaje (DESAJ)	La diversidad de experiencia entre miembros dificulta el aprovechamiento de nuevos contenidos de formación. El conocimiento actualizado corre el riesgo de perderse cuando no se transmite adecuadamente a compañeros con mayor antigüedad.	A20, A26
34	Túnel del tiempo (TUTI)	Tras cambios estructurales, los miembros del equipo asumen equivocadamente que la comunicación será más fluida, sin prever la necesidad de mecanismos adicionales de coordinación explícita.	A20, A26
35	Sobras de aficionado a la tecnología (SAT)	Cuando no existe una red colaborativa entre desarrolladores y personal técnico (como mesa de ayuda, operaciones o mantenimiento), se impide el conocimiento compartido y se fragmenta el proceso de desarrollo.	A26