

UNIVERSIDAD DE VALLADOLID
MÁSTER UNIVERSITARIO
Ingeniería Informática



TRABAJO FIN DE MÁSTER

**Priorización de la deuda técnica alineada
con el negocio: un caso de estudio
industrial**

Realizado por **Antonio Camilo Martínez González**



Universidad de Valladolid

20 de septiembre de 2024

Tutor: Yania Crespo González-Carvajal

Universidad de Valladolid



Máster universitario en Ingeniería Informática

D. Yania Crespo González-Carvajal, profesora del departamento de Informática, área de Lenguajes y Sistemas Informáticos.

Expone:

Que el alumno D. Antonio Camilo Martínez González, ha realizado el Trabajo final de Máster en Ingeniería Informática titulado "PRIORIZACIÓN DE LA DEUDA TÉCNICA ALINEADA CON EL NEGOCIO: UN CASO DE ESTUDIO INDUSTRIAL".

Y que dicho trabajo ha sido realizado por el alumno bajo la dirección de quien suscribe, en virtud de lo cual se autoriza su presentación y defensa.

En Valladolid, 20 de septiembre de 2024

Vº. Bº. del Tutor:

D. Yania Crespo González-Carvajal

D. Fernando Redondo Sánchez, CEO de la empresa Anfix.

Expone:

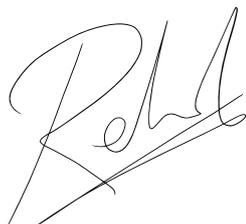
Que el trabajador de Anfix D. Antonio Camilo Martínez González, ha realizado el Trabajo final de Máster en Ingeniería Informática titulado “PRIORIZACIÓN DE LA DEUDA TÉCNICA ALINEADA CON EL NEGOCIO: UN CASO DE ESTUDIO INDUSTRIAL”.

El presente trabajo ha sido realizado por el trabajador, con la colaboración del firmante y otros compañeros de trabajo. Conociendo el contenido del trabajo y lo escrito en la memoria, doy el visto bueno desde el punto de vista de la empresa Anfix para su presentación, defensa y publicación como Trabajo de Fin de Máster del Máster en Ingeniería Informática de la Universidad de Valladolid.

En Valladolid, 20 de septiembre de 2024

CEO Anfix:

D. Fernando Redondo Sánchez

A handwritten signature in black ink, appearing to be 'FRS', written over a horizontal line.

Agradecimientos

Quiero expresar mi más sincero agradecimiento a mi tutora, Yania Crespo González-Carvajal, por su incondicional apoyo a lo largo de la realización de este trabajo. Su interés, dedicación y paciencia fueron fundamentales para poder culminar con éxito este proyecto.

A mis compañeros de trabajo, quienes me brindaron su ayuda y colaboración en el desarrollo de este estudio de caso, mi más profundo agradecimiento. En especial a Fer y Miguel, quienes a lo largo de los años me han ayudado a mejorar profesionalmente y a adquirir nuevos conocimientos. Gracias por la confianza depositada en mí y por su infinita paciencia.

A mi madre, aunque ahora no podamos estar juntos, sé que estarías muy orgullosa de este logro. Espero que este triunfo te traiga alegría y que pronto podamos compartirlo juntos.

A mi pareja, Angelli, por tu constante apoyo a lo largo de estos años. Has sido una fuente de inspiración y motivación para que pudiera culminar este reto personal. Tu compañía y aliento han sido fundamentales en cada paso de este camino.

No quiero olvidar a mis familiares, en especial a mi tío Titi, quien de una manera u otra siempre ha estado presente, apoyándome y ayudándome a perseguir mis sueños.

Finalmente, a todas las personas y amigos que, a lo largo de esta investigación, aportaron su granito de arena para que pudiera seguir adelante y culminar esta etapa de crecimiento personal y profesional.

A todos, gracias de corazón.

Resumen

Este Trabajo de Fin de Máster se enfoca en la aplicación de una metodología para la **priorización** de la **deuda técnica (DT)** en la empresa Anfix, alineando las decisiones técnicas con las estrategias comerciales de la empresa. La implementación de esta metodología no solo mejoró la eficiencia operativa, sino que también fortaleció la colaboración entre los equipos técnicos y de negocio, promoviendo una toma de decisiones más consensuada y estratégica. La revisión exhaustiva de la literatura permitió identificar una evolución en las metodologías de priorización de DT, desde enfoques puramente técnicos hacia modelos integrados que consideran aspectos de negocio. En este contexto, se seleccionó el **Framework Tracy** por su capacidad para equilibrar las perspectivas técnica y comercial. Este marco facilitó la identificación y priorización de las áreas críticas de DT en función de su impacto en los activos de TI y en los procesos esenciales del negocio. La participación activa de los **stakeholders** fue crucial en la definición de métricas claves (como el impacto en la experiencia del cliente, costos operativos, valor comercial y escalabilidad del sistema), asegurando que la priorización reflejara tanto las necesidades técnicas como los objetivos comerciales. Los resultados evidenciaron una mejora significativa en la eficiencia operativa y en la capacidad de Anfix para innovar sin comprometer la estabilidad del sistema. En conclusión, la adecuada priorización de la DT es esencial para mantener la competitividad y sostenibilidad en un entorno tecnológico dinámico. La experiencia de Anfix demuestra que integrar las decisiones técnicas con las estrategias de negocio, apoyado en una metodología sólida y la **colaboración** entre stakeholders, es fundamental para gestionar la DT de forma proactiva y evitar su acumulación futura.

Descriptores

Deuda técnica, priorización, Framework Tracy, estudio de caso, colaboración interdisciplinaria.

Abstract

This Master's thesis focuses on the application of a methodology for prioritizing **technical debt (TD)** at Anfix company, aligning technical decisions with the company's business strategies. The implementation of this methodology not only improved operational efficiency but also strengthened collaboration between technical and business teams, promoting more consensual and strategic decision-making. An exhaustive review of the literature identified an evolution in TD **prioritization** methodologies, shifting from purely technical approaches to integrated models that consider business aspects. In this context, the **Tracy Framework** was selected for its ability to balance technical and commercial perspectives. This framework facilitated the identification and prioritization of critical areas of TD based on their impact on IT assets and essential business processes. Active **stakeholder** participation was crucial in defining key metrics (such as impact on customer experience, operational costs, commercial value, and system scalability), ensuring that the prioritization reflected both technical needs and commercial objectives. The results showed a significant improvement in operational efficiency and in Anfix's ability to innovate without compromising system stability. In conclusion, proper prioritization of TD is essential for maintaining competitiveness and sustainability in a dynamic technological environment. Anfix's experience demonstrates that integrating technical decisions with business strategies, supported by a solid methodology and **collaboration** among stakeholders, is fundamental for proactively managing TD and avoiding its future accumulation.

Keywords

Technical debt, prioritization, Tracy Framework, case study, interdisciplinary collaboration.

Índice general

Índice general	III
Índice de figuras	V
Índice de tablas	VI
1 Introducción	1
1.1. Contexto y Motivación	2
1.2. Objetivos	3
1.3. Preguntas de investigación	3
1.4. Estructura del documento	4
2 Marco teórico	6
2.1. Calidad del Software y Deuda Técnica	6
2.2. Gestión de la DT	8
2.3. Priorización de la DT	14
3 Estado del arte	16
3.1. Estrategia para la revisión de la literatura	16
3.2. Selección de una metodología para la priorización de la DT enfocada en el negocio	23
3.3. Descripción detallada del Framework Tracy	30
4 Caso de estudio	41
4.1. Introducción al caso de estudio	41
4.2. Descripción del caso	44
4.3. Metodología del caso de estudio	44
4.4. Análisis del caso	45
4.5. Ejecución del caso de estudio	45
4.6. Priorización de la DT	62

5	Resultados y Discusión	67
5.1.	Identificación y clasificación de la DT	67
5.2.	Priorización de la DT	68
5.3.	Impacto en los Procesos de Negocio y Activos de TI	71
5.4.	Discusión	72
6	Conclusiones y trabajos futuros	76
6.1.	Conclusiones generales del trabajo	76
6.2.	Revisión de la literatura	76
6.3.	Análisis y comparación de metodologías	77
6.4.	Resultados obtenidos en Anfix	77
6.5.	Reflexiones finales	78
6.6.	Trabajo futuro	78
	Bibliografía	80
	Apéndices	83
	Apéndice A Documentación de la deuda técnica identificada	85

Índice de figuras

2.1. Cuadrante sobre la DT planteado por Martin Fowler [22]	9
3.1. Clasificación de documentos por base de datos	20
3.2. Clasificación de documentos por año de publicación	21
3.3. Clasificación de documentos por país de origen	22
3.4. Clasificación de documentos por tipo	23
3.5. Framework Tracy [8]	31
3.6. Plantilla de relación entre los diferentes artefactos propuestos por el framework Tracy [10]	38
3.7. Plantilla de trabajo propuesta para el Framework Tracy [10]	39
3.8. Tabla para relacionar a partir de las métricas los Activos de TI y los Procesos de Negocio [10]	40
4.1. Alternativas para diseño de unidades y contexto de análisis en estudios de casos	43
4.2. Clasificación de la DT	48
4.3. Diagrama de burbujas para representar visualmente los datos de la matriz cruzada (Tabla 4.3)	52
4.4. Relación entre los procesos de negocio, activos de TI y los elementos de configuración identificados en Anfix	55
4.5. Plantilla de priorización de la DT en los procesos de Ventas, Producción, Finanzas e Infraestructura	57
4.6. Plantilla de priorización de la DT en los procesos de Producción, Soporte, Marketing Recursos Humanos e Infraestructura	58
4.7. Plantilla de valor empresarial con los datos obtenidos de Anfix	61

Índice de tablas

3.1. Número de documentos encontrados en diferentes fuentes para las cadenas de búsquedas “Technical Debt Prioritization” y “Technical Debt Prioritization Methodology”	18
3.2. Caracterización de los trabajos revisados sobre priorización de la DT.	26
3.3. Comparación de metodologías y técnicas para la priorización de la DT según varios criterios.	28
4.1. Clasificación de los tipos de DT y la cantidad de ocurrencias	47
4.2. Relación de elementos de configuración y números de DT	50
4.3. Matriz de datos cruzados entre tipos de DT y elementos de configuración . . .	51
4.4. Prioridad de la DT basada en la relación de los tipos de procesos de negocio y los estados de los activos de TI	59
4.5. Partida de DT seleccionadas para su priorización con respecto a su impacto potencial en el negocio (Grupo 1)	64
4.6. Partida de DT seleccionadas para su priorización con respecto a su impacto potencial en el negocio (Grupo 2)	64
4.7. Partida de DT seleccionadas para su priorización con respecto a su impacto potencial en el negocio (Grupo 3)	65
4.8. Partida de DT seleccionadas para su priorización con respecto a su impacto potencial en el negocio (Grupo 4)	65
A.1. Listado de incidencias registradas	100

Capítulo 1

Introducción

En el año 1992 Ward Cunningham [20] utilizó el término “Deuda Técnica (DT)” como una metáfora para explicar cómo algunas prácticas empleadas en el proceso de desarrollo de software podían requerir de un esfuerzo adicional, como si se tratase de pago de intereses. Es decir, que la DT se refiere al coste en el que incurre un proyecto de software cuando los desarrolladores toman atajos, hacen arreglos rápidos o retrasan las actualizaciones necesarias para cumplir con los plazos o las fechas de lanzamiento. Aunque puede ser útil a corto plazo para ciertos propósitos de los desarrolladores o el negocio, al igual que la deuda financiera, la DT incurre en intereses a lo largo del tiempo, lo que hace que su pago sea más costoso cuanto más tiempo se deje sin resolver [29] [6].

Desde su origen, el término DT ha evolucionado para incluir una variedad de tipos, que pueden ser causados por diferentes factores. Algunas causas comunes de la DT incluyen el uso de soluciones temporales para cumplir con plazos de entrega ajustados, la falta de adopción de buenas prácticas de desarrollo de software, la falta de mantenimiento adecuado del código, y la falta de atención a la calidad del código [46].

A lo largo de los años, la DT se ha ido convirtiendo en un tema importante en el mundo del desarrollo de software, ya que ha sido reconocido como un problema común que puede tener consecuencias graves. Algunas de las consecuencias de la DT incluyen la necesidad de realizar actualizaciones de software costosas y la disminución de la calidad del software, lo que puede afectar la satisfacción del cliente. Hoy en día, el concepto de DT es ampliamente aceptado en la comunidad de desarrollo de software y existen muchas herramientas y técnicas para ayudar a los desarrolladores a administrar y mitigar sus efectos, entre las que se incluyen revisiones de código, pruebas automatizadas, refactorización y otras mejores prácticas diseñadas para promover el desarrollo de software escalable y mantenible [47]. Además, se ha podido notar que en los últimos años existe una tendencia ascendente (muy considerable) con respecto a las investigaciones sobre los diferentes tipos de DT, la gestión de esta, las actividades relacionadas, los enfoques y las herramientas que se pueden utilizar para su detección, medición, gestión y subsanación.

En el entorno empresarial actual, los equipos de desarrollo de software se encuentran bajo una presión constante para ofrecer productos de software de alta calidad dentro de plazos y presupuestos ajustados. Esto a menudo da como resultado que se acumule DT, ya que los equipos se ven obligados a realizar concesiones y tomar atajos para cumplir con estas limitaciones. Dentro de las posibles consecuencias de acumulación de la DT se destaca un sistema incontrolable e inflexible, incapacidad de incluir nuevas características sin poner en riesgo las existentes, introducción de nuevas características con fallos, entre otras [1].

Los desarrolladores de software suelen dedicar una cantidad significativa de tiempo a resolver problemas presentes en el código en lugar de centrarse en nuevas funcionalidades [2]. En la encuesta realizada por Stack Overflow en el año 2022 a más de 70.000 desarrolladores en todo el mundo, se encontró que el 63 % de los encuestados invertía entre 30 a 60 minutos al día buscando soluciones a problemas que se presentan debido a la DT, mientras que el 25 % más de 60 minutos. Para equipos de 50 desarrolladores la cantidad de tiempo que dedican respondiendo preguntas oscila entre 333 y 651 horas por semana [39].

La priorización de la DT es el proceso en el cual se decide qué elementos de la DT pueden reembolsarse primero y cuáles deberían posponerse. En muchos casos, las empresas prefieren gastar los recursos disponibles incorporando nuevas funciones que corregir códigos de mala calidad. Si los recursos son limitados, pueden tener dificultades para decidir sobre cuáles elementos tienen un mayor valor y cómo tener un equilibrio en su coste [3]. La toma de estas decisiones es crucial para equilibrar el desarrollo rápido con la mantenibilidad futura del software [3]. De esta manera, la priorización de la DT debe basarse no solo en la gravedad de los problemas, sino también en factores como la frecuencia con la que el código afectado es modificado y el riesgo que representa para la estabilidad general del sistema. Este enfoque permite a los equipos de desarrollo enfocarse en las áreas críticas, minimizando el impacto negativo a largo plazo [5].

Partiendo de lo anterior, el problema que se pretende exponer en este trabajo es cómo abordar la priorización de la DT en una organización, específicamente en la empresa de desarrollo de software de facturación y contabilidad Anfix [11], teniendo en cuenta fundamentalmente los criterios que se encuentren más alineados con el negocio. Para ello, en el Capítulo 3, se realizará un análisis de las diferentes metodologías existentes de priorización de la DT que maximicen en alguna medida los beneficios del negocio. De las diferentes metodologías revisadas, se tomarán los criterios que mejor se adapten en Anfix y se aplicarán en dicha organización con el fin de evaluar los resultados obtenidos para comprobar si se necesitan ajustar o complementar los criterios elegidos hasta resolver los problemas del negocio a través de la priorización de la DT.

1.1. Contexto y Motivación

Este trabajo se realiza bajo el marco del Trabajo de Fin de Máster (TFM), asociado al Máster en Ingeniería Informática de la Universidad de Valladolid, pero tomando como

referencia una empresa real para la aplicación de la metodología seleccionada y el posterior análisis de los resultados.

Como se mencionó anteriormente en este capítulo, se presentará el caso de Anfix, una empresa reconocida en el sector de la tecnología y los servicios empresariales. Esta empresa se dedica a proporcionar soluciones de contabilidad y gestión empresarial a sus clientes, y ha experimentado un crecimiento significativo en los últimos años. Sin embargo, como muchas otras empresas del sector, Anfix ha acumulado una cierta cantidad de DT debido a diversos factores como algunas decisiones de carácter comercial, la necesidad de lanzar rápidamente nuevas funcionalidades y características en su software para mantenerse competitiva en el mercado, la migración hacia arquitecturas basadas en microservicios, entre otras.

La empresa reconoce la importancia de abordar esta problemática con el fin de mantener la calidad y la eficiencia de su producto, así como evitar riesgos a largo plazo. Por lo tanto, existe un claro interés por parte de Anfix en aplicar las soluciones y enfoques propuestos en este TFM para priorizar y gestionar de manera efectiva su DT, con el objetivo de maximizar los beneficios para el negocio y garantizar la satisfacción de sus clientes.

1.2. Objetivos

Luego de haber expuesto el planteamiento, el contexto y la motivación se expone el objetivo principal de este TFM:

- Aplicar una metodología para la priorización de la DT en la empresa Anfix, alineando los intereses técnicos y del negocio con el fin último de ayudar a reducir la DT.

Con el fin de desglosar el objetivo principal, se definen los siguientes subobjetivos:

- Realizar una revisión de la literatura disponible en las fuentes seleccionadas para recopilar información sobre las metodologías de priorización de la DT.
- Analizar las diferentes metodologías aplicadas en un contexto empresarial y evaluar cómo se puede aplicar de forma práctica y efectiva en el contexto concreto de la empresa Anfix.
- Exponer los resultados obtenidos tras la aplicación de la metodología seleccionada para la empresa Anfix.

1.3. Preguntas de investigación

En este sentido, para esclarecer más sobre este tema se han identificado algunas preguntas de investigación:

- ¿Existen diferentes metodologías de priorización específicas para cada tipo de DT? ¿Se basan en el contexto/negocio de la empresa? ¿o ambas?
- ¿Cuáles son las metodologías de priorización de la DT aplicadas en un contexto empresarial y cómo se han evaluado en la literatura?
- ¿Que indicadores se han utilizado para medir el éxito de la implantación de una metodología de priorización de DT?
- ¿Es posible adaptar y aplicar una de estas metodologías de priorización de la DT en función de un contexto/proyecto/negocio?

1.4. Estructura del documento

El presente documento describe el proceso de investigación y estudio llevado a cabo para adaptar y aplicar una de las metodologías existentes en la priorización de la DT en un contexto empresarial.

Este primer capítulo incluye una breve introducción al tema, a través de su planteamiento, contexto y motivación; también se plantea el objetivo general, los objetivos específicos, las preguntas de investigación y se describe la estructura del documento.

En el segundo capítulo 2 (Marco teórico), se detallan los conceptos más relevantes sobre los temas de interés a tratar, entre los cuales se destaca la calidad de software, la DT, los tipos de DT, y metodologías para la priorización de la DT.

El tercer capítulo 3 (Estado del arte), examina la situación actual en términos de conocimiento y estudios relacionados con el tema (DT), así como el estado del arte actual de la priorización de la DT. Esto se realiza mediante una búsqueda sistemática de fuentes académicas y literatura gris. Además, se expone la metodología de priorización de la DT seleccionada, presentando las características fundamentales y los contextos de aplicación.

El cuarto capítulo 4 (Caso de estudio) expone el diseño y la ejecución del estudio de caso seleccionado para esta investigación. Se describe el contexto en el que se desarrolla el caso, incluyendo información sobre la organización. Posteriormente, se presenta un análisis exhaustivo de los datos recopilados, identificando los principales elementos de DT presentes en el proyecto y su priorización tanto desde una perspectiva técnica como empresarial. Este análisis incluye gráficos, tablas y diagramas que facilitan la visualización de los resultados obtenidos.

En el quinto capítulo 5 (Resultados y discusión) se exponen los resultados de la aplicación de la metodología seleccionada y los ajustes realizados para su correcta implementación.

En el sexto capítulo capítulo 6 (Conclusiones) se exponen las conclusiones de los resultados al aplicar la metodología de priorización de DT en Anfix teniendo en cuenta todo el procedimiento para lograr dichos resultados.

Finalmente se muestra un Anexo [A](#) donde se presentará la documentación de la DT identificada, eliminando los datos que son confidenciales de la empresa.

Marco teórico

2.1. Calidad del Software y Deuda Técnica

La calidad del software se refiere a la capacidad de este software para cumplir con los requisitos, expectativas y necesidades del usuario, así como con los estándares de calidad establecidos. Según los modelos de calidad, se definen diversas características y subcaracterísticas para evaluar el software mediante indicadores y métricas.

Existen varios estándares y marcos de calidad de software que se utilizan para evaluar y guiar las prácticas de desarrollo. Uno de los más conocidos es el estándar ISO 25010, que establece que la calidad del producto de software debe considerar la adecuación funcional, eficiencia de desempeño, compatibilidad, usabilidad, fiabilidad, seguridad, mantenibilidad y portabilidad [24].

Como se ha comentado en el Capítulo 1, la DT es un concepto que describe el coste a largo plazo derivado de decisiones rápidas y subóptimas en el desarrollo de software, como atajos en diseño, implementación, pruebas y documentación. Estas decisiones pueden ser necesarias para cumplir con plazos o reducir costes a corto plazo, pero generan un código más difícil de mantener y mejorar en el futuro.

La DT se puede identificar mediante la observación de varias señales, como código desordenado, falta de pruebas adecuadas, documentación insuficiente y problemas de escalabilidad, etc.. Si no se aborda adecuadamente, la DT se acumula, aumentando el riesgo de errores, retrasos en el desarrollo y mayores costes de mantenimiento a lo largo del tiempo. En esencia, la DT actúa como un obstáculo para la calidad del software, afectando negativamente su sostenibilidad y eficiencia futura.

A medida que la tecnología ha avanzado y el desarrollo de software se ha vuelto más complejo, la DT se ha convertido en un desafío cada vez mayor para los equipos de desarrollo. Esto se debe a la necesidad constante de innovación por parte de las empresas para impulsar su crecimiento y competitividad. Afortunadamente, también ha habido una mayor conciencia sobre la DT en la industria del software, lo que ha llevado a un mayor enfoque en la calidad del código y a la adopción de mejores prácticas de desarrollo [35].

Tipos de DT

Varios estudios[29, 30], han documentado los diferentes tipos de DT que se pueden identificar en el software. A continuación, se presenta una descripción detallada de los principales tipos de DT:

- **Deuda de requisitos (Requirements debt):** Se refiere a la brecha entre la especificación óptima de requisitos y la implementación real del sistema, bajo condiciones y restricciones del dominio. Esta deuda surge cuando los requisitos iniciales no se especifican adecuadamente o se modifican significativamente durante el desarrollo, lo que lleva a inconsistencias y funcionalidades no óptimas [28].
- **Deuda arquitectónica (Architectural debt):** Es causada por decisiones de arquitectura que comprometen algunos aspectos de la calidad interna, como la capacidad de mantenimiento. Esto puede incluir decisiones apresuradas sobre la estructura del sistema o la falta de una arquitectura escalable, lo que puede dificultar las futuras modificaciones y la adición de nuevas funcionalidades [20].
- **Deuda de diseño (Design debt):** Se refiere a los atajos técnicos que se toman en el diseño detallado de la aplicación. Esta deuda se manifiesta en patrones de diseño deficientes o complejos que pueden hacer que el código sea difícil de entender y mantener [14].
- **Deuda de código (Code debt):** Es el código mal escrito que viola las mejores prácticas de codificación o las reglas de codificación como, por ejemplo, la duplicación de código, la falta de comentarios, y el código excesivamente complejo, lo que dificulta su mantenimiento y comprensión [31].
- **Deuda de pruebas (Test debt):** Se refiere a los atajos que se toman en las pruebas realizadas al software. Esto incluye la falta de pruebas unitarias, de integración y de aceptación, lo que puede llevar a la introducción de errores no detectados en el sistema [32].
- **Deuda de construcción (Build debt):** Se refiere a los fallos en un sistema de software, en su sistema de compilación o en su proceso de compilación, que hacen que la compilación sea demasiado compleja y difícil. Un proceso de compilación ineficiente puede ralentizar el desarrollo y aumentar los errores [12].
- **Deuda de documentación (Documentation debt):** Se refiere a la documentación insuficiente, incompleta o desactualizada en cualquier aspecto del desarrollo de software como, por ejemplo, documentación de arquitectura desactualizada, ausencia de documentación de una API, lo que puede dificultar el trabajo de los desarrolladores y la comprensión del sistema [23].
- **Deuda de infraestructura (Infrastructure debt):** Se refiere a una configuración subóptima de procesos relacionados con el desarrollo, tecnologías, herramientas de

soporte, etc. Tal configuración subóptima afecta negativamente la capacidad del equipo para producir un producto de calidad. Incluye problemas con el entorno de desarrollo y la falta de automatización en procesos clave [26].

- **Deuda de versionado (Versioning debt):** Se refiere a los problemas en el control de versiones del código fuente, como bifurcaciones de código innecesarias o cambios de versión de software antiguos a nuevos. Esto puede llevar a conflictos de integración y pérdida de código importante [13].
- **Deuda de defectos (Defect debt):** Se refiere a los defectos, errores o fallos que se encuentran en los sistemas de software. La acumulación de defectos no corregidos puede llevar a un deterioro significativo de la calidad del software [36].

Cuadrante de DT de Martin Fowler

En 2009, Martin Fowler [22] publicó un influyente artículo sobre los matices de la DT y su papel en el desarrollo de software señalando que la metáfora de la deuda es poderosa para comunicar tanto a partes técnicas como no técnicas, añadiendo una distinción entre deuda imprudente y prudente, y entre deliberada e inadvertida, en forma de cuadrante, lo que le otorga a la metáfora más profundidad y precisión. [44].

En la Figura 2.1, se observa que la deuda prudente y deliberada se asume con conocimiento de sus beneficios a largo plazo, mientras que la deuda imprudente e inadvertida resulta no solo de ignorar buenas prácticas sino de no considerarlas importantes/necesarias. También se puede tener deuda prudente inadvertida, que ocurre cuando se realiza un software valioso con un código limpio, pero posteriormente se reconoce que el diseño debía ser diferente. Esta situación es común porque incluso los mejores desarrolladores aprenden durante el proceso de programación.

El cuadrante de Fowler es crucial para comprender que la decisión de abordar o posponer una DT debe tomarse de manera informada. Además, es fundamental reconocer que la DT es inevitable en el desarrollo de software, incluso para los equipos más experimentados. Siendo conscientes del cuadrante de Fowler, los equipos pueden gestionar esta deuda de manera más efectiva, asegurando así la mejora continua de la calidad del software.

2.2. Gestión de la DT

El proceso de gestión de la DT se puede realizar desde diferentes perspectivas de acuerdo con diversos estudios realizados, por lo tanto a continuación se expone esta gestión teniendo en cuenta las actividades, enfoques, y herramientas [3].

Actividades para la gestión de la DT

Los principales autores en el campo de la gestión de la DT han identificado varias actividades críticas para manejar eficazmente la DT en proyectos de software. Estas



Figura 2.1: Cuadrante sobre la DT planteado por Martin Fowler [22]

actividades aseguran que la DT se identifique, mida, priorice y gestione de manera sistemática para minimizar su impacto negativo en el proyecto a largo plazo [3].

- **Identificación de la DT:** La primera actividad en la gestión de la DT es identificar los componentes del sistema que contienen DT. Esto incluye detectar código de baja calidad, diseños subóptimos, decisiones arquitectónicas que comprometen la calidad del sistema, etc.. Esta permite a los equipos ser conscientes de los problemas técnicos existentes y a su vez, facilita la planificación para abordar estos problemas antes de que se conviertan en riesgos mayores.
- **Medición de la DT:** Esta actividad implica cuantificar la DT en términos de esfuerzo necesario para remediarla, su impacto en el sistema y el riesgo asociado. Herramientas como SonarQube [38] y métricas de código son comúnmente usadas para esta tarea. La medición debe proporcionar datos objetivos para priorizar el trabajo de mitigación/subsanación. Por otra parte, ayuda a justificar la necesidad de recursos para abordar la DT.
- **Priorización de la DT:** Consiste en clasificar los elementos de DT según su impacto en los objetivos comerciales y técnicos del proyecto. Esto puede incluir factores como la criticidad del componente afectado, la frecuencia de cambios y el riesgo de fallos. Esta asegura que los recursos se dirijan a los problemas que tienen el mayor impacto en el negocio y el sistema. Además, facilita la toma de decisiones informadas sobre qué DT abordar primero.

- Planificación y mitigación/subsanación de la DT: Desarrollar un plan de acción para abordar la DT priorizada. Esto incluye asignar tareas específicas a los miembros del equipo, establecer plazos y asegurar que las actividades de mitigación/subsanación se integren en el ciclo de desarrollo. Esta asegura que la DT se aborde de manera oportuna y eficiente. También, reduce el riesgo de que la DT continúe acumulándose y afecte la calidad del software.
- Monitorizar y controlar la DT: Implica el seguimiento continuo del estado de la DT y la efectividad de las actividades de mitigación/subsanación. Utilizar herramientas de gestión de proyectos como JIRA para registrar y monitorizar el progreso de las mismas. Esta permite ajustar las estrategias de gestión según sea necesario. Ayuda a mantener a los equipos responsables y enfocados en la reducción de la DT.
- Prevención de la DT: Implementar prácticas y políticas para prevenir la acumulación futura de DT. Esto puede incluir la adopción de prácticas de desarrollo ágil, revisiones de código regulares y educación continua del equipo sobre la importancia de la calidad del código. Esta minimiza la ocurrencia de DT. Promueve una cultura de calidad y sostenibilidad en el desarrollo de software.
- Documentación de la DT: Documentar la DT existente, incluyendo detalles sobre la naturaleza del problema, su impacto potencial y las decisiones tomadas para su mitigación. Esta facilita la comunicación entre los miembros del equipo y otros interesados. Proporciona una referencia clara y accesible para futuras evaluaciones y acciones.
- Comunicación de la DT: Asegurar que todos los interesados comprendan la DT y sus implicaciones a través de una comunicación clara y constante. Esto puede incluir reuniones regulares, informes de estado y documentación accesible. Esta mejora la transparencia y colaboración. Asegura que las decisiones sobre la DT sean comprendidas y apoyadas por todos los interesados.

Enfoques de gestión de la DT

Diversos estudios muestran enfoques de gestión de la DT con base en las actividades que se requieren para este proceso. Estos se presentan de acuerdo con categorías. Es importante mencionar que se ha tomado como referencia principal la revisión sistemática realizada por Alfayez et al. [3], y se complementará con otras referencias para sostener la variedad de estudios.

Los enfoques que se basan en la identificación se centran en el análisis del código fuente para evidenciar posibles infracciones. Posteriormente, se realizan cálculos de métricas basadas en este mismo código para identificar problemas de diseño o arquitectura; también se pueden analizar dependencias de diferentes tipos de elementos del software; se realizan listas de comprobación respecto a escenarios de incoherencia de la DT; y adicionalmente, se realiza la comparación de soluciones respecto a la relación coste-beneficio. Por ejemplo,

en [45] se exploran métodos de análisis estático de código, en [37] se detallan métricas de calidad, y en [41] se discute sobre listas de comprobación.

Por otra parte, los enfoques basados en la medición se centran en calcular la DT mediante fórmulas o modelos matemáticos; utilizan métricas de código fuente; realizan estimaciones de la DT en relación con la experiencia y los conocimientos técnicos; estiman diferentes tipos de costes de la DT; aplican métricas operativas en función de la calidad del producto; y también realizan cálculos entre la solución real y la óptima. Por ejemplo, en [43] se presentan modelos matemáticos específicos, en [37] se discuten estimaciones basadas en experiencia, y en [21] se analizan los costes de la DT.

De igual forma, los enfoques que se refieren a la documentación señalan que esta debe tener: un identificador único de la DT, su ubicación, responsable del reembolso, tipo de DT, información general de la DT, fecha y hora en que se identifica la DT, principal coste estimado del reembolso, coste adicional estimado para tolerar la DT, probabilidad de interés en su reembolso, diferencia estimada respecto al valor de los intereses y al valor futuro de los intereses, contexto de implementación (ej. Lenguaje de programación), y reglas de propagación de la DT en relación con el sistema de software. Por ejemplo, en [45] se detalla un sistema de documentación eficiente.

Frente a los enfoques basados en el monitoreo se muestra que estos se enfocan en definir umbrales para las métricas de calidad relacionadas con la DT [41], rastrear la propagación de la DT en el sistema de software [37], realizar una medición periódica para identificar y hacer seguimiento del cambio de la DT [37], supervisar la DT en relación con su calidad y estabilidad, además de realizar medidas mediante gráficos para observar las tendencias de la DT a lo largo del tiempo.

Asimismo, los enfoques que se centran en el reembolso de la DT, o mitigación/subsanación, incluyen procesos de refactorización como cambios en el código [43], en la arquitectura o el diseño sin alterar el sistema de software y mejorar la calidad del sistema; reescribir el código que presenta la DT [21]; automatizar el trabajo que contiene repeticiones manuales; realizar reingeniería de software para mejorar sus características y calidad operativa [37]; y resolver los errores que se conocen.

Por otro lado, los enfoques relacionados con la comunicación establecen la creación de cuadros de mando para que todas las partes interesadas conozcan la existencia de la DT; crear un Backlog para registrar todos los elementos de la DT y estos puedan ser notificados y tratados respecto a su nivel de importancia; realizar una lista de DT incluyendo todos sus elementos; visualizar la propagación de la DT mostrando sus causas, conexiones y afectaciones. Por ejemplo, en [43] se habla sobre cuadros de mando y en [21] sobre visualización de la propagación.

Finalmente, los enfoques que se basan en la prevención permiten mejorar los procesos de desarrollo actuales evitando la presencia de DT; también se realiza apoyo a la toma de decisiones para evaluar la DT potencial; se realiza la planificación de costes teniendo en cuenta el ciclo de vida del sistema para minimizar la DT; y se realiza el análisis de factores humanos que, a causa de situaciones como la ignorancia, pueden causar DT de

manera no intencionada. Por ejemplo, en [43] se trata sobre la mejora de procesos, y en [37] sobre la planificación de costes.

Herramientas Asistenciales para la Gestión de la DT

Ante el creciente reconocimiento de la DT se han desarrollado herramientas diseñadas para facilitar su gestión. De esta manera, asisten a los equipos en la identificación y mitigación de este fenómeno, ayudando a mejorar tanto la calidad del código como la eficiencia del equipo de desarrollo.

Entre las herramientas más utilizadas en la industria se encuentran:

- **SonarQube:** es una plataforma líder en análisis estático de código fuente. Proporciona una evaluación continua de la calidad del código para detectar bugs potenciales, vulnerabilidades de seguridad y malas prácticas en el código. Funciona con más de 20 lenguajes de programación. Uno de los aspectos más valorados de SonarQube es su capacidad para integrarse con sistemas de control de versiones y entornos CI/CD (Integración Continua y Despliegue Continuo), lo que permite a los equipos de desarrollo detectar y corregir problemas en una etapa temprana del ciclo de desarrollo de software. La herramienta ofrece dashboards detallados que presentan métricas de calidad, lo que facilita la identificación de tendencias y la priorización de problemas basados en su severidad [38]. Además, se puede ampliar mediante plugins lo cual ha traído la ventaja de integrar múltiples herramientas en SonarQube como se menciona en los siguientes puntos.
- **JDepend:** es una herramienta diseñada específicamente para aplicaciones Java que se puede integrar en SonarQube a través de un plugin. Esta analiza paquetes Java para determinar la cohesión entre clases y paquetes. JDepend proporciona métricas que ayudan a los desarrolladores a entender la calidad del diseño de su software, lo que les permite identificar áreas problemáticas y reducir la complejidad a través de mejoras en el diseño. Es especialmente útil para proyectos grandes donde la gestión de dependencias es crucial para la mantenibilidad.
- **PMD:** es una herramienta de análisis estático de código fuente que también se integra en SonarQube mediante un plugin. Esta se utiliza para identificar problemas comunes como bugs potenciales, código muerto (sin usar), sobre-complejidad, y malas prácticas de codificación. PMD soporta varios lenguajes de programación y se centra en una rápida ejecución de análisis para que los desarrolladores puedan integrarlo en sus workflows sin mucha sobrecarga. La herramienta es altamente configurable y permite a los equipos definir sus propias reglas de codificación para asegurar que el código se adhiere a los estándares internos[33].
- **CodeClimate y CodeClimate Velocity:** CodeClimate es una herramienta de análisis de código que opera principalmente en la nube, diseñada para ayudar a las organizaciones a mejorar la calidad de su código. Esta herramienta realiza una

evaluación automatizada basada en configuraciones personalizables que permiten ajustar los análisis a los estándares específicos de cada empresa. CodeClimate desglosa los problemas del código por categoría y severidad, sugiriendo mejoras concretas que optimizan tanto la calidad como la mantenibilidad del código. Su capacidad para identificar áreas problemáticas facilita la implementación de soluciones que garantizan un código más eficiente y sostenible en el tiempo [17]. Además, CodeClimate Velocity es una extensión enfocada en los aspectos operativos del proceso de desarrollo de software. Mientras que CodeClimate se centra en la calidad del código, Velocity analiza el flujo de trabajo de los equipos de desarrollo, ayudando a identificar cuellos de botella y a generar insights sobre cómo los cambios en el código afectan el rendimiento general del equipo. Esta herramienta es particularmente útil para gerentes de proyectos y líderes técnicos que desean mejorar la eficiencia y la efectividad de sus equipos, optimizando la productividad y asegurando una mejor gestión del ciclo de desarrollo de software [18].

- **CAST Highlight:** Es una herramienta de análisis de software que proporciona insights rápidos y precisos sobre el portafolio de software de una organización. Es particularmente útil para grandes empresas que gestionan múltiples bases de código a gran escala. Highlight evalúa la DT, el riesgo de seguridad, la complejidad del código y la obsolescencia de la tecnología. Además, proporciona comparaciones, basadas en benchmarking, que permiten a las empresas entender cómo se posiciona su calidad de código respecto a la industria[15].
- **CodeScene:** es una herramienta de análisis de código que va más allá de las métricas tradicionales al combinar la evaluación técnica del código con un análisis socio-técnico que considera la dinámica del equipo de desarrollo. La herramienta identifica “hotspots” o áreas del código que experimentan cambios frecuentes, lo que sugiere posibles problemas de DT y mayor riesgo de errores. Además, CodeScene analiza la propiedad del código, es decir, quién ha contribuido más a cada parte, y detecta posibles conflictos cuando múltiples desarrolladores trabajan en las mismas áreas, lo que podría afectar la mantenibilidad del software. También proporciona métricas de productividad y colaboración, midiendo la fricción entre equipos y el impacto de las decisiones técnicas en los objetivos de negocio. Su integración en pipelines de DevOps permite realizar análisis automáticos y continuos, facilitando la detección temprana de problemas y la mejora constante del código. Las visualizaciones avanzadas y los informes detallados de CodeScene hacen que sea una herramienta para equipos de desarrollo que buscan optimizar tanto la calidad del código como la eficiencia del equipo, ofreciendo una visión integral y predictiva que apoya decisiones estratégicas en el desarrollo de software[19].
- **Kiuwan:** es una plataforma avanzada que se especializa en el análisis de la calidad y seguridad del código fuente, proporcionando herramientas para identificar y gestionar problemas de DT, vulnerabilidades de seguridad y cumplimiento normativo en el desarrollo de software. A través de su análisis estático, Kiuwan detecta problemas críticos en el código según estándares internacionales como OWASP y CWE, y ofrece

soluciones para prevenir riesgos de seguridad desde las primeras etapas del desarrollo. La plataforma también facilita el cumplimiento de normativas como PCI-DSS, GDPR, y HIPAA, asegurando que el código siga las regulaciones pertinentes. Kiuwan se integra fácilmente en el ciclo de vida del desarrollo del software (SDLC) y es compatible con herramientas de CI/CD como Jenkins y GitLab, permitiendo automatizar los análisis de calidad y seguridad en tiempo real. Su soporte multilingüaje abarca más de 30 tecnologías, lo que la convierte en una solución versátil para proyectos que involucran diferentes plataformas. Además, Kiuwan ofrece visualizaciones detalladas y reportes personalizables que ayudan a los equipos técnicos y gerenciales a comprender y actuar sobre los problemas detectados, facilitando una mejor colaboración y gestión de los equipos de desarrollo [27].

Además, varios Entornos de Desarrollo Integrados (IDE) modernos, como Microsoft Visual Studio y Eclipse para Java, incluyen funcionalidades para la refactorización de código, facilitando la reducción de la DT en el flujo de trabajo diario.

Cabe destacar que la prevención de la DT a menudo depende de la mejora continua del proceso de desarrollo de software. Prácticas como la integración continua, que exige una alta cobertura de pruebas unitarias y de integración, pueden prevenir significativamente la acumulación de DT, especialmente en lo que respecta a pruebas y calidad de código.

Aunque las herramientas disponibles se centran principalmente en la deuda de código, de diseño y de pruebas, existe una necesidad de herramientas que aborden otros tipos de DT, como la arquitectónica, que son más abstractos y menos estudiados. Idealmente, una herramienta de gestión de DT debería abarcar múltiples tipos de deuda para facilitar una gestión integral.

2.3. Priorización de la DT

La priorización de la DT es un paso fundamental una vez que esta ha sido identificada y evaluada. Debido a la naturaleza acumulativa de la DT, no todas las deudas pueden abordarse de inmediato, por lo que se deben tomar decisiones estratégicas para determinar cuáles deben ser mitigadas/subsanadas primero y cuáles pueden posponerse sin comprometer significativamente la estabilidad o evolución del software.

El proceso de priorización tiene como objetivo clasificar los elementos de DT según su impacto en la sostenibilidad del proyecto, los riesgos que presentan y el costo asociado a su mitigación/subsanación. Esto asegura que los equipos de desarrollo enfoquen sus recursos en áreas críticas que maximicen el beneficio a largo plazo, alineándose con los objetivos de negocio.

Aplicación de la priorización de la DT

La aplicación de la priorización de la DT es un proceso complejo que ha sido abordado desde diversas perspectivas en la literatura académica. Diferentes estudios han propuesto

metodologías y enfoques que varían en función de los criterios de evaluación, las herramientas empleadas, y las necesidades específicas de cada proyecto. Estos estudios se enfocan en cómo priorizar la DT de manera que los equipos de desarrollo puedan gestionar eficientemente sus recursos, minimizando riesgos y maximizando el valor entregado.

Existen enfoques que enfatizan el uso de métricas técnicas, tales como la complejidad del código y el coste de mantenimiento [37, 35], mientras que otros se centran en el impacto en los objetivos de negocio y la urgencia de la resolución de la DT [28]. Algunos autores también destacan la importancia de herramientas de soporte a la decisión que combinan análisis multicriterio con técnicas automatizadas para facilitar este proceso [45, 21].

En el próximo capítulo, se realizará una búsqueda exhaustiva en la literatura académica para identificar y analizar los enfoques más relevantes y efectivos en la priorización de la DT. Este análisis permitirá seleccionar la(s) metodología(s) que mejor se alinea(n) con los objetivos de este trabajo. Posteriormente, se aplicará un estudio de caso industrial en la empresa Anfix, donde se implementará(n) dicha(s) metodología(s) en un contexto real, con el fin de priorizar eficazmente la DT, teniendo en cuenta tanto los aspectos técnicos como los estratégicos.

Capítulo 3

Estado del arte

En este capítulo se presenta la estrategia usada para la revisión del estado del arte con el fin de hacer un análisis y evaluación sobre las metodologías propuestas en la literatura para la priorización de la DT en diferentes contextos empresariales. Se presenta el resultado de la revisión y, al final del capítulo, se selecciona una de las metodologías previamente evaluadas que esté mayormente enfocada al negocio.

3.1. Estrategia para la revisión de la literatura

Cadenas de búsquedas y bases de datos de información

Para realizar una búsqueda efectiva en bases de datos académicas, se utiliza una cadena de búsqueda, la cual es una combinación estructurada de términos clave unidos por operadores lógicos [Y] u [O]. Estos operadores ayudan a delimitar y enfocar los resultados, garantizando que se recuperen documentos que contengan todos los términos especificados.

Una cadena de búsqueda bien diseñada permite refinar y ampliar los resultados, encontrando así artículos más relevantes y útiles para una revisión sistemática. En este contexto, las cadenas de búsquedas empleadas son: “Technical Debt Prioritization” y “Technical Debt Prioritization Methodology” respectivamente. Estas cadenas combinan las palabras clave “Technical”, “Debt”, “Prioritization”, y “Methodology” para maximizar la relevancia de los documentos encontrados. De esta manera, se asegura que la búsqueda sea exhaustiva y específica, facilitando la identificación de estudios que aborden los aspectos críticos de la priorización de la DT.

Las bases de datos seleccionadas para la búsqueda fueron SCOPUS, SCIENCE DIRECT, IEEE y ACM, debido a su relevancia y cobertura en el área de la Ingeniería de Software. Adicionalmente, se utilizó la técnica de “bola de nieve” para identificar artículos relevantes a través de las referencias incluidas en los artículos consultados. Esto permitió una recolección amplia y detallada de la literatura relevante.

Criterios de inclusión y exclusión

En esta sección se pretende definir los criterios de inclusión y exclusión de la literatura encontrada en las búsquedas realizadas en las bases de datos. A continuación se muestran estos criterios más detalladamente:

Los criterios de inclusión para la literatura en esta revisión sistemática son:

- Artículos publicados entre el 1 de enero de 2010 y el 31 de diciembre de 2023.
- Publicados en fuentes reconocidas e indexadas.
- Temas principales relacionados con la gestión de la DT y su priorización.
- Temas principales relacionados con metodologías para la priorización de la DT.
- Relevancia para responder las preguntas de investigación definidas.
- Preferiblemente, los artículos deben ser revisados por pares.

Los trabajos excluidos en esta revisión sistemática se seleccionaron con base en criterios específicos para asegurar la objetividad y reproducibilidad del proceso. Los criterios de exclusión son los siguientes:

- Duplicación: Trabajos duplicados en diferentes bases de datos o publicados en revistas y congresos con la misma temática.
- Relevancia: Trabajos que no respondan a ninguna de las preguntas de investigación propuestas. Las preguntas de investigación se detallan en la Sección 1.3, y los trabajos deben abordar directamente estos temas.
- Accesibilidad: Trabajos que requieren un pago adicional para brindar acceso, ya que solo se considerarán artículos de acceso abierto para asegurar la transparencia y la reproducibilidad.
- Foco Temático: Trabajos enfocados en la priorización de la DT en áreas específicas como Salud, Educación, Agricultura, Ciencias Biológicas, Aeronáutica, etc., ya que el enfoque de esta revisión es más amplio y no se limita a una única área de aplicación.
- Idioma: Trabajos no escritos en inglés, para mantener la coherencia en la revisión y dado que la mayoría de la literatura relevante está en inglés.

Con estos criterios de inclusión y de exclusión, claramente definidos, se asegura que la revisión sistemática sea objetiva y reproducible, permitiendo que otros investigadores puedan obtener resultados consistentes utilizando el mismo enfoque.

Proceso de búsqueda y selección de trabajos

El proceso de búsqueda utilizado implicó una búsqueda manual en las bases de datos seleccionadas de artículos específicos publicados, documentos de conferencias, libros y capítulos de libros publicados desde 2010. Se aplicó una cadena de búsqueda utilizando los términos “Technical Debt Prioritization” y “Technical Debt Prioritization Methodology” para encontrar artículos relacionados con los temas clave antes de revisar los resúmenes. Posteriormente, se llevó a cabo una revisión preliminar de los resúmenes para identificar aquellos documentos que parecían ser relevantes para el tema de metodologías para la priorización de la DT y se llevó a cabo la técnica de “bola de nieve”. Los trabajos seleccionados se registraron en la Tabla 3.1 para su posterior revisión.

Fuente	Cadena TDP ^a	Cadena TDPM ^b
Web of Science	66	3
Scopus	82	6
ACM	3	0
IEEE	46	11
Science Direct	16	12
Total: 225	213	22

Tabla 3.1: Número de documentos encontrados en diferentes fuentes para las cadenas de búsquedas “Technical Debt Prioritization” y “Technical Debt Prioritization Methodology”

^aTDP: Technical Debt Prioritization

^bTDPM: Technical Debt Prioritization Methodology

Después de aplicar los criterios de inclusión/exclusión a los 225 artículos registrados en la tabla 3.1, se seleccionó un conjunto final de 33 fuentes para su análisis, siendo estas evaluadas en profundidad más adelante. A continuación se exponen como se distribuyen estas fuentes a partir de su relevancia para la investigación.

Fuentes Primarias

Se identificaron 24 estudios primarios que fueron seleccionados como la base principal de esta revisión. Los estudios primarios son aquellos que presentan datos originales y evidencia directa sobre la priorización de la DT en diferentes contextos empresariales. Estos trabajos incluyen investigaciones empíricas, estudios de caso, experimentos y desarrollos de metodologías específicas que abordan directamente las preguntas de investigación planteadas. La inclusión de estos estudios es fundamental para obtener una comprensión profunda y detallada de las prácticas actuales y las innovaciones en la priorización de DT.

Fuentes Secundarias

Fueron identificados 2 estudios secundarios para nuestra investigación. Los estudios secundarios comprenden Revisiones Sistemáticas de la Literatura (SLR) y Mapas de

Literatura, que sintetizan y analizan de manera integral los hallazgos de múltiples estudios primarios. Estos estudios ofrecen una visión global de las tendencias, metodologías y brechas en la investigación sobre la priorización de DT, proporcionando un contexto más amplio y facilitando la identificación de áreas que requieren mayor exploración. La inclusión de estudios secundarios enriquece la revisión al consolidar conocimientos existentes y alinear la investigación con las prácticas más consolidadas en el campo.

Otras fuentes de interés

Se identificaron 7 fuentes pertenecientes a literatura gris a partir de una búsqueda en Google con la cadena de búsqueda “Prioritization of technical debt”. La literatura gris incluye fuentes no académicas como blogs técnicos, informes de instituciones reconocidas, publicaciones en foros especializados, y documentos de conferencias no indexadas. La inclusión de la literatura gris es esencial para capturar enfoques prácticos, tendencias emergentes y desarrollos recientes que aún no han sido publicados en medios académicos tradicionales. Estas fuentes complementan los estudios primarios y secundarios, proporcionando una perspectiva actualizada y aplicable a contextos empresariales reales, lo que enriquece la comprensión de las metodologías de priorización de DT desde una perspectiva práctica. Además, dentro de estas otras fuentes se incluyen también aquellas que citan los diferentes software utilizados o mencionados en el análisis, proporcionando un respaldo técnico y metodológico a las herramientas empleadas en la investigación.

Los datos extraídos de cada trabajo serán:

- Autores: los autores/colaboradores del artículo.
- Tipo: artículo, libro, conferencia, paper, blog.
- Base de datos: base de datos donde se encontró el documento.
- Título: El título del trabajo.
- País: país de origen.
- Año: El año de publicación.

Clasificación de documentos

Los datos son tabulados y a continuación se presenta un análisis de metadatos clasificados de acuerdo con la base de datos, el año de publicación, país de origen, tipo de documento:

Clasificación por base de datos

En la Figura 3.1, se representan los documentos seleccionados según las bases de datos utilizadas:

- La categoría “IEEE Xplore” tiene el mayor número de documentos.
- Las categorías “Science Direct” y “ACM Digital Library” siguen con menos documentos.
- Las categorías “MDPI”, “ISO”, y “Sage Publications” tienen menos contribuciones.
- La categoría “Unknown” agrupa varias fuentes desconocidas o no especificadas (en su mayoría sitios web o libros).

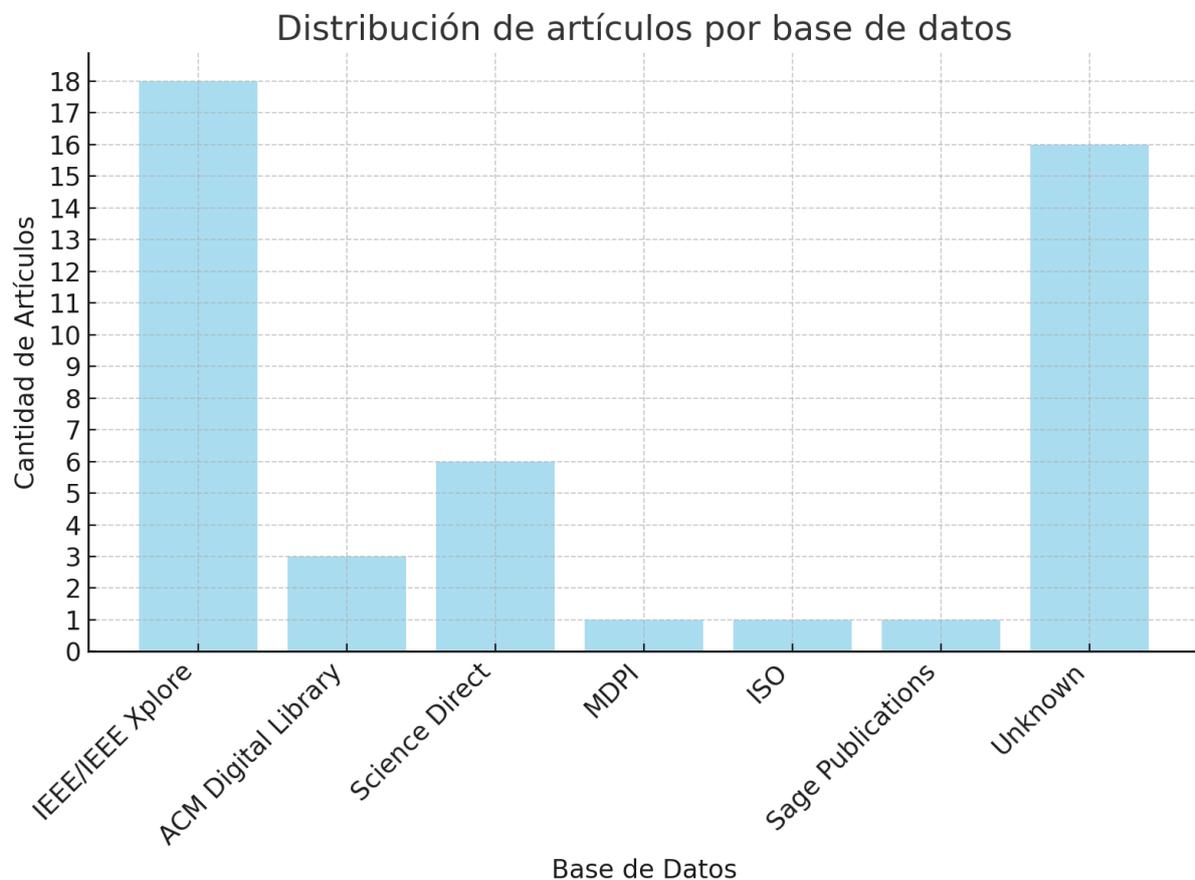


Figura 3.1: Clasificación de documentos por base de datos

Clasificación por año

En la Figura 3.2, se presenta clasificación de los documentos según el año de publicación:

- Se observa una mayor cantidad de publicaciones en los años 2019, 2022 y 2023.
- Hay una distribución más uniforme en otros años, con menos publicaciones en 2015 y 2014.

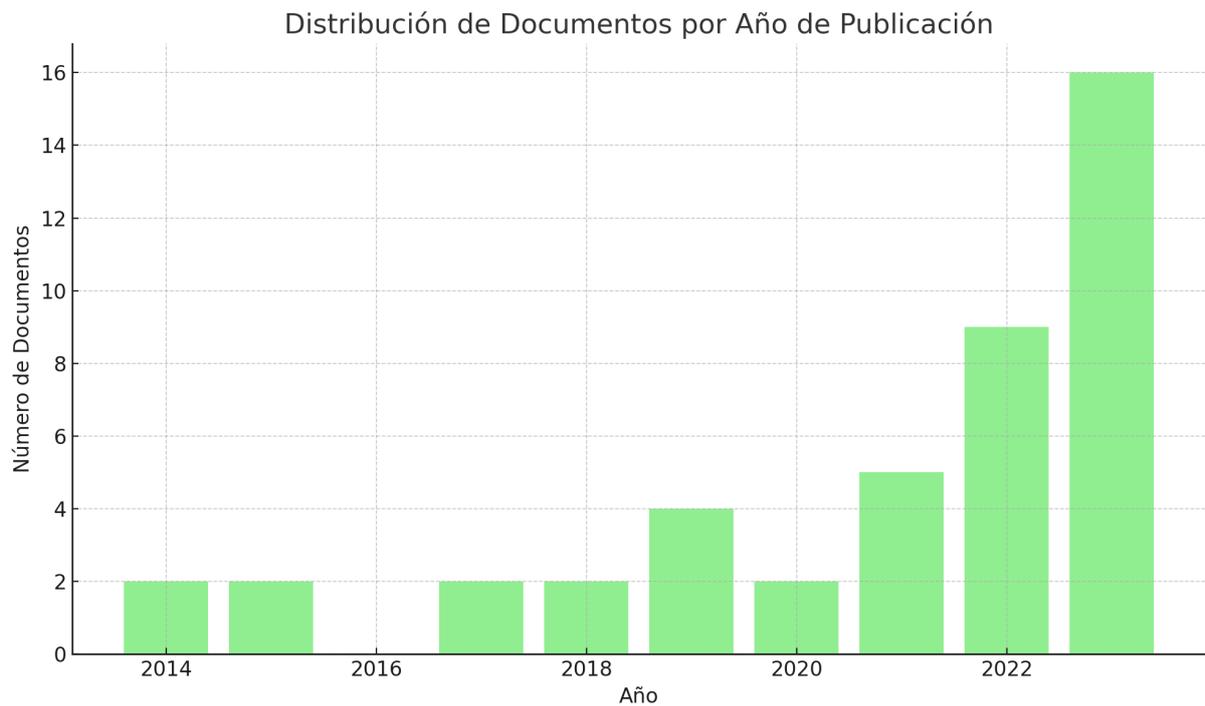


Figura 3.2: Clasificación de documentos por año de publicación

Este gráfico proporciona una visión clara de cómo se han distribuido las publicaciones a lo largo del tiempo, facilitando la identificación de tendencias en la investigación sobre la priorización de la DT.

Clasificación por país

En la Figura 3.3, se realiza la clasificación de documentos según el país de origen de cada uno. Este gráfico proporciona una visión clara de cómo se distribuyen los documentos por país, facilitando la identificación de las contribuciones de diferentes regiones en la investigación sobre la priorización de la DT:

- Brasil y Estados Unidos tienen el mayor número de documentos.
- Polonia y Alemania también tienen varias contribuciones.
- Otros países como Noruega, China, Italia, y Suiza tienen menos documentos.
- La categoría Unknown agrupa varios documentos cuyos países de origen no están especificados (en su mayoría sitios web).

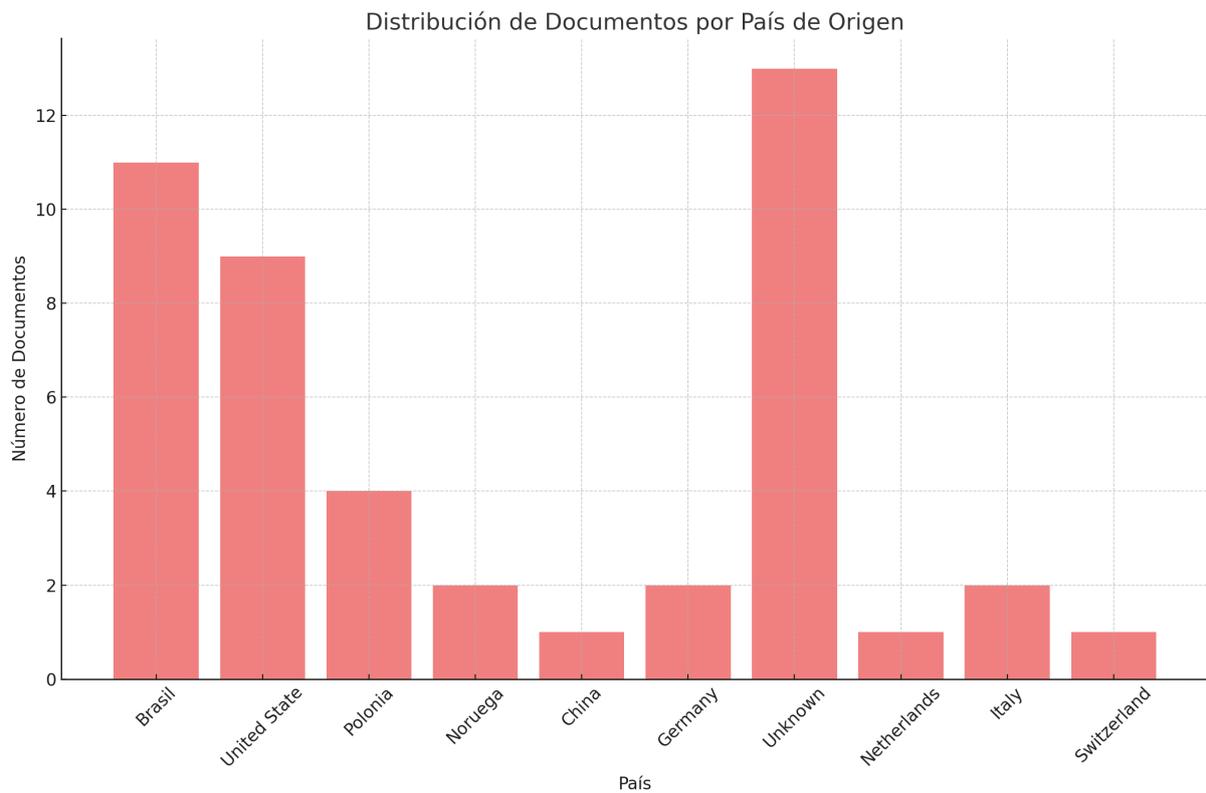


Figura 3.3: Clasificación de documentos por país de origen

Clasificación por tipo de documento

En la Figura 3.4, se representa la clasificación de documentos según el tipo. Este gráfico proporciona una visión clara de cómo se distribuyen los diferentes tipos de documentos, facilitando la identificación de las fuentes y formatos más utilizados en la investigación sobre la priorización de la DT:

- La mayoría de los documentos provienen de “conferencias”.
- Los “artículos” también representan una proporción significativa de estos documentos.
- Los “blogs” hacen referencia sitios web.
- “software”, “reportes” y “libro” son las categorías menos encontradas.

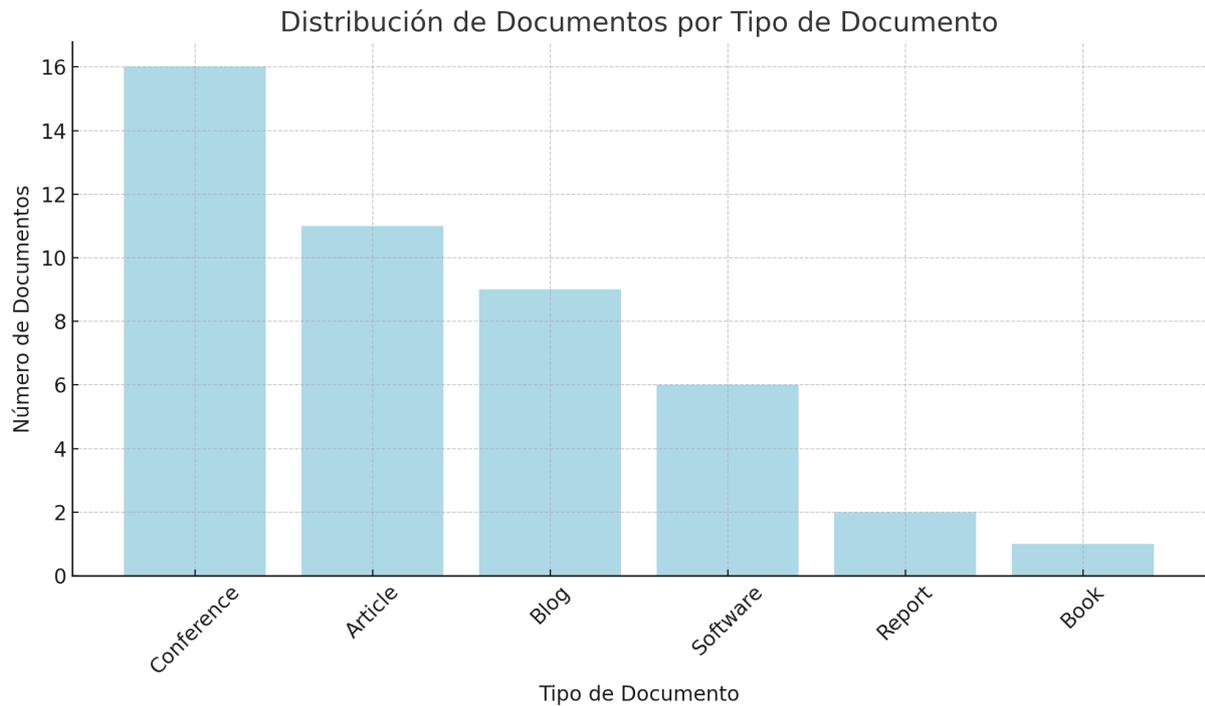


Figura 3.4: Clasificación de documentos por tipo

3.2. Selección de una metodología para la priorización de la DT enfocada en el negocio

Análisis de la bibliografía

En esta subsección, se abordan las *Preguntas de Investigación 1 y 2 1.3*:

- **Pregunta 1:** ¿Existen diferentes metodologías de priorización específicas para cada tipo de DT? ¿Se basan en el contexto/negocio de la empresa? ¿O ambas?
- **Pregunta 2:** ¿Cuáles son las metodologías de priorización de la DT aplicadas en un contexto empresarial y cómo se han evaluado en la literatura?

Para responder a estas preguntas, se realiza un análisis exhaustivo de la literatura existente sobre priorización de DT, enfocándose en metodologías que han sido aplicadas en entornos empresariales y enfocadas al negocio. El análisis se estructura en tres categorías principales: algoritmos evolutivos, enfoques multi-criterio y metodologías orientadas al negocio. A través de esta clasificación, se examinan las fortalezas y debilidades de cada enfoque, su alineación con los objetivos comerciales y su aplicabilidad en diferentes contextos empresariales. Además, se evalúa cómo estas metodologías han sido validadas y aplicadas en estudios de caso y revisiones sistemáticas, proporcionando una visión integral

de su efectividad y relevancia en el ámbito empresarial. Este enfoque permite identificar si las metodologías están diseñadas específicamente para ciertos tipos de DT, si consideran el contexto empresarial, o si combinan ambos aspectos, respondiendo así a ambas preguntas de investigación. A continuación, se presenta una revisión de la literatura organizada.

Algoritmos Evolutivos

Los autores del trabajo [4] introducen un enfoque basado en algoritmos evolutivos multi-objetivo (MOEA) para la priorización de DT. Mediante una evaluación empírica en sistemas de software de código abierto, confirman su aplicabilidad en la industria, mejorando tanto la priorización como el valor del pago de DT. Además, su revisión sistemática en 2020 [3] identifica 24 enfoques distintos de priorización, destacando la escasez de metodologías que integren simultáneamente costo, valor y restricciones de recursos. En 2023, continúan su trabajo enfocándose en la priorización de DT identificada por SonarQube [5], subrayando la importancia de equilibrar valor y costo en la priorización.

Enfoques Multi-Criterio

En el trabajo [34] proponen una estrategia basada en múltiples criterios de decisión para gestionar la DT. Validada mediante un estudio de caso, esta metodología demuestra mejoras en la productividad y eficacia al considerar factores como el impacto, urgencia y esfuerzo necesario para cada DT. Este trabajo subraya la importancia de utilizar criterios variados para una gestión integral de la DT.

Metodologías Orientadas al Negocio

Los estudios de [40, 41, 42, 43] desarrollan técnicas y enfoques para la priorización de la DT que abarcan todo el ciclo de vida del desarrollo de software. Su adopción del paradigma DevOps y la introducción del enfoque continuo de valoración de la DT (CoDVA). Seguido a esto [42] demuestra cómo una gestión efectiva de la DT puede mejorar la estabilidad y calidad del sistema, además de optimizar el valor del producto desde una perspectiva empresarial. Su metodología flexible permite la adaptación a diversos dominios, integrando múltiples criterios de priorización que responden a las necesidades específicas de diferentes organizaciones.

Por otro lado, en [29] realizan una revisión sistemática que clasifica los enfoques de priorización de la DT en una taxonomía, identificando factores cualitativos y herramientas faltantes, y proponiendo direcciones para futuras investigaciones. Este trabajo proporciona una visión integral de los métodos actuales y resalta la necesidad de enfoques más completos para una priorización efectiva.

Otros de los estudios llamativos es [6, 8, 7, 10, 37, 9], que presentan una metodología significativa a través del framework Tracy [8], desarrollado con la participación de 49 profesionales de la industria. Tracy se centra en alinear la priorización de la DT con los objetivos comerciales mediante la gestión de procesos empresariales (BPM). Su evaluación en diversos contextos industriales ha demostrado su eficacia para mejorar la alineación y

comunicación entre stakeholders técnicos y comerciales, identificando factores comerciales que afectan la priorización de la DT.

Caracterización y Comparativa de los trabajos revisados

El estado del arte en la priorización de la DT abarca una variedad de enfoques metodológicos que van desde estrategias generales hasta métodos específicos orientados al negocio. Para organizar de manera clara y coherente la revisión de la literatura, se ha estructurado el análisis en dos partes principales: una caracterización general de los trabajos revisados y una comparación detallada de las metodologías y técnicas específicas para la priorización de la DT.

Caracterización

A continuación, se presenta una tabla que resume los diferentes trabajos analizados, clasificándolos según su tipo (metodología, herramienta, revisión sistemática, etc.) y destacando sus principales contribuciones.

Referencia	Título	Tipo de Trabajo	Año
[4]	Technical Debt Prioritization: A Search-Based Approach	Metodología	2019
[3]	A systematic literature review of technical debt prioritization	Revisión Sistemática	2020
[5]	How SonarQube-identified technical debt is prioritized: An exploratory case study	Herramienta	2023
[34]	A strategy based on multiple decision criteria to support technical debt management	Metodología	2017
[40]	Continuous Debt Valuation Approach (CoDVA) for Technical Debt Prioritization	Metodología	2020
[41]	Adopting DevOps Paradigm in Technical Debt Prioritization and Mitigation	Metodología	2022
[43]	Technical Debt Prioritization in Telecommunication Applications: Why the Actual Refactoring Deviates from the Plan and How to Remediate It? Case Study in the COVID Era	Metodología	2022
[42]	Business-driven technical debt management using Continuous Debt Valuation Approach (CoDVA)	Metodología	2023
[29]	A systematic literature review on Technical Debt prioritization: Strategies, processes, factors, and tools	Revisión Sistemática	2021
[6]	Aligning technical debt prioritization with business objectives: A multiple-case study	Metodología	2018
[8]	Tracy: A Business-Driven Technical Debt Prioritization Framework	Metodología	2019
[10]	Business-Driven Technical Debt Prioritization: An Industrial Case Study	Metodología	2021
[37]	Business-Driven Technical Debt Prioritization: A Replication Study	Metodología	2022
[9]	What's behind tight deadlines? Business causes of technical debt	Metodología	2023
[44]	Architectural technical debt in microservices: A case study in a large company	Metodología	2019
[21]	PriorTD: A Method for Prioritization Technical Debt	Herramienta	2022
[16]	Technical Debt Prioritization Using Predictive Analytics	Metodología	2017

Tabla 3.2: Caracterización de los trabajos revisados sobre priorización de la DT.

La tabla 3.2 anterior proporciona una visión general de cada trabajo revisado, clasificándolos según su tipo y destacando sus principales contribuciones al campo de la priorización de la DT. Esta caracterización permite identificar rápidamente qué tipo de enfoque adopta cada estudio y cómo contribuye al entendimiento y manejo de la DT.

Comparativa de metodologías

Tras la caracterización de los distintos estudios, se lleva a cabo una comparación detallada de las metodologías y técnicas propuestas para la priorización de la DT. Esta evaluación se limita únicamente a las investigaciones que introducen una metodología o técnica concreta, excluyendo herramientas y revisiones sistemáticas, con el fin de asegurar una valoración coherente y pertinente.

Descripción del enfoque	Alineación con Objetivos Comerciales	Facilidad de Uso	de	Impacto en la Productividad	Aplicabilidad
Decisión Multi-Criterio (Ribeiro, 2017) Múltiples criterios de decisión (impacto, urgencia, esfuerzo)	Medio	Alto		Alto	General
Algoritmos Evolutivos (Alfayez & Boehm, 2019) Algoritmos multi-objetivo (MOEA)	Medio	Medio		Alto	Código abierto
Estrategia de Ventas Integradas (Stochel, 2020) Integración de ventas en priorización de DT	Alto	Medio		Alto	Telecomunicaciones
Integración en DevOps (Stochel, 2022a) Priorización de DT dentro de procesos DevOps	Alto	Alto		Alto	General
Refactorización para Remediación (Stochel, 2022b) Técnicas de refactorización para reducir DT	Medio	Medio		Alto	Telecomunicaciones
Enfoque CoDVA (Stochel, 2023) Priorización continua de DT	Alto	Medio		Alto	Telecomunicaciones
Gestión de Procesos Empresariales (BPM) (Almeida, 2018) Gestión de procesos empresariales para DT	Alto	Medio		Alto	General
Framework Tracy (Almeida, 2019a) Priorización orientada al negocio mediante BPM	Alto	Alto		Alto	General
Evolución de Tracy (Almeida, 2021) Alineación con objetivos comerciales	Alto	Medio		Alto	General
Replicación de Tracy en Startups (Almeida, 2022) Adaptación de Tracy para startups	Alto	Alto		Alto	Startups
Análisis de Causas Empresariales (Almeida, 2023) Identificación de causas comerciales de DT	Alto	Medio		Medio	General
Migración a Microservicios (Soares, 2019) Migración a micros...	Medio	Medio		Medio	Microservicios

En la tabla 3.3, se destaca que las metodologías orientadas al negocio, como el Framework Tracy y las estrategias integradas en DevOps, presentan una alta alineación con los objetivos comerciales, lo que las hace particularmente valiosas para organizaciones que buscan no solo gestionar la DT desde una perspectiva técnica, sino también alinearla con sus metas empresariales. Estas metodologías también tienden a tener un alto impacto en la productividad y una buena aplicabilidad en diversos contextos.

Por otro lado, enfoques como los Algoritmos Evolutivos y el Análisis Predictivo ofrecen un alto impacto en la productividad mediante la optimización de la priorización, aunque su alineación con objetivos comerciales puede ser más limitada en comparación con metodologías específicamente orientadas al negocio.

Selección del Framework Tracy

El Framework Tracy se destaca entre las metodologías revisadas para la priorización de la DT por varias razones clave que lo hacen particularmente adecuado para contextos empresariales donde la alineación con los objetivos comerciales es crucial. A continuación, se detallan las razones que justifican su selección:

- **Alineación con los Objetivos Comerciales:** El principal valor del Framework Tracy radica en su capacidad para integrar la gestión de la DT con los objetivos estratégicos de la empresa. Este enfoque permite que la priorización de DT no solo se guíe por factores técnicos, sino que también considere los impactos comerciales. Esto se logra a través de la participación directa de stakeholders comerciales en el proceso de priorización, lo que asegura que las decisiones tomadas refuercen los objetivos de negocio, como la rentabilidad, la competitividad y la satisfacción del cliente. En estudios previos, como el de [8], Tracy ha demostrado su eficacia en mejorar la toma de decisiones estratégicas mediante la alineación de prioridades técnicas con metas comerciales. Esto es especialmente valioso en industrias donde la agilidad y la capacidad de respuesta a las demandas del mercado son fundamentales.
- **Facilidad de Uso:** El diseño del Framework Tracy ha sido iterativamente perfeccionado para asegurar su facilidad de uso. Este framework ha sido evaluado y refinado en múltiples estudios de caso, lo que ha permitido ajustar su aplicación a diversos entornos empresariales y contextos de desarrollo de software. El enfoque de Tracy incluye herramientas y guías específicas que simplifican su adopción en distintos contextos organizativos, lo que permite a las empresas integrar rápidamente esta metodología en sus procesos existentes sin requerir una curva de aprendizaje pronunciada.
- **Impacto en la Productividad:** Una de las ventajas más significativas del Framework Tracy es su impacto positivo en la productividad organizacional. Al facilitar una mejor comunicación y alineación entre equipos técnicos y comerciales, Tracy minimiza las fricciones y malentendidos que pueden surgir en la priorización de la DT. Esto, a su vez, permite una ejecución más rápida y eficiente de las tareas, lo

que resulta en una reducción de costos y tiempos de desarrollo. Por ejemplo, en un estudio industrial de cinco meses realizado por [10], se observó que la implementación del Framework Tracy no solo mejoró la priorización de la DT, sino que también optimizó el uso de los recursos, lo que resultó en un incremento de la productividad del equipo y una mejor alineación de los resultados con las expectativas comerciales.

- **Alta Aplicabilidad en Diversos Contextos Industriales:** El Framework Tracy ha demostrado su flexibilidad y efectividad en una amplia gama de industrias y contextos. Desde grandes corporaciones hasta startups de software, Tracy ha sido validado en diversos entornos, lo que subraya su adaptabilidad. En particular, su replicación en una startup de software, como se describe en [37], mostró cómo Tracy puede ser adaptado para satisfacer las necesidades específicas de organizaciones más pequeñas y dinámicas, sin perder su eficacia en la alineación de los objetivos técnicos y comerciales.
- **Validación Empírica y Estudio de Casos:** El Framework Tracy no solo ha sido teóricamente propuesto, sino que ha sido ampliamente validado a través de estudios empíricos en diferentes escenarios industriales. Esta validación incluye estudios de caso que abarcan distintos tamaños de empresa y sectores, lo que refuerza su credibilidad y relevancia práctica. Por ejemplo, el estudio de caso industrial mencionado en [10] proporciona evidencia concreta de cómo Tracy ha sido implementado y adaptado a lo largo del tiempo, mostrando mejoras continuas en la priorización de la DT y su impacto positivo en el alineamiento con las metas comerciales.

Conclusión: El Framework Tracy se presenta como la metodología más robusta y adecuada para la priorización de la DT en contextos donde la alineación con los objetivos comerciales es esencial. Su facilidad de uso, su impacto positivo en la productividad y su validación en diversos contextos industriales lo convierten en una herramienta esencial para la gestión efectiva de la DT en proyectos de software.

3.3. Descripción detallada del Framework Tracy

Este framework integra procesos y métricas de negocio para priorizar la DT de manera que maximice el valor empresarial. Esta estructura se representa en la Figura 3.5, la cual ilustra los principales componentes del framework y sus interrelaciones. En dicha figura, podemos observar cómo los activos de TI, los elementos de configuración y las fuentes de valor se organizan para proporcionar una visión clara del impacto que cada componente tiene en la priorización de la DT, permitiendo a las empresas tomar decisiones informadas y alineadas con sus metas estratégicas [8].

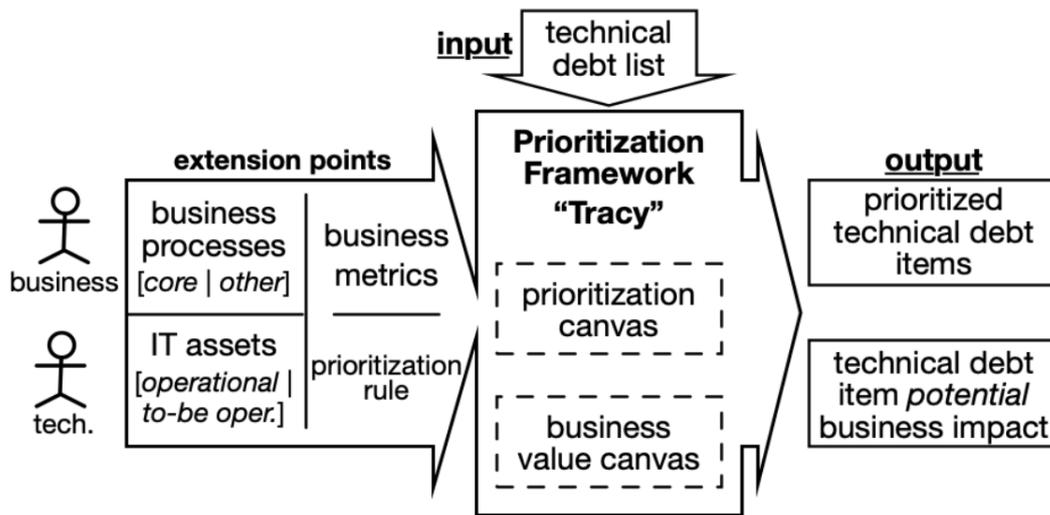


Figura 3.5: Framework Tracy [8]

Principales conceptos

El Framework Tracy se sustenta en una serie de conceptos fundamentales que permiten su aplicación efectiva en diversos contextos empresariales:

- Lista de DT: Conjunto de partidas de DT identificadas en los sistemas de la organización. Estas partidas se priorizan según su impacto en los activos de TI y los procesos comerciales que apoyan.
- Elementos de Configuración: Este es un artefacto que puede verse afectado por una partida de DT, como un fragmento de código, una clase, una biblioteca, un módulo, un sistema, una base de datos, un servidor, un elemento de arquitectura o un servicio. Los elementos de configuración generalmente pertenecen al dominio de las partes interesadas técnicas, mientras que las partes interesadas comerciales a menudo no dominan la información sobre los elementos de configuración y sus responsabilidades. Un elemento de configuración puede estar compuesto y depender de otros elementos de configuración. En este modelo se propone que los elementos de configuración pueden presentar tres estados:
 - Operativo (operational): un artefacto que está en producción, siendo utilizado directa o indirectamente por clientes o usuarios;
 - Operativo a futuro (To-Be operational): un nuevo artefacto que está en desarrollo o en planificación;
 - Legado (legacy): un artefacto que se planea discontinuar o reemplazar por otro.

Como dato importante, cabe destacar que una partida de DT afecta a un elemento de configuración. Por ejemplo, un “elemento de deuda de prueba” puede afectar un “módulo de servicio”; un “elemento de deuda de construcción” puede afectar una aplicación; y un “elemento de deuda arquitectónica” puede involucrar instancias de servidor. Los elementos de configuración admiten activos de TI.

- **Activos de TI:** Concepto abstracto que representa cualquier solución, producto, servicio o aplicación dentro del portafolio de la empresa, los cuales son soportados por los elementos de configuración. Estos activos se evalúan en función de su capacidad para generar valor comercial, independientemente de los detalles técnicos específicos. Los activos de TI pueden clasificarse de manera similar a los elementos de configuración, es decir, pueden ser operacionales, futuros operacionales o heredados.
- **Fuentes de valor:** Las fuentes de valor representan todo aquello que genera valor para el negocio a partir de los activos de TI. Estas pueden ser funciones o características de un sistema que soportan procesos clave del negocio. Las fuentes de valor se clasifican en:
 - **Centrales (Core):** Forman parte del núcleo del negocio.
 - **Otras (Others):** No son parte del núcleo del negocio.

Además, las fuentes de valor pueden clasificarse en función de su frecuencia de uso (alta o baja). Esta información puede obtenerse mediante herramientas de seguimiento o en base a la percepción de los stakeholders.

Protocolo de Recolección y Análisis de Datos para la Implementación del Framework Tracy

Recopilación de la Lista de partidas de DT

El primer paso consiste en recopilar una lista de las partidas de DT presentes en los equipos involucrados. Esta lista sirve como base para todo el proceso de análisis y priorización posterior.

Actividades: Actividades:

- **Identificación de Problemas:** Utilización de herramientas de gestión de proyectos como Jira para registrar incidencias, errores y aspectos del código que requieren atención.
- **Revisión de Backlogs:** Análisis de los backlogs existentes para identificar tareas pendientes relacionadas con la DT.
- **Reuniones Iniciales:** Organización de sesiones de trabajo con los equipos de desarrollo para asegurar que todos los elementos de DT sean identificados y documentados adecuadamente.

Stakeholders involucrados:

- **Developer Manager:** Responsable de coordinar la identificación de las partidas de DT en los equipos de desarrollo.
- **Miembros del Equipo de Desarrollo:** Contribuyen con su conocimiento técnico para identificar y documentar las deudas existentes.
- **Product Manager:** Ayuda a contextualizar las DT en relación con las funcionalidades del producto.
- **Chief Product Officer (CPO):** Proporciona una visión estratégica para asegurar que la identificación de DT esté alineada con los objetivos del negocio.

Identificación y Clasificación de las Partidas de DT

Una vez recopilados las partidas de DT, es fundamental establecer un orden de prioridades basado en criterios técnicos. Este paso asegura que los aspectos más críticos y que impactan negativamente en el desarrollo y mantenimiento del sistema sean abordados primero.

Actividades:

- **Evaluación de Impacto Técnico:** Analizar cómo cada elemento de deuda afecta la estabilidad, rendimiento y escalabilidad del sistema.
- **Clasificación por Severidad:** Categorizar las incidencias en niveles de severidad (crítica, alta, media, baja) según su impacto técnico.
- **Estimación de Esfuerzo:** Determinar el esfuerzo necesario para resolver cada partida de DT, considerando tiempo y recursos.

Stakeholders involucrados:

- **Developer Manager:** Lidera la evaluación y clasificación de las incidencias técnicas.
- **Equipo de Desarrollo:** Colabora en la evaluación del impacto y el esfuerzo requerido para cada elemento.
- **Arquitecto de Software:** Aporta conocimientos sobre la estructura del sistema y las implicaciones técnicas de cada deuda.

Identificación de Elementos de Configuración Afectados

Este paso implica identificar los componentes específicos del sistema que se ven afectados por la DT. Comprender qué elementos de configuración están involucrados facilita el análisis de cómo estas partidas de DT impactan en los procesos de negocio y en los activos de TI.

Actividades:

- **Mapeo de Deuda a Componentes:** Asociar cada partida de DT con los módulos, bases de datos, servicios externos u otros componentes del sistema.
- **Documentación de Relación:** Crear diagramas y tablas que muestren la relación entre las deudas técnicas y los elementos de configuración afectados.
- **Análisis de Dependencias:** Identificar dependencias entre diferentes componentes que puedan amplificar el impacto de la DT.

Stakeholders involucrados:

- **Equipo de Infraestructura:** Aporta información sobre los componentes tecnológicos y su interrelación.
- **Product Manager:** Ayuda a comprender cómo los elementos de configuración impactan en las funcionalidades del producto.
- **Developer Manager y Equipo de Desarrollo:** Colaboran en la identificación precisa de los elementos afectados.

Identificación de Activos de TI y Procesos de Negocio. Modelado de su Relación

Es esencial determinar cómo los elementos de configuración afectados por la DT influyen en los procesos de negocio y los activos de TI de la empresa. Este análisis permite comprender el impacto de la DT en la operatividad y competitividad del negocio.

Actividades:

- **Mapeo de Procesos de Negocio:** Identificar y listar los procesos de negocio que dependen de los elementos de configuración afectados.
- **Modelado de Impacto:** Utilizar herramientas de modelado para visualizar cómo la DT afecta cada proceso de negocio.
- **Evaluación de Consecuencias:** Analizar las repercusiones en términos de eficiencia, costos, satisfacción del cliente y cumplimiento normativo.

Stakeholders involucrados:

- **Partes Interesadas Empresariales:** Representantes de áreas como Ventas, Producción, Finanzas, RRHH, Marketing, Soporte y Operaciones.
- **Product Manager:** Facilita la comprensión de cómo los procesos de negocio se relacionan con las funcionalidades del producto y los activos de TI.
- **Developer Manager:** Asegura que el análisis técnico se alinee con los procesos de negocio identificados.

Priorización de las Partidas de DT Basada en Objetivos Empresariales

Este paso consiste en alinear la priorización de la DT con los objetivos estratégicos del negocio. Mediante el mapeo de procesos de negocio priorizados, los activos de TI y los elementos de configuración, se asegura que las acciones correctivas contribuyan al valor empresarial.

Actividades:

- **Mapeo de Procesos a DT:** Relacionar los procesos de negocio priorizados con las partidas de DT identificadas.
- **Asignación de Prioridades:** Utilizar una escala de priorización (por ejemplo, del 1 al 6) para clasificar cada partida de DT según su alineación con los objetivos empresariales.
- **Evaluación de Impacto Empresarial:** Considerar métricas como impacto en la experiencia del cliente, riesgos de seguridad, valor de negocio, cumplimiento legal, escalabilidad, costos operativos, infraestructura, rendimiento e innovación. Se pueden utilizar métricas propias de la empresa, que le aporten valor añadido.

Stakeholders involucrados:

- **CEO y CPO:** Proveen la visión estratégica y los objetivos empresariales que guían la priorización.
- **Product Manager:** Asegura que la priorización técnica esté alineada con las necesidades del producto y del mercado.
- **Developer Manager y Equipo de Desarrollo:** Colaboran en la asignación de prioridades técnicas basadas en la evaluación realizada.

Comparación y Discusión de Priorizaciones desde Perspectivas Empresariales y Técnicas

El último paso implica comparar las prioridades asignadas desde las perspectivas empresarial y técnica, y discutir los resultados con las partes interesadas para validar y ajustar la priorización final de la DT.

Actividades:

- **Análisis Comparativo:** Evaluar las diferencias y similitudes entre las prioridades técnicas y empresariales.
- **Reuniones de Discusión:** Organizar sesiones de revisión con las partes interesadas para presentar y discutir los resultados de la priorización.
- **Ajuste de Prioridades:** Incorporar feedback y realizar ajustes necesarios para asegurar una priorización equilibrada y alineada con ambos puntos de vista.
- **Documentación de Decisiones:** Registrar las decisiones tomadas durante las discusiones para futuras referencias y seguimiento.

Stakeholders involucrados:

- **Partes Interesadas de Ambos Equipos:** Incluyendo representantes de áreas técnicas y empresariales.
- **CEO y CPO:** Participan en la validación de la priorización final para asegurar su alineación con la estrategia empresarial.
- **Product Manager:** Facilita la comunicación entre los equipos técnicos y empresariales.
- **Developer Manager y Equipo de Desarrollo:** Aportan la perspectiva técnica necesaria para ajustar las prioridades.

Tiempo total estimado

Es importante reconocer que el tiempo de resolución para cada uno de los pasos en la implementación del Framework Tracy puede variar significativamente en función de múltiples factores. La cantidad de DT identificada es un factor determinante, ya que un mayor número de incidencias requerirá más tiempo para su recopilación, análisis y priorización. Además, las prioridades propias del negocio pueden influir en la rapidez con la que se deben abordar ciertas partidas de DT, especialmente aquellas que impactan directamente en los objetivos estratégicos de la empresa. La cantidad de personas o equipos involucrados también juega un papel crucial; un mayor número de stakeholders puede facilitar una distribución más eficiente de las tareas, pero también puede introducir complejidades adicionales en la coordinación y toma de decisiones. Otros factores como la experiencia del equipo, la disponibilidad de recursos y la complejidad técnica de las deudas identificadas también pueden afectar los plazos estimados. Considerando estas variables, el cálculo medio del tiempo requerido para completar todo el proceso de implementación del Framework Tracy se sitúa aproximadamente en 8 a 12 semanas. Este margen de tiempo incluye posibles retrasos y ajustes necesarios para adaptarse a las particularidades de cada organización, ofreciendo así una estimación más realista y flexible para la planificación del proyecto.

Plantillas de trabajo como Herramientas de Apoyo en la Implementación del Framework Tracy

En la implementación del Framework Tracy, las plantillas de trabajo desempeñan un papel crucial al proporcionar estructuras sistemáticas que facilitan la recolección, análisis y priorización de la DT. Estas herramientas no solo estandarizan el proceso, sino que también aseguran una alineación coherente entre los aspectos técnicos y los objetivos empresariales. A continuación, se describen detalladamente las principales plantillas utilizadas a lo largo de la aplicación.

Plantilla de Relación Procesos de Negocio, Activos de TI y Elementos de Configuración

La Plantilla de Relación Procesos de Negocio, Activos de TI y Elementos de Configuración es una herramienta esencial dentro del Framework Tracy que facilita la visualización y comprensión de las interdependencias entre los procesos de negocio, los activos de TI que los soportan y los elementos de configuración afectados por la DT. Esta plantilla permite mapear de manera clara y estructurada cómo cada proceso de negocio crítico está respaldado por activos de TI específicos y cómo la DT en estos activos puede influir directamente en la eficiencia y efectividad de los procesos. La Figura 3.6 ilustra de forma ejemplar la relación entre los elementos de configuración, los activos de TI y los procesos de negocio que pueden presentarse en una empresa, proporcionando una representación visual que complementa y refuerza la comprensión teórica de estas interrelaciones.

Relación entre los elementos de configuración, activos de TI y procesos del negocio

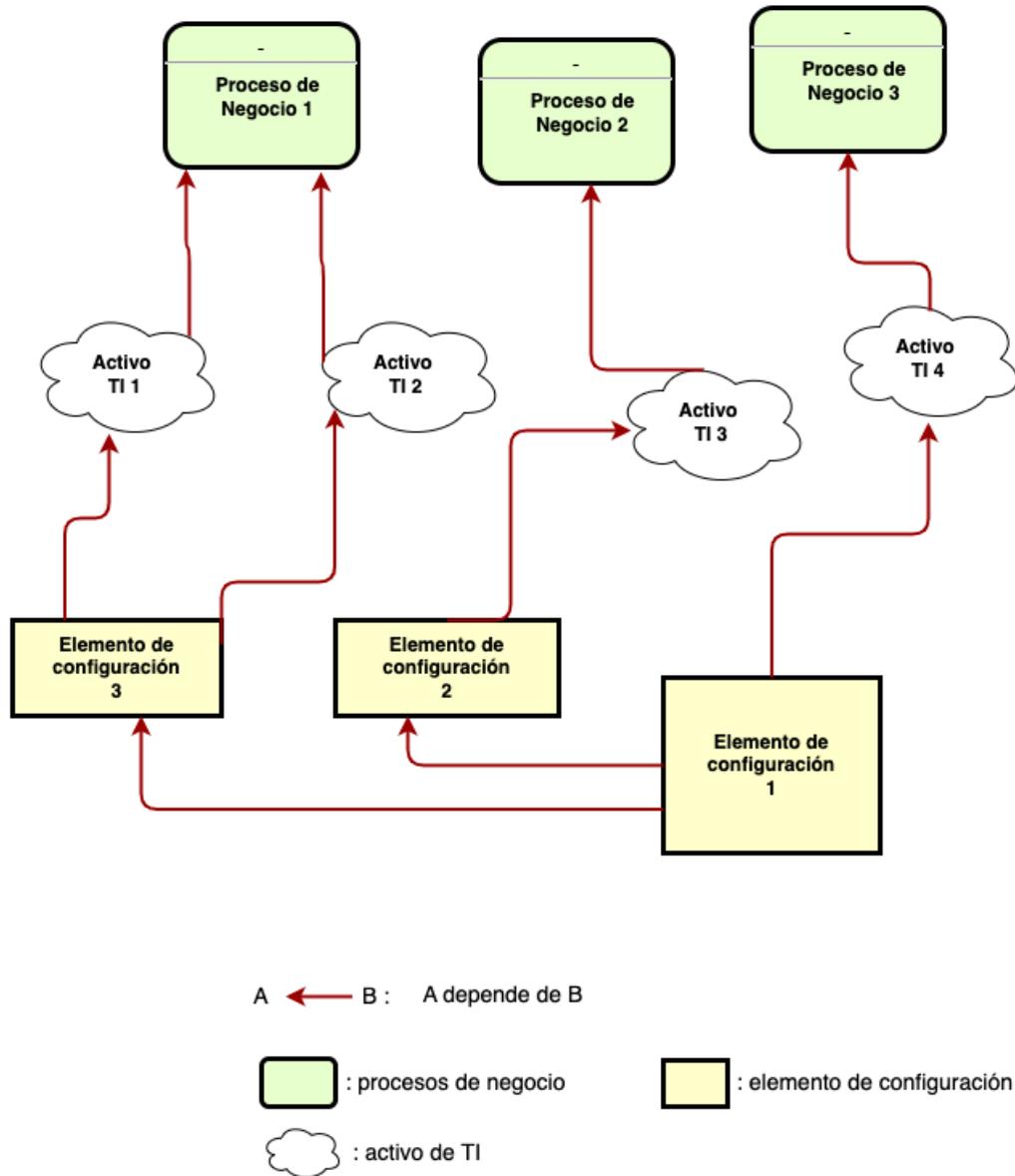


Figura 3.6: Plantilla de relación entre los diferentes artefactos propuestos por el framework Tracy [10]

Plantilla de priorización

Este framework también propone una plantilla que ayuda a visualizar las principales entidades involucradas en la priorización de la DT (los activos de TI, las fuentes de valor y sus relaciones). El objetivo de esta plantilla es guiar a los participantes con ejercicios para pensar sobre los activos de TI, las fuentes de valor y su clasificación. Los participantes pueden mirar “la misma página” y discutir las percepciones de valor empresarial. La

Figura 3.7 muestra de forma ilustrativa como los activos de TI operativos, heredados y futuros respaldan las fuentes de valor clasificadas según su valor comercial (centrales u otros) y su frecuencia de uso (alta o baja). El atributo de uso de la fuente de valor solo se puede relacionar con activos de TI operativos y heredados. Dado que un futuro activo de TI aún no se está utilizando, no consideramos su frecuencia de uso para la priorización de la DT [10].

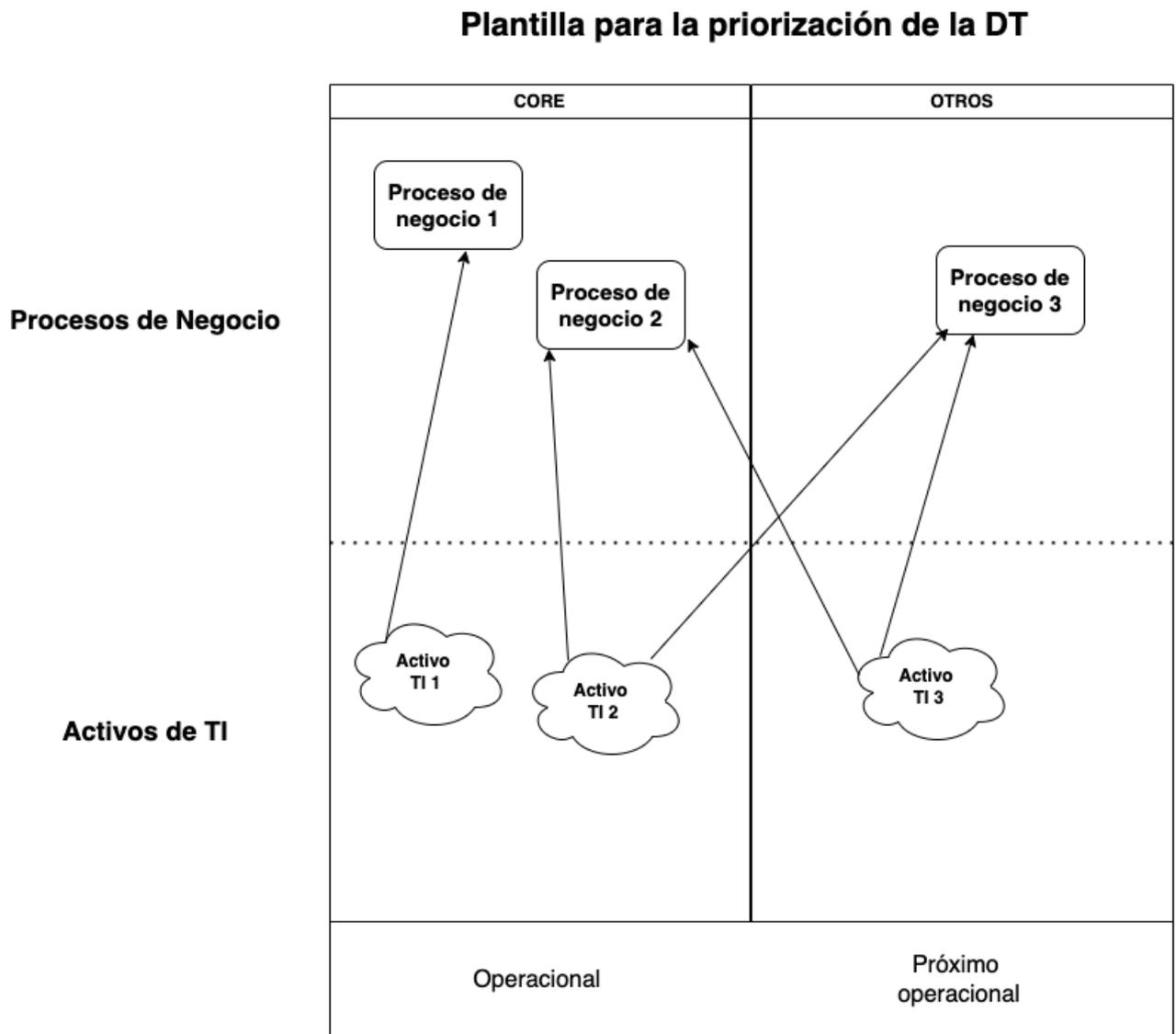


Figura 3.7: Plantilla de trabajo propuesta para el Framework Tracy [10]

Plantilla de reglas de priorización

La regla de priorización se basa en la relación entre los activos de TI y las fuentes de valor, clasificando la prioridad de la DT en una escala de 1 (máxima prioridad) a 10

(mínima prioridad). Por ejemplo, una partida DT que afecta un activo de TI que soporta una fuente de valor central y de alta frecuencia de uso tendrá una prioridad alta (1). En cambio, una partida de DT que afecta un activo heredado con baja frecuencia de uso tendrá una prioridad baja (10). La Figura 3.8 muestra como se pueden distribuir las prioridades asignadas en la “plantilla de priorización”. Los elementos relacionados con activos que soportan procesos de negocio críticos y de alta frecuencia de uso reciben una prioridad alta, mientras que aquellos que afectan procesos secundarios o activos heredados reciben una prioridad más baja.

Plantilla del valor empresarial



Frecuencia / Impacto en el tiempo		Frecuencia / Impacto en el tiempo		Frecuencia / Impacto en el tiempo	
(a, 2)	Métrica 1	(a, c, 3)	Métrica 4	(b, 2)	Métrica 5
(b, 3)	Métrica 2	(b, 2)	Métrica 3	(a, c, 3)	Métrica 7
(c, 1)	Métrica 3	(a, b, 1, 2)	Métrica 5		
		(c, 1)	Métrica 6		

Figura 3.8: Tabla para relacionar a partir de las métricas los Activos de TI y los Procesos de Negocio [10]

Las reglas de priorización del Framework Tracy son flexibles y permiten adaptaciones según el contexto empresarial específico. Las empresas pueden ajustar las escalas de prioridad y agrupar relaciones similares con la misma prioridad. Por ejemplo, en algunos casos, los propietarios de productos sugirieron agrupar las prioridades en rangos más reducidos para facilitar la toma de decisiones.

Caso de estudio

4.1. Introducción al caso de estudio

En el capítulo anterior, se llevó a cabo una comparación y evaluación detallada de diferentes metodologías para la priorización de la DT. Tras este análisis, se seleccionó la metodología más adecuada para abordar la DT en el contexto de una empresa real. El presente capítulo se centra en la aplicación práctica de esta metodología seleccionada en un caso de estudio industrial, utilizando como escenario la empresa Anfix, una organización del sector tecnológico.

El objetivo principal de este caso de estudio es analizar la efectividad de la metodología seleccionada para priorizar la DT, evaluando su impacto en la gestión del software y los procesos de la empresa. A lo largo de este capítulo, se expondrán los resultados obtenidos durante su implementación, así como el análisis de dichos resultados en función de los objetivos estratégicos de Anfix.

Hay distintos enfoques de indagación en la ejecución de estudios empíricos que pueden ser utilizados dependiendo de las interrogantes planteadas, el nivel de control que el investigador ejerce sobre los eventos actuales y el enfoque en acontecimientos históricos o contemporáneos. La investigación de casos se recomienda cuando las preguntas de investigación se centran en el “cómo” y el “por qué”, y cuando el investigador tiene limitado control sobre el fenómeno en estudio, abarcando eventos actuales en un entorno real. Los casos pueden ser exploratorios, descriptivos o explicativos, y pueden involucrar un único caso o varios casos con el objetivo de llegar a generalizaciones basadas en el análisis de evidencias. Los estudios de casos incluyen diversos elementos, tales como preguntas de investigación, proposiciones, unidades de análisis, relaciones entre datos y proposiciones, así como criterios para la interpretación de los hallazgos. El método de estudio de casos se organiza en las siguientes actividades principales [25]:

- Preparación del estudio de caso: Se establece el plan de intervención, se diseña el enfoque de estudio y el entorno, se planifica el acceso y los principios para la

clasificación de las pruebas, y se establece un cronograma para la realización del caso.

- **Recolección de evidencias:** Se procede a obtener las pruebas que respaldan la línea de razonamiento que responde a las cuestiones de investigación. Las pruebas más significativas pueden incluir documentos, registros electrónicos, observaciones y objetos físicos.
- **Ejecución y análisis de evidencias:** Representa la etapa más compleja del método. Aquí se examinan, clasifican, tabulan o someten a pruebas diversas combinaciones de pruebas, tanto cualitativas como cuantitativas. Requiere la definición de estrategias y técnicas de análisis.
- **Informar los resultados obtenidos:** Comunica los hallazgos del análisis mediante la definición de audiencias específicas y la gestión ética de la identidad de las fuentes de pruebas. La estructura del informe se compone de la descripción, explicación y valoración de los resultados del caso.

El diseño del estudio de casos puede ser: holístico (un único caso/contexto con una única unidad de análisis), estructurado (un único caso/contexto con varias unidades de análisis), o multi-casos (varios casos/contextos, cada uno con una única unidad de análisis o varios casos/contextos, cada uno con varias unidades de análisis) [25]. Estas alternativas se pueden observar en la Figura 4.1.

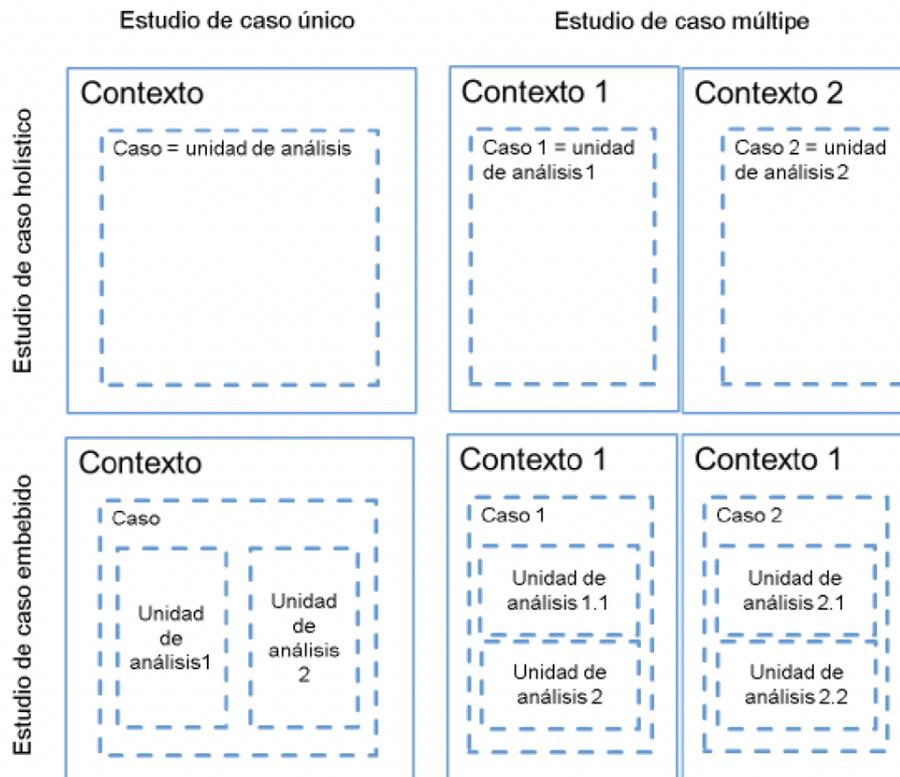


Figura 4.1: Alternativas para diseño de unidades y contexto de análisis en estudios de casos

En este estudio se ha optado por un enfoque holístico, utilizando un único caso de estudio con una sola unidad de análisis. Según la metodología seleccionada en capítulos anteriores, el estudio se centró en la empresa Anfix, debido a su relevancia en la industria de tecnología empresarial y su necesidad de priorizar eficazmente la DT. Este caso de estudio holístico es apropiado para analizar en profundidad el impacto de la DT en los activos de TI y las fuentes de valor de la empresa, ya que se trata de una situación representativa y significativa para la investigación.

La unidad de análisis en este caso es el sistema de gestión empresarial completo de Anfix, compuesto por múltiples módulos y servicios interconectados. El uso de una única unidad de análisis permitió concentrar los esfuerzos de investigación en el impacto de la DT sobre los módulos más críticos para el negocio. Esto asegura un enfoque detallado en las áreas más relevantes del sistema que requieren priorización, alineando así las decisiones técnicas con los objetivos estratégicos de Anfix.

Este enfoque se alinea con la metodología de Yin (1981), quien señala que los estudios de caso holísticos son útiles cuando se debe examinar un fenómeno en su contexto real, donde las fronteras entre el fenómeno y el contexto no están claramente definidas. En el

caso de Anfix, la aplicación del framework Tracy permitió priorizar la DT dentro de este contexto complejo, integrando tanto las perspectivas técnicas como las de negocio.

4.2. Descripción del caso

En esta sección, se presenta la preparación detallada del estudio de caso, utilizando la metodología seleccionada en el capítulo anterior. La preparación del estudio de caso incluye la definición clara del problema, la identificación de los datos necesarios, la selección de las técnicas de recolección de datos, y la planificación del análisis de los mismos. A través de esta preparación meticulosa, se pretende asegurar que el estudio de caso proporcione resultados significativos y aplicables a la gestión y priorización de la DT en el contexto específico de esta investigación.

Contexto de la empresa

Anfix es una empresa española que ofrece soluciones de contabilidad y gestión empresarial en la nube. Su plataforma ha sido diseñada para facilitar la automatización de procesos contables y financieros, permitiendo a las pequeñas y medianas empresas optimizar sus operaciones. Durante los últimos años, Anfix ha experimentado un rápido crecimiento, lo que ha traído consigo una acumulación significativa de DT debido a la necesidad de lanzar nuevas funcionalidades de forma acelerada y a la migración a arquitecturas basadas en microservicios. La acumulación de esta DT supone un riesgo para la calidad y la escalabilidad del sistema.

Problema y Objetivos

El principal desafío de Anfix es gestionar de manera eficaz la DT acumulada, lo cual impacta tanto en el rendimiento del software como en la capacidad de la empresa para mantenerse competitiva. Este caso de estudio se centra en la aplicación de una metodología de priorización de la DT previamente seleccionada, con el fin de optimizar el proceso de toma de decisiones técnicas y alinear dichas decisiones con los objetivos estratégicos de la empresa. El objetivo es garantizar que las áreas más críticas de la DT sean abordadas de manera prioritaria, optimizando los recursos y mejorando la calidad del producto final.

4.3. Metodología del caso de estudio

Método de investigación

El estudio de caso sigue un enfoque descriptivo y explicativo basado en la metodología seleccionada en el capítulo anterior. En este caso, se optó por utilizar el framework Tracy, que prioriza la DT en función de su impacto en los activos de TI y las fuentes de valor para el negocio. La metodología aplicada permite mapear las relaciones entre fuentes de

valor, activos de TI y elementos de configuración para determinar las prioridades de la DT de manera efectiva.

Criterios de selección del caso

Anfix fue seleccionada como caso de estudio por su relevancia en el sector de la tecnología empresarial, y por la importancia que tiene para la empresa la correcta gestión de su DT acumulada. La empresa proporciona un entorno adecuado para probar la metodología seleccionada, ya que presenta tanto un volumen importante de DT como una necesidad urgente de priorización para mantener su competitividad en el mercado.

4.4. Análisis del caso

Preparación del estudio de caso

La preparación del estudio de caso incluyó la definición clara del problema de la DT en Anfix, la identificación de los datos necesarios, la selección de técnicas de recolección de datos y la planificación del análisis. Este proceso meticuloso fue esencial para asegurar que el estudio de caso proporcionara resultados significativos y aplicables para la priorización de la DT en Anfix.

Unidad de análisis

La unidad de análisis en este estudio es el sistema completo de gestión empresarial de Anfix, compuesto por múltiples módulos y servicios interconectados. La aplicación del Framework Tracy se centra en priorizar la DT en los módulos más críticos para el negocio, basándose en el impacto en los activos de TI y las fuentes de valor del negocio.

4.5. Ejecución del caso de estudio

Aplicación del Framework Tracy

La aplicación del Framework Tracy en Anfix siguió una serie de pasos detallados que incluyeron la identificación, clasificación, evaluación y priorización de la DT. Este proceso permitió alinear las decisiones técnicas con los objetivos estratégicos del negocio, asegurando que las áreas críticas fueran abordadas de manera prioritaria.

Identificación de la DT

El primer paso crucial en la implementación del framework Tracy en Anfix es la identificación de la DT. Para llevar a cabo esta tarea, la empresa utiliza la herramienta Jira, una plataforma líder en la gestión de proyectos y seguimiento de incidencias. Jira permite a Anfix mantener un listado de backlogs donde se registran sistemáticamente

los diferentes problemas detectados tanto por los usuarios como por los empleados de la empresa.

Para evitar posibles sesgos en la identificación de la DT por parte de los diferentes stakeholders, se decidió comenzar con reuniones individuales. Estas reuniones permitieron que cada participante aportara su perspectiva sin verse influenciado por las opiniones de otros. El objetivo era recopilar una visión clara y completa de los problemas existentes en el sistema desde diferentes puntos de vista: el técnico y el de negocio.

Durante el proceso de identificación de la DT en Anfix, las tareas se llevaron a cabo en un periodo de dos semanas. En la primera semana, el Developer Manager, algunos miembros del equipo de desarrollo, la Product Manager y la CPO participaron en reuniones individuales para identificar las áreas del sistema afectadas, proporcionando detalles tanto de las incidencias registradas en Jira como de otras no documentadas. En la segunda semana, se realizaron reuniones conjuntas en las que se contrastaron y revisaron los datos recopilados, alcanzando un consenso sobre las partidas de DT identificadas, bajo la coordinación del Developer Manager, facilitando así la discusión para integrar las perspectivas técnicas y de negocio.

Como resultado de esta fase de identificación, se documentan detalladamente todos los errores, incidencias y aspectos del código que requieren refactorización. La información se recopila en Jira y se convierte en un recurso valioso para entender la magnitud y el impacto de la DT, permitiendo a los equipos de desarrollo y a los stakeholders tomar decisiones informadas para su gestión y resolución. El listado resultante de la fase de identificación se puede consultar en la Tabla A.1 del Anexo A. Se ha decidido trasladar estos detalles al Anexo para facilitar la lectura de este Capítulo.

Clasificación de la DT

Una vez identificada la DT, es fundamental clasificarla adecuadamente para priorizar las acciones correctivas y planificar los recursos necesarios de manera eficiente. La clasificación de la DT permite a los equipos de desarrollo y a los stakeholders comprender mejor las áreas críticas que requieren atención inmediata, así como aquellas que pueden ser abordadas a largo plazo.

En la Tabla 4.1 se muestra la cantidad de ocurrencias de cada tipo de DT en las diferentes partidas de DT identificadas. Es importante destacar que una partida de DT puede tener asociados varios tipos de DT, lo que explica por qué el número de ocurrencias de un tipo específico puede ser mayor que el número total de partidas. Esta información es fundamental para entender la concentración de DT en distintas áreas del proyecto y para tomar decisiones sobre las futuras acciones correctivas.

Tipo de DT	Cantidad de ocurrencias en partidas de DT
Mantenimiento	38
Rendimiento	28
Arquitectura	25
Funcionalidad	13
Usabilidad	6
Operaciones	5
Código	5
Dependencia	4
Seguridad	2
Documentación	1
Integración	1
Test	1

Tabla 4.1: Clasificación de los tipos de DT y la cantidad de ocurrencias

En la Figura 4.2, se presenta un diagrama de Pareto que muestra la cantidad de ocurrencias de cada tipo de DT en las diferentes partidas de DT identificadas en Anfix. Esta visualización permite observar claramente que los tipos de DT como mantenimiento, rendimiento y arquitectura concentran la mayoría de las ocurrencias. De acuerdo con el análisis, más del 70 % de las ocurrencias de DT están concentradas en estos tres tipos. El diagrama también permite identificar las áreas donde los esfuerzos de mitigación y refactorización deben enfocarse para maximizar la reducción de DT en el sistema.

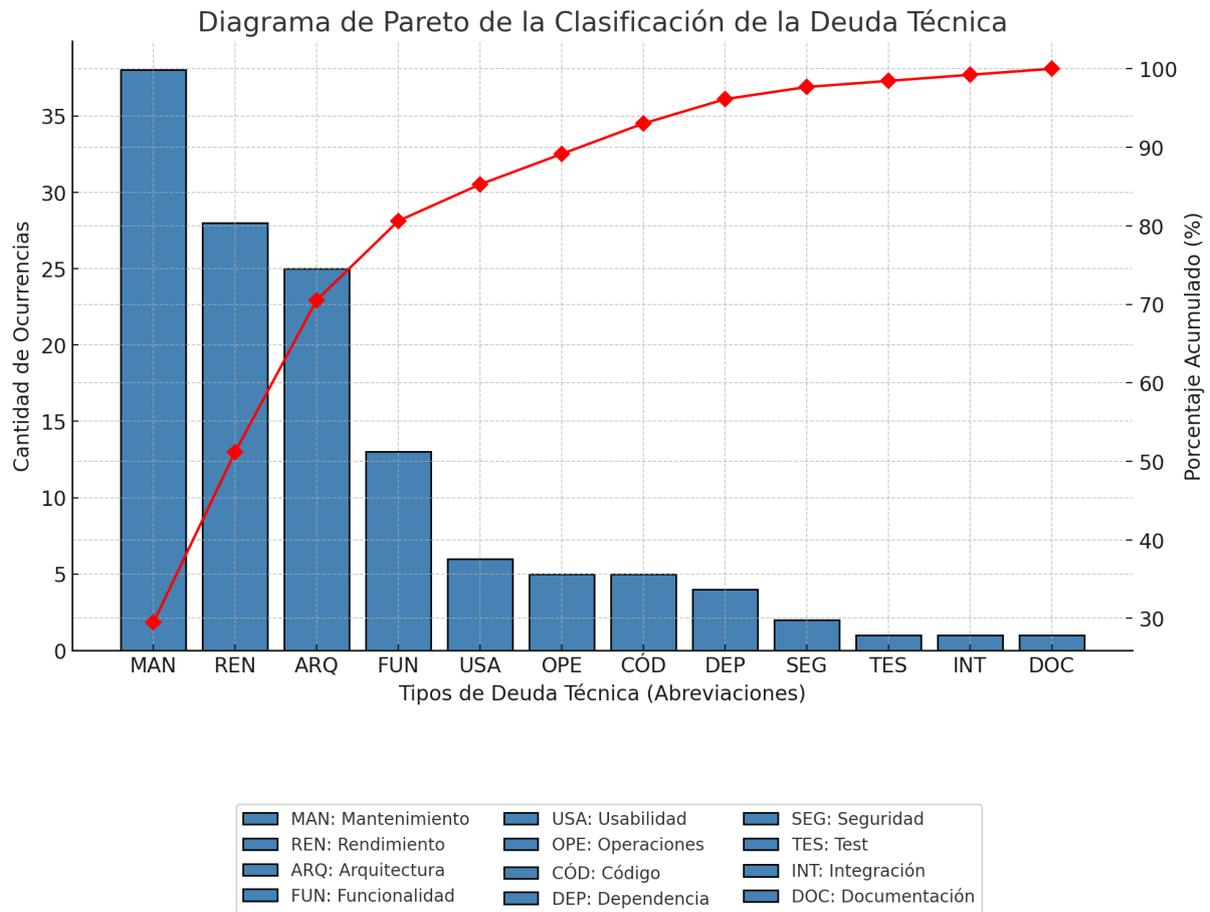


Figura 4.2: Clasificación de la DT

Esta combinación de información gráfica y tabular resulta clave para comprender la distribución de la DT en el proyecto y priorizar de manera eficiente las acciones correctivas en función de la severidad y frecuencia de la afectación.

Procesos de negocio identificados

Actualmente Anfix se estructura en torno a un conjunto de procesos de negocio esenciales que determinan su capacidad para competir en el mercado. Estos procesos se dividen generalmente en dos categorías principales: los procesos “core”, que son el núcleo de la creación de valor y constituyen las actividades fundamentales que distinguen a la empresa en su sector, y los procesos de soporte, que facilitan y optimizan el funcionamiento de los primeros. La correcta identificación, gestión y optimización de ambos tipos de procesos resulta crucial para asegurar la eficiencia operativa y la sostenibilidad a largo plazo de la organización. A continuación se exponen los procesos identificados:

- Procesos core:

- **Producción:** mantenimiento evolutivo y correctivo del producto.
 - **Ventas:** captación de clientes y venta del producto.
 - **Financiero:** gestión de recursos financieros y fiscales.
 - **RRHH:** gestión de recursos humanos.
- Procesos de soporte:
- **Marketing:** promoción del producto, captación y retención de clientes.
 - **Operaciones:** gestión de la infraestructura.
 - **Soporte:** atención al cliente, soporte técnico y formación.

Elementos de Configuración Identificados

Como parte del análisis del impacto de la DT en Anfix, es esencial identificar los elementos de configuración dentro de la infraestructura tecnológica del producto. Estos elementos representan los activos de TI clave que soportan tanto los procesos operativos como los estratégicos de la empresa. Siguiendo el enfoque propuesto en el framework Tracy, la identificación de estos elementos permite comprender cómo los subsistemas afectados por DT influyen directamente en los activos tecnológicos y, por ende, en los procesos de negocio que estos soportan. A continuación, se presenta una descripción detallada de los elementos de configuración identificados, seguida de un análisis que establece la relación entre estos elementos y las partidas de DT correspondientes.

- **Módulos Front-end del producto:** Son los componentes visuales que interactúan directamente con los usuarios finales, permitiendo la ejecución de procesos clave como la gestión de clientes y facturación. La DT en estos módulos puede impactar directamente la experiencia del usuario.
- **Módulos Back-end (API) del producto:** Soportan la lógica de negocio y el intercambio de datos entre las interfaces y la base de datos. La DT en estos módulos podría afectar la eficiencia de los procesos internos, lo que resulta en un impacto indirecto en los tiempos de respuesta del producto.
- **Bases de Datos MySQL y MongoDB:** Estas bases de datos almacenan la información estructurada y no estructurada que soporta el producto. La DT en este nivel puede comprometer la integridad de los datos y generar retrasos en los tiempos de consulta.
- **Módulos de las aplicaciones móviles (Android y iOS):** Estos módulos permiten a los usuarios interactuar con la plataforma desde dispositivos móviles. Dado que muchas interacciones de clientes se realizan a través de estas plataformas, la DT en este contexto podría afectar significativamente la satisfacción del usuario.

- **Servicios externos y extensiones de integración:** Estos componentes permiten que el producto se conecte y colabore con otras plataformas, servicios o aplicaciones externas. Su función principal es facilitar la interoperabilidad y extender las capacidades del sistema, integrándose con tecnologías de terceros que mejoran o complementan las funcionalidades del producto. La DT en estos servicios puede generar dificultades en la comunicación entre sistemas, provocando retrasos, fallos en la transferencia de datos o limitaciones en la escalabilidad y adaptabilidad del producto a nuevas tecnologías o mercados^{1 2}.

Una vez identificados los elementos de configuración, se realizó un análisis detallado para mapear las partidas DT a los elementos correspondientes. Este proceso es fundamental para priorizar adecuadamente cada partida, dado que cada elemento de configuración soporta un conjunto específico de procesos de negocio. De acuerdo con el framework Tracy, el impacto potencial de la DT en los activos de TI y su relación con los procesos de negocio es clave para la toma de decisiones. La Tabla 4.2 muestra cómo las partidas de DT están asociadas con los elementos de configuración, permitiendo una priorización basada en el impacto que estos tienen sobre los procesos de negocio.

Elemento de Configuración	Número de la partida de DT
Módulos Front-end del producto	DT-7, DT-10, DT-11, DT-12, DT-18, DT-31, DT-32, DT-36, DT-39, DT-42, DT-43
Módulos Back-end (API) del producto	DT-1, DT-2, DT-3, DT-4, DT-6, DT-8, DT-13, DT-14, DT-16, DT-19, DT-20, DT-21, DT-23, DT-25, DT-26, DT-27, DT-28, DT-29, DT-30, DT-34, DT-35, DT-37, DT-38, DT-40, DT-43, DT-44, DT-45
Bases de Datos MySQL que soporta el producto	DT-9, DT-15, DT-22, DT-24, DT-26, DT-33, DT-43
Servicios externos y extensiones de integración	DT-5, DT-17, DT-41

Tabla 4.2: Relación de elementos de configuración y números de DT

La tabla 4.3 representa una matriz de datos cruzados que relaciona los tipos de DT identificados en Anfix con los elementos de configuración afectados. Esta matriz permite visualizar la distribución de las diferentes partidas de DT a lo largo de los componentes clave del sistema, facilitando la identificación de áreas críticas que requieren atención prioritaria. La matriz está estructurada con las filas representando los tipos de DT identificados, como Mantenimiento, Rendimiento, Arquitectura y Funcionalidad, mientras que las columnas corresponden a los elementos de configuración del sistema en Anfix: Módulos Front-end

¹WooCommerce es una plataforma de comercio electrónico de código abierto construida sobre WordPress. Más información en: <https://woocommerce.com/>

²PrestaShop es una solución de comercio electrónico de código abierto para crear tiendas en línea. Más información en: <https://www.prestashop.com/>

(interacción con usuarios), Módulos Back-end (lógica de negocio), Bases de Datos (gestión de datos) y Servicios Externos (integraciones con terceros). Cada celda de la matriz muestra el número de ocurrencias de partidas de DT que afectan a un elemento de configuración específico.

Tipo de DT	Módulos Front-end	Módulos Back-end	Bases de Datos	Servicios externos
Mantenimiento	10	22	7	3
Rendimiento	6	19	3	1
Arquitectura	6	15	3	1
Funcionalidad	1	7	5	2
Usabilidad	5	0	0	1
Operaciones	0	5	0	0
Código	2	2	0	1
Dependencia	0	2	0	2
Seguridad	2	1	1	0
Documentación	0	1	0	0
Integración	1	0	0	0
Test	0	1	0	0

Tabla 4.3: Matriz de datos cruzados entre tipos de DT y elementos de configuración

Para complementar la matriz y facilitar la interpretación de los datos, se ha elaborado un gráfico de burbujas (ver Figura 4.3) que representa visualmente la relación entre los tipos de DT y los elementos de configuración, además de permitir identificar rápidamente las áreas con mayor acumulación de DT:

- **Eje X:** Elementos de configuración.
- **Eje Y:** Tipos de DT.
- **Tamaño de burbujas:** Indica el número de ocurrencias en cada combinación; burbujas más grandes representan una mayor cantidad de DT.

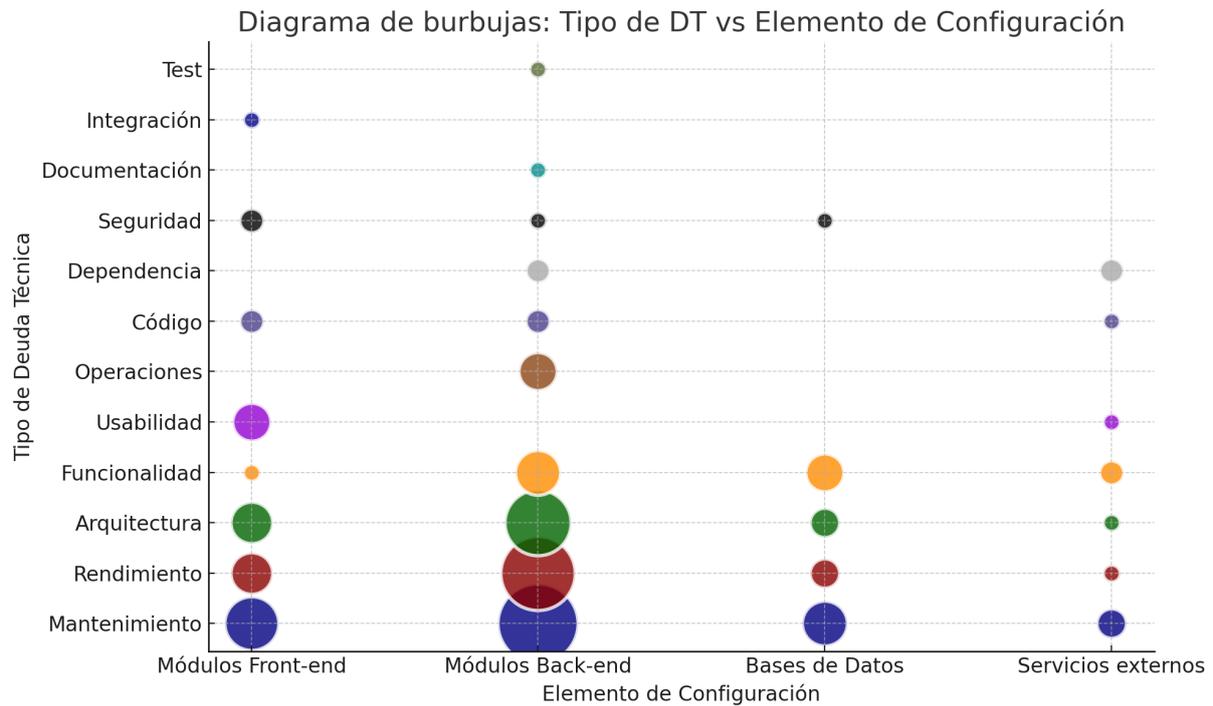


Figura 4.3: Diagrama de burbujas para representar visualmente los datos de la matriz cruzada (Tabla 4.3)

Al examinar la matriz y los datos del diagrama de burbujas, se pueden extraer varias observaciones claves:

1. Módulos Back-end:

- En los módulos back-end se registra una alta concentración de DT, siendo el mantenimiento el tipo más frecuente con 22 ocurrencias, lo que indica numerosos aspectos que requieren atención para evitar comprometer la estabilidad y capacidad de actualización del sistema. Además, existen 19 ocurrencias relacionadas con problemas de rendimiento, lo cual puede afectar negativamente la eficiencia y los tiempos de respuesta. Asimismo, 15 ocurrencias en arquitectura señalan desafíos que podrían limitar la escalabilidad y adaptabilidad futura del sistema.

2. Módulos de Front-end:

- Los módulos front-end también requieren mejoras significativas, ya que se identificaron 10 ocurrencias de mantenimiento que evidencian la necesidad de abordar problemas en la interfaz de usuario para garantizar una experiencia fluida. Adicionalmente, las 5 ocurrencias en usabilidad sugieren áreas donde la interfaz podría optimizarse para aumentar la satisfacción del usuario y facilitar el uso. Con 6 ocurrencias cada uno, los problemas de rendimiento y arquitectura

indican que hay aspectos del front-end que podrían mejorarse para optimizar la velocidad y la estructura del código.

3. Base de datos:

- En las bases de datos, la DT se manifiesta con 7 ocurrencias en mantenimiento y 5 en funcionalidad, lo que indica que existen cuestiones relacionadas con la integridad y eficiencia de los datos que deben ser atendidas. Además, con 3 ocurrencias tanto en arquitectura como en rendimiento, se sugiere que la estructura y el desempeño de las bases de datos son áreas que requieren mejoras para asegurar un funcionamiento óptimo.

4. Servicios externos y extensiones de integración:

- Aunque presentan un menor número de ocurrencias, es importante no subestimar su impacto, ya que problemas en estos elementos pueden afectar la interoperabilidad y la capacidad de ofrecer funcionalidades clave a los usuarios.

Activos de TI identificados

Los activos de TI juegan un papel clave en el soporte de los procesos de negocio de Anfix, ya que representan los sistemas, servicios y aplicaciones que generan valor para la empresa. De acuerdo con el enfoque del framework Tracy, estos activos de TI deben ser gestionados con el objetivo de maximizar su contribución al negocio, minimizando los riesgos asociados a la DT acumulada.

- **Jira:** Herramienta de gestión de proyectos que permite la planificación y el seguimiento de tareas. La DT en este activo podría afectar la eficiencia operativa del equipo de desarrollo, lo que impactaría en la capacidad para entregar nuevas funcionalidades o corregir errores a tiempo.
- **Confluence:** Plataforma de colaboración utilizada para la documentación y el intercambio de conocimiento. Una DT en Confluence podría ralentizar el acceso a información crítica, afectando la productividad y la toma de decisiones.
- **Anfix web y Anfix app:** Representan los puntos de acceso principales para los usuarios de Anfix, tanto en la web como en dispositivos móviles. Cualquier DT en estos activos puede comprometer la experiencia del usuario final, afectando la satisfacción del cliente y la retención.
- **AWS:** Infraestructura en la nube que soporta la escalabilidad y disponibilidad del producto. La DT en este activo podría generar problemas de rendimiento o disponibilidad, impactando directamente en la confiabilidad del servicio.
- **DataDog:** Servicio de monitoreo que permite visualizar el estado del sistema. La DT en este servicio podría derivar en una falta de visibilidad sobre el rendimiento del sistema, lo que afectaría la capacidad de respuesta ante incidentes.

- **Hubspot:** Herramienta de marketing y CRM que facilita la gestión de relaciones con los clientes. La DT aquí podría afectar la automatización y la calidad de las interacciones con los clientes, afectando potencialmente las ventas.
- **Stripe:** Plataforma de pagos que maneja transacciones financieras. La DT en este activo podría comprometer la seguridad y confiabilidad de las transacciones, afectando la confianza del cliente.
- **Factorial:** Sistema de gestión de recursos humanos. Una DT en Factorial podría impactar la eficiencia en la administración del personal y la gestión de nóminas.

Cada uno de estos activos de TI está compuesto por diversos elementos de configuración que, en conjunto, permiten su operación. De acuerdo con el framework Tracy, la DT que afecta a estos activos no solo impacta su operatividad directa, sino también la estabilidad y efectividad de los procesos de negocio que soportan.

Relación entre los procesos de negocio, activos de TI y elementos de configuración

El framework Tracy resalta la importancia de entender la relación entre los procesos de negocio, los activos de TI que los soportan, y los elementos de configuración que permiten su operatividad. En este sentido, es esencial analizar cómo los elementos de configuración influyen directamente en los activos de TI, y cómo estos activos, a su vez, soportan los procesos de negocio clave de la empresa. Esta comprensión nos permite priorizar de manera efectiva la DT, tomando en cuenta no solo el impacto técnico, sino también el impacto que tiene sobre el valor que los activos de TI aportan al negocio.

Siguiendo este enfoque, hemos identificado las interdependencias críticas entre los procesos de negocio de Anfix, los activos de TI que los soportan, y los elementos de configuración que subyacen en dichos activos. Estas relaciones se representan en el diagrama presentado en la Figura 4.4. Este diagrama es fundamental para visualizar cómo los elementos de configuración (módulos, bases de datos, servicios externos, etc.) impactan la funcionalidad de los activos de TI, y cómo, en última instancia, estos activos apoyan o impulsan los procesos de negocio.

Relación entre los elementos de configuración, activos de TI y procesos del negocio

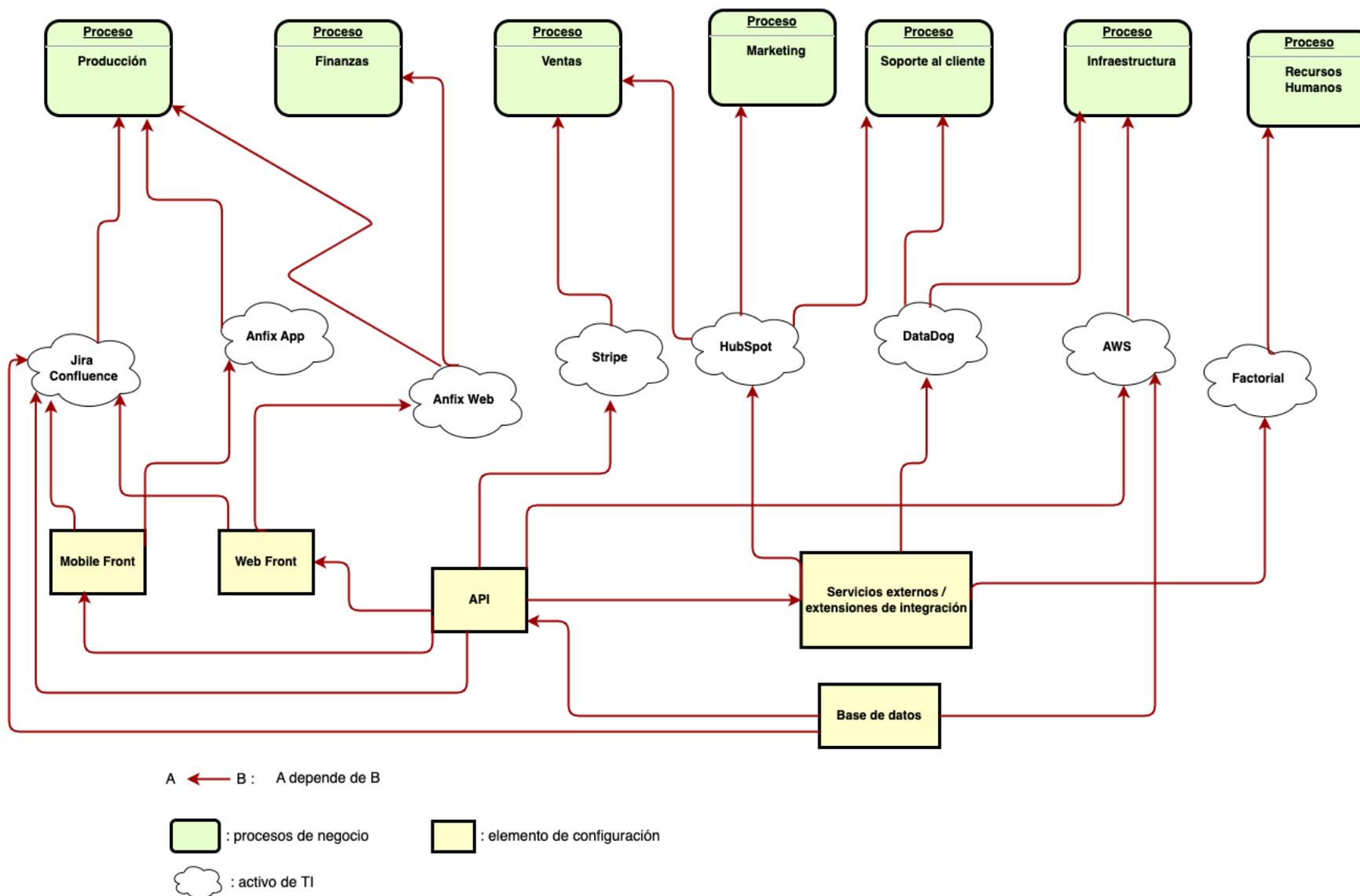


Figura 4.4: Relación entre los procesos de negocio, activos de TI y los elementos de configuración identificados en Anfix

Métricas empresariales identificadas para la priorización

Con el fin de entender las distintas visiones sobre las métricas que utiliza la empresa para abordar la priorización de la DT, se analizó el contexto en colaboración con el CEO, el CPO, el Developer Manager y el Product Manager. Para minimizar la influencia entre los participantes, se realizaron entrevistas individuales, solicitándoles que definieran las métricas para la priorización adaptada a su entorno específico (por ejemplo, productos, equipo, y factores de negocio). Las diversas métricas vistas fueron las siguientes:

Impacto a partir de la experiencia del cliente:

- Priorizar elementos que afecten directamente la experiencia y satisfacción del cliente.

Riesgo y Seguridad:

- Priorizar aquellas que presentan riesgos significativos para la seguridad y estabilidad del sistema.

Valor de Negocio:

- Enfocar en elementos que ofrecen el mayor retorno de inversión en términos de eficiencia y ahorro de costos.

Legalidad:

- Priorizar aquellas que aseguren el cumplimiento de leyes y regulaciones.

Escalabilidad:

- Enfocar en soluciones que permitan el crecimiento sostenido del sistema.

Costes operativos:

- Priorizar aquellas que influyan directamente en los costes operativos de la empresa.

Infraestructura:

- Considerar aquellas que afectan la infraestructura del sistema y de la empresa, además de impedir el uso correcto del mismo.

Rendimiento:

- Priorizar aquellas que afectan el rendimiento del sistema y de la empresa.

Innovación:

- Abordar elementos que permiten la implementación de nuevas tecnologías y funcionalidades.

Plantilla de priorización

En el contexto de Anfix, se identificó que la “Plantilla de Priorización” propuesta por el framework Tracy no cubría completamente las necesidades específicas de la empresa para la gestión de la DT. Esto se debe a que la mayoría de los activos de TI de Anfix están operativos (Figura 4.5 y Figura 4.6), excepto Anfix App, que se encuentra en fase de desarrollo. Por lo tanto, la categorización original basada en el estado de los activos (“operativos” vs. “operativos a futuro”) resultaba insuficiente para una priorización efectiva.

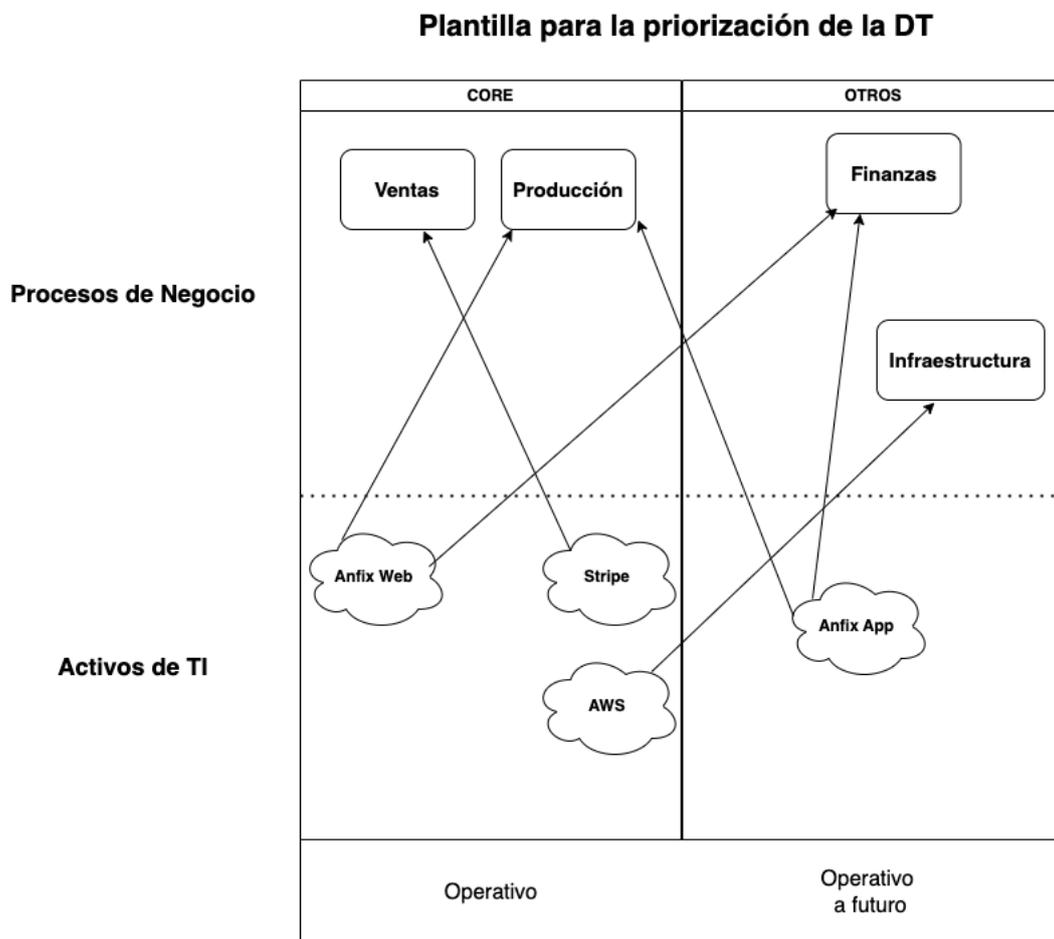


Figura 4.5: Plantilla de priorización de la DT en los procesos de Ventas, Producción, Finanzas e Infraestructura

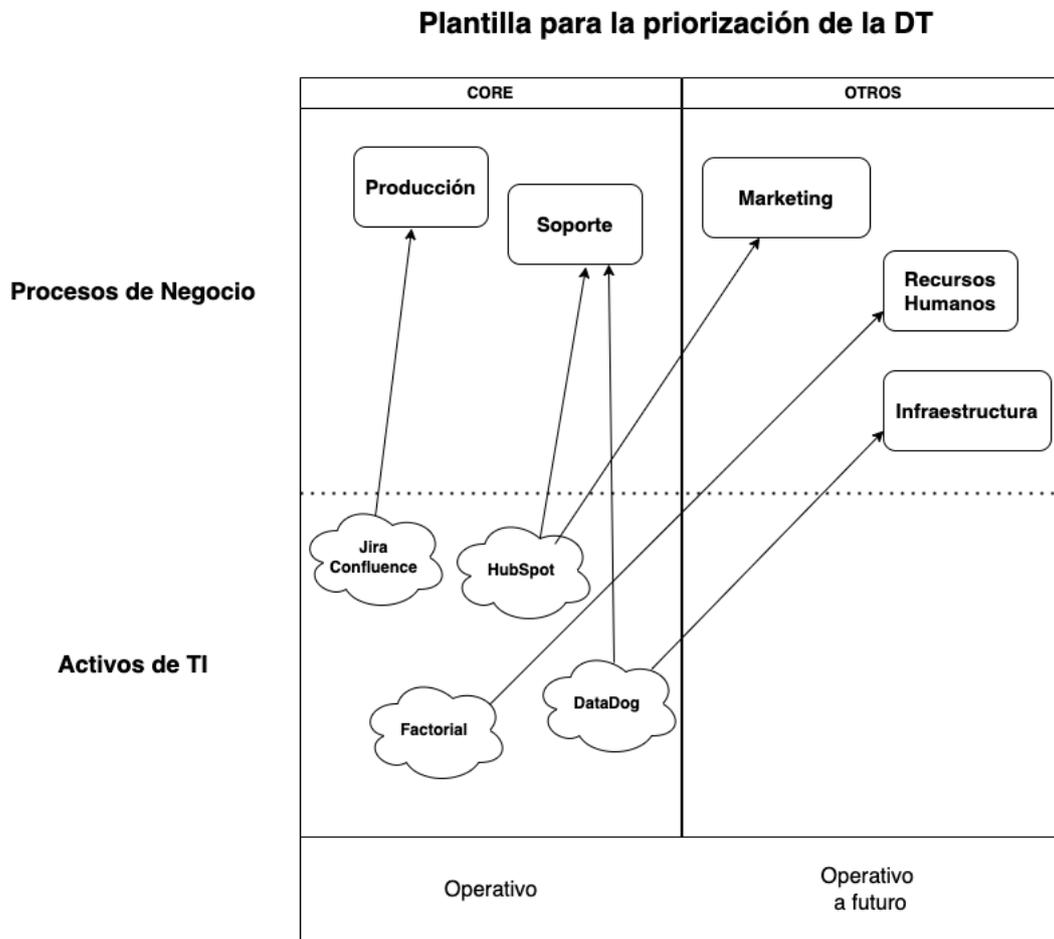


Figura 4.6: Plantilla de priorización de la DT en los procesos de Producción, Soporte, Marketing Recursos Humanos e Infraestructura

Para abordar esta limitación, Anfix decidió incorporar una nueva dimensión en la priorización de la DT: el impacto en el negocio. Esta decisión se fundamentó en:

- **La clasificación detallada de la DT vista anteriormente (Subsección 4.5)**
- **Las métricas empresariales identificadas, que reflejan las prioridades y objetivos estratégicos de la empresa (Subsección 4.5)**

A partir de estas métricas y la clasificación de la DT, se diseñó una nueva tabla de priorización (Tabla 4.4), que asigna prioridades a las partidas de DT en función de su impacto en el negocio. La prioridad se establece en una escala del 1 al 6, donde 1 representa la máxima prioridad y 6 la mínima. A continuación se explica detalladamente cada escala:

- **Prioridad 1 (Máxima):** Se asigna a las deudas que afectan críticamente la experiencia del cliente, presentan riesgos significativos de seguridad o pueden causar incumplimientos legales. Estas deudas son las más urgentes de abordar, ya que tienen el potencial de afectar severamente la continuidad del negocio y la reputación de la empresa.
- **Prioridad 2:** Se otorga a las deudas que impactan fuertemente el valor de negocio y el rendimiento del sistema. Abordarlas puede generar ahorros significativos y mejorar la eficiencia operativa.
- **Prioridad 3:** Incluye deudas que afectan la escalabilidad e infraestructura del sistema. Si bien no son tan urgentes como las anteriores, su mitigación/subsanación es importante para asegurar el crecimiento sostenible y evitar futuros problemas.
- **Prioridad 4:** Corresponde a deudas que influyen en los costes operativos. Abordarlas puede optimizar gastos, pero no representan un riesgo inmediato para el negocio.
- **Prioridad 5:** Se asigna a deudas que, aunque tienen un impacto bajo en el corto plazo, son relevantes para la innovación y la incorporación de nuevas tecnologías, manteniendo la competitividad de la empresa.
- **Prioridad 6 (Mínima):** Incluye deudas con bajo impacto general que pueden ser abordadas en el largo plazo sin afectar significativamente al negocio.

Relación	Tipo de proceso de negocio	Impacto en el negocio	Prioridad de la DT
(a)	Core	Alto	1 (máxima)
(b)	Otros	Alto	2
(c)	Core	Medio	3
(e)	Core	Bajo	4
(d)	Otros	Medio	5
(f)	Otros	Bajo	6 (mínima)

Tabla 4.4: Prioridad de la DT basada en la relación de los tipos de procesos de negocio y los estados de los activos de TI

Plantilla de valor empresarial

Luego de haber identificado las métricas que se tienen en cuenta en la empresa para abordar la priorización de la DT, pasaremos a completar la “plantilla de valor empresarial” que propone el framework Tracy (ver Figura 3.8), muestra cómo los procesos empresariales y los activos de TI se vinculan con métricas que pueden generar un impacto en el negocio en el corto, mediano o largo plazo. Esta plantilla permite a las partes interesadas identificar y clasificar el valor que los procesos y activos de TI aportan al negocio. En la Figura 4.7 se muestra cómo se ha utilizado esta plantilla con las métricas recopiladas en Anfix.

Plantilla del valor empresarial

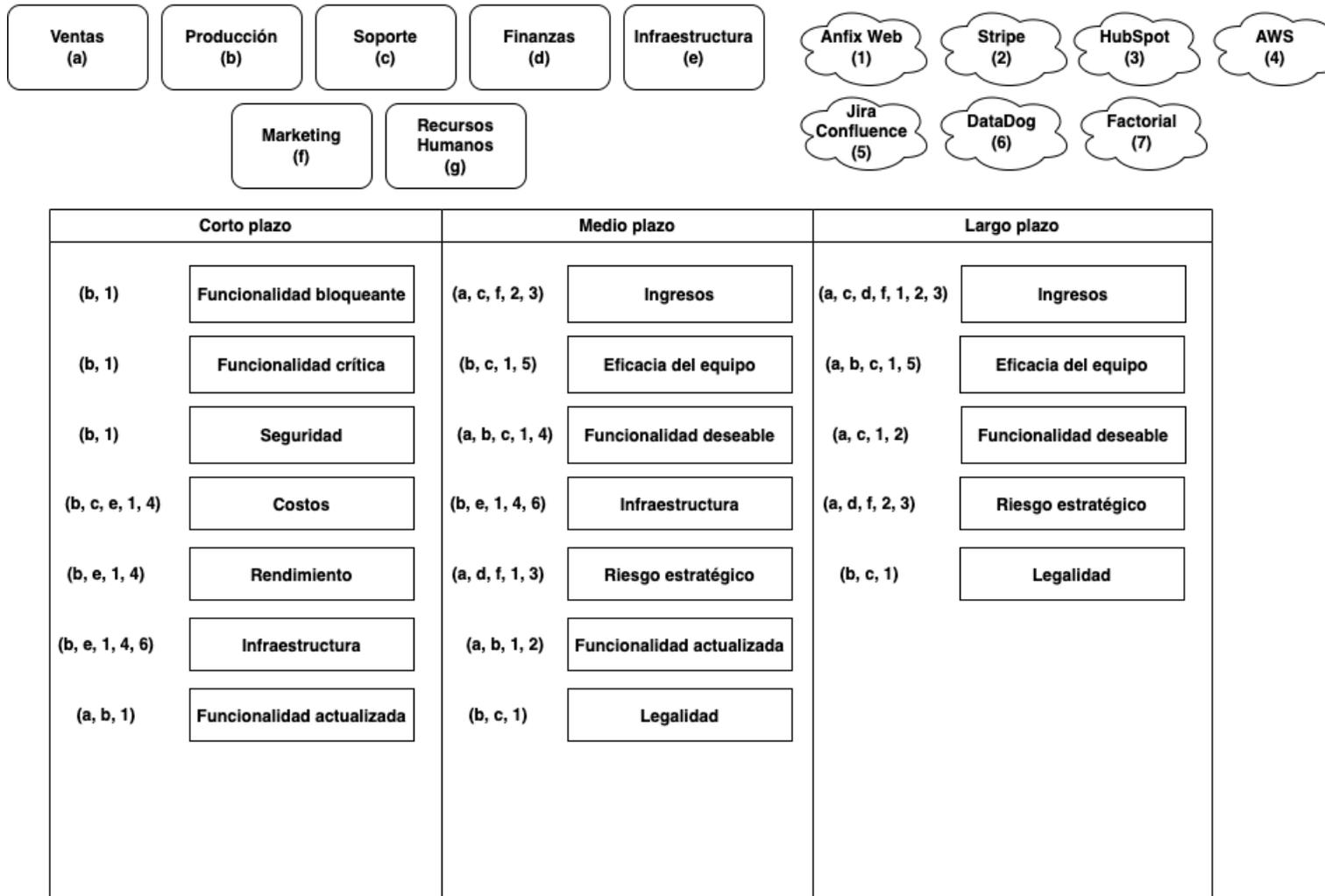


Figura 4.7: Plantilla de valor empresarial con los datos obtenidos de Anfix

El análisis de la plantilla de priorización incluyó un total de siete procesos de negocio (Ventas, Producción, Soporte, Finanzas, Infraestructura, Marketing y Recursos Humanos) y siete activos de TI (Anfix Web, Stripe, HubSpot, AWS, Jira Confluence, Datadog y Factorial). Esta cobertura exhaustiva garantiza que se consideren todos los elementos clave que impactan el rendimiento y crecimiento de Anfix.

En el corto plazo, destacan métricas críticas para la estabilidad y el funcionamiento inmediato de la empresa. El proceso de Producción y el activo Anfix Web desempeñan un papel crucial en la funcionalidad crítica y la seguridad, asegurando que las operaciones diarias se mantengan estables y eficientes sin interrupciones que puedan comprometer la entrega de servicios a los clientes. Además, aspectos como costos y rendimiento vinculados tanto a Producción como a Infraestructura reflejan la importancia de una gestión eficiente de recursos en esta etapa inicial.

En el mediano plazo, las métricas clave se orientan hacia la generación de ingresos y la eficacia del equipo. Procesos como Ventas, Soporte y Marketing, respaldados por activos de TI como HubSpot y Anfix Web, son fundamentales para mejorar la interacción con los clientes y optimizar los procesos comerciales. También se observa un enfoque en la infraestructura y en la incorporación de funcionalidades deseables, preparando a la empresa para un crecimiento sostenido mediante la implementación de mejoras tecnológicas y la optimización de procesos.

En el largo plazo, las métricas de impacto se centran en la eficacia del equipo, el riesgo estratégico y el cumplimiento legal, subrayando la importancia de la sostenibilidad y la conformidad normativa. Procesos como Finanzas, Recursos Humanos e Infraestructura, junto con activos como AWS y Jira Confluence, juegan un papel crucial en la preservación de la estabilidad del negocio, minimizando riesgos a largo plazo y asegurando el cumplimiento de las normativas necesarias para la operación continua de la empresa.

4.6. Priorización de la DT

En este apartado, se presenta la priorización de las diferentes partidas de DT a partir de la relación entre los activos de TI, los procesos de negocio a los que dan soporte, y el impacto potencial en el negocio. Los datos que se presentan en las tablas reflejan tanto los resultados obtenidos en la investigación realizada en la empresa Anfix como la aplicación práctica del Framework Tracy.

Criterios de agrupación y estructuración de las tablas

Para facilitar el análisis y minimizar posibles sesgos en la asignación de prioridades, se decidió agrupar las partidas de DT en cuatro tablas, donde cada tabla incluye partidas de DT pertenecientes a diferentes elementos de configuración. Es decir, cada tabla presenta una partida de DT que afecta a los Módulos Back-end, otra a los Módulos Front-end, otra a las Bases de Datos y otra a los Servicios Externos. Esta estrategia permite:

- Analizar las partidas de DT de manera equilibrada, asegurando que se consideren todas las áreas críticas del sistema en cada etapa del análisis.
- Evitar la concentración de prioridades en un solo tipo de elemento de configuración, lo que podría ocurrir si se agruparan todas las partidas de un mismo tipo en una sola tabla.
- Facilitar la comparación y priorización cruzada de las partidas de DT que afectan a diferentes componentes del sistema, considerando su impacto relativo en el negocio.

Las partidas de DT se seleccionaron para cada tabla con el objetivo de representar adecuadamente los diferentes elementos de configuración y reflejar una variedad de impactos potenciales en el negocio. Esto permite una evaluación integral de la DT en el sistema completo, promoviendo una asignación de prioridades más objetiva y alineada con los objetivos estratégicos de Anfix.

Presentación de las tablas de priorización

A continuación, se presentan las Tablas 4.5, 4.6, 4.7 y 4.8 que sintetizan la priorización de las partidas de DT en función de tres criterios: el activo de TI afectado, el proceso de negocio asociado y el impacto potencial en diferentes plazos.

DT	Activo TI	Proceso de negocio	Prioridad	Impacto potencial en el negocio		
				corto plazo	medio plazo	largo plazo
DT-9	Anfix web, Anfix app, AWS	Producción, Infraestructura, Ventas	1	Costos, Rendimiento, Infraestructura	Ingresos, Riesgo Estratégico	-
DT-4	Anfix web, Anfix app y AWS	Producción, Infraestructura, Ventas	3	Rendimiento, Infraestructura	Ingresos	-
DT-7	Anfix web	Producción y Ventas	4	Rendimiento, Experiencia de Usuario	Ingresos	-
DT-5	Anfix web (Servicio externo)	Producción, Ventas	4	Funcionalidad bloqueante	Ingresos	-

Tabla 4.5: Partida de DT seleccionadas para su priorización con respecto a su impacto potencial en el negocio (Grupo 1)

DT	Activo TI	Proceso de negocio	Prioridad	Impacto potencial en el negocio		
				corto plazo	medio plazo	largo plazo
DT-15	Anfix web, Anfix app, AWS	Producción, Ventas, Infraestructura	1	Costos, Infraestructura, Rendimiento	Ingresos	-
DT-18	Anfix web	Producción, Ventas	3	Costos, Rendimiento	Ingresos	-
DT-17	Anfix web, AWS	Producción, Infraestructura, Ventas	4	Seguridad, Infraestructura, Costos	Ingresos	-
DT-13	Confluence, Anfix web	Producción	6	-	Costos, Eficacia del equipo	-

Tabla 4.6: Partida de DT seleccionadas para su priorización con respecto a su impacto potencial en el negocio (Grupo 2)

DT	Activo TI	Proceso de negocio	Prioridad	Impacto potencial en el negocio		
				corto plazo	medio plazo	largo plazo
DT-24	Anfix web, HubSpot	Producción, Soporte, Finanzas	3	Costos, Infraestructura	Ingresos, Funcionalidad deseable	-
DT-32	Anfix web	Producción, Soporte	3	Costos, Rendimiento, Infraestructura	Ingresos, Funcionalidad deseable	-
DT-41	Anfix web	Producción, Infraestructura	4	Rendimiento, Infraestructura	Funcionalidad deseable	-
DT-19	Anfix web, AWS	Producción, Infraestructura	5	Infraestructura	Eficacia del equipo	-

Tabla 4.7: Partida de DT seleccionadas para su priorización con respecto a su impacto potencial en el negocio (Grupo 3)

DT	Activo TI	Proceso de negocio	Prioridad	Impacto potencial en el negocio		
				corto plazo	medio plazo	largo plazo
DT-43	Anfix web, Anfix app, AWS	Producción, Infraestructura, Soporte	1	Seguridad, Costos, Rendimiento, Infraestructura	Ingresos	-
DT-33	Anfix web, Anfix app	Producción, Infraestructura, Soporte	3	Costos, Rendimiento, Infraestructura	Ingresos	-
DT-22	Anfix web, Anfix app	Producción, Infraestructura, Soporte	3	Costos, Rendimiento, Infraestructura	Ingresos	-
DT-39	Anfix web	Producción	4	Rendimiento	-	-
DT-34	Anfix web, AWS	Producción, Infraestructura	4	Rendimiento	Infraestructura	-
DT-38	AWS	Infraestructura	5	Rendimiento	Infraestructura	-

Tabla 4.8: Partida de DT seleccionadas para su priorización con respecto a su impacto potencial en el negocio (Grupo 4)

La priorización realizada nos permite focalizar los esfuerzos en las áreas más críticas, garantizando que las decisiones tomadas contribuyan al rendimiento, la seguridad y la sostenibilidad del sistema. Al considerar factores como el impacto en el negocio, las métricas empresariales y los horizontes temporales, hemos obtenido una visión integral que facilita la gestión efectiva de la DT.

En el próximo capítulo, abordaremos la discusión de los resultados obtenidos, analizando en profundidad las implicaciones de los hallazgos y evaluando cómo las estrategias propuestas pueden influir en la evolución futura de Anfix. Esta reflexión nos permitirá validar la efectividad del enfoque adoptado y considerar posibles ajustes para mejorar aún más la gestión de la DT en la organización.

Capítulo 5

Resultados y Discusión

En este capítulo se presentan los resultados obtenidos tras la aplicación del Framework Tracy en el caso de estudio de Anfix, una empresa tecnológica dedicada a ofrecer soluciones de contabilidad y gestión empresarial en la nube. A través de la priorización de la DT basada en el impacto entre la relación de los activos de TI, los procesos del negocio y las métricas de importancia para la empresa, se identificaron, clasificaron y priorizaron las partidas de DT más críticas. Los resultados obtenidos se analizan en función de los objetivos estratégicos de la empresa, así como las implicaciones prácticas para la gestión de la DT. Además, se discuten los principales desafíos encontrados durante el proceso de priorización y las lecciones aprendidas.

5.1. Identificación y clasificación de la DT

El primer paso en la aplicación del Framework Tracy fue la identificación de las partidas de DT acumuladas en Anfix, utilizando la herramienta **Jira** para registrar las incidencias y áreas críticas que requerían atención. Como se detalló en el capítulo anterior, se identificaron **45 partidas de DT**, clasificados según su impacto en los activos de TI y los procesos de negocio

El análisis inicial de las incidencias permitió categorizar cada deuda en función de su impacto, urgencia, esfuerzo necesario y origen, ya fuera por causas intencionales o no intencionales. De las 45 incidencias, la mayoría correspondían a causas no intencionales, indicando que gran parte de la DT surgió debido a la acumulación progresiva de problemas técnicos relacionados con la evolución del sistema y la migración a una arquitectura de microservicios.

La **Tabla A.1** y las gráficas (Figuras 4.1 y 4.2) proporcionan un análisis detallado sobre la distribución de los diferentes tipos de DT identificados.

Mantenimiento (38 ocurrencias)

El mayor número de incidencias está relacionado con problemas de **mantenimiento**. Esto sugiere que la acumulación de DT afecta considerablemente la capacidad de la empresa para mantener el sistema en funcionamiento eficiente. Por ejemplo, se identificaron módulos con código obsoleto que dificultaban la implementación de nuevas funcionalidades y requerían un esfuerzo considerable para su actualización.

Rendimiento (28 ocurrencias)

El **rendimiento** es el segundo tipo de DT más frecuente. Dado que Anfix opera en un entorno SaaS (Software as a Service), cualquier degradación en el rendimiento del sistema impacta directamente la experiencia del usuario. Un ejemplo específico fue la detección de consultas ineficientes a la base de datos que incrementaban los tiempos de carga de ciertas funcionalidades clave.

Arquitectura (25 ocurrencias)

Las incidencias relacionadas con la **arquitectura** reflejan la complejidad de la transición hacia microservicios y su impacto en la estructura subyacente del sistema. Se encontraron dependencias circulares entre servicios que dificultaban el despliegue independiente y afectaban la escalabilidad de la plataforma.

Funcionalidad y Usabilidad (13 y 6 ocurrencias respectivamente)

Aunque estos tipos de DT son menos frecuentes, son críticos para el usuario final. Los problemas de **funcionalidad** incluyen fallos en características esenciales, como la generación de informes financieros. Los problemas de **usabilidad** afectan la eficiencia con la que los usuarios realizan sus tareas, como interfaces confusas o flujos de trabajo poco intuitivos.

5.2. Priorización de la DT

La priorización de las partidas de DT se ha estructurado en cuatro tablas que agrupan las deudas según los diferentes elementos de configuración afectados. Esta estrategia ha permitido un análisis equilibrado y una asignación de prioridades más objetiva, evitando la concentración de esfuerzos en una sola área y facilitando la comparación cruzada entre diferentes componentes del sistema.

Cada tabla presenta una selección representativa de las partidas de DT, asignándoles una prioridad en una escala del 1 al 6. Esta escala refleja la urgencia y el impacto de cada deuda en el negocio, permitiendo focalizar los recursos en las áreas más críticas y alineadas con las necesidades estratégicas de la empresa.

Análisis de la priorización

Al revisar los datos presentados en las cuatro tablas de priorización, se identifican un total de 18 partidas de DT distribuidas en diferentes niveles de prioridad. A continuación, se detalla la cantidad identificada en cada valor de la escala de prioridades y su representación:

Prioridad 1 (Máxima)

- **Cantidad de partidas:** 3 (DT-9, DT-15 y DT-43)
- **Representación:** Las partidas de DT con prioridad 1 son aquellas que afectan críticamente la experiencia del cliente, presentan riesgos significativos de seguridad o pueden causar incumplimientos legales. En el caso de Anfix, DT-9 y DT-15 impactan directamente en procesos clave como Producción, Infraestructura y Ventas, además de involucrar activos de TI esenciales como Anfix web, Anfix app y AWS. DT-43 también se clasifica en esta categoría debido a su impacto en la seguridad, costos, rendimiento e infraestructura. Estas deudas son urgentes de abordar para evitar compromisos en la continuidad del negocio y la reputación de la empresa.

Prioridad 2

- **Cantidad de partidas:** 0
- **Representación:** En las tablas presentadas, no se identificaron partidas de DT asignadas a la prioridad 2. Esta ausencia puede indicar que, dentro del alcance del estudio, las deudas que impactan fuertemente el valor de negocio y el rendimiento del sistema han sido adecuadamente categorizadas en otras prioridades, principalmente en la máxima prioridad o en niveles inferiores según su impacto específico.

Prioridad 3

- **Cantidad de partidas:** 6 (DT-4, DT-18, DT-24, DT-32, DT-33, DT-22).
- **Representación:** Las partidas con prioridad 3 afectan la escalabilidad e infraestructura del sistema. Estas deudas son importantes para asegurar el crecimiento sostenible y evitar futuros problemas, aunque no son tan urgentes como las de prioridad 1. En Anfix, estas deudas impactan en activos de TI como Anfix web, Anfix app, AWS, HubSpot y Confluence, influyendo en procesos como Producción, Infraestructura y Ventas. Abordarlas contribuirá significativamente a la eficiencia operativa y al soporte de objetivos a mediano plazo.

Prioridad 4

- **Cantidad de partidas:** 6 (DT-7, DT-5, DT-17, DT-41, DT-39, DT-34).

- **Representación:** Las partidas de prioridad 4 influyen en los costes operativos y, en algunos casos, en aspectos como la seguridad e infraestructura. Aunque no representan un riesgo inmediato, optimizar estos elementos puede reducir gastos y mejorar la operatividad del sistema. En Anfix, estas deudas afectan a diversos activos de TI como Anfix web, AWS y Confluence, y están asociadas a procesos de Producción, Ventas e Infraestructura, lo que subraya la importancia de gestionarlas para mantener la eficiencia y reducir costes a largo plazo.

Prioridad 5

- **Cantidad de partidas:** 2 (DT-19 y DT-38)
- **Representación:** Las deudas de prioridad 5 tienen un impacto medio en el negocio, enfocándose en la eficacia del equipo y aspectos relacionados con la infraestructura. Estas deudas son relevantes para la optimización de procesos internos y la mejora de la productividad, aunque no representan un riesgo crítico. En Anfix, DT-19 y DT-38 impactan en activos de TI como Anfix web y AWS, afectando procesos de Producción e Infraestructura, lo que indica la necesidad de gestionarlas para mejorar la eficacia operativa y preparar el terreno para futuros desarrollos.

Prioridad 6 (Mínima)

- **Cantidad de partidas:** 1 (DT-13)
- **Representación:** Las partidas de prioridad 6 tienen un bajo impacto general y pueden ser abordadas a largo plazo sin afectar significativamente al negocio. En el caso de Anfix, DT-13 está asociada a Confluence y afecta procesos de Producción, con impacto en la eficacia del equipo. Aunque estas deudas no son urgentes, su resolución contribuirá a la mejora continua y a la optimización de recursos (tiempo de adaptación para las nuevas incorporaciones) en el futuro.

Esta distribución refleja una focalización significativa en las prioridades 3 y 4, que abarcan deudas relacionadas con la escalabilidad, infraestructura y costes operativos. La asignación de tres partidas a la prioridad 1 resalta la atención hacia las áreas más críticas que afectan directamente la estabilidad y seguridad del negocio. La ausencia de partidas en la prioridad 2 y la presencia de una sola partida en la prioridad 6 indican un equilibrio en la gestión de la DT, priorizando adecuadamente las necesidades inmediatas y futuras sin descuidar aspectos menos críticos pero igualmente importantes para la sostenibilidad y crecimiento de Anfix.

5.3. Impacto en los Procesos de Negocio y Activos de TI

Impacto cuantitativo

La ejecución de las soluciones priorizadas tuvo un impacto medible en los procesos de negocio core y en la infraestructura de TI de Anfix. Se utilizaron herramientas de monitoreo como **New Relic** y **Google Analytics** para recopilar datos antes y después de las intervenciones, comparando periodos de tres meses.

1. **Rendimiento del sistema:** Tras la corrección de problemas en los módulos de back-end y bases de datos, se observó una mejora del **30 % en los tiempos de respuesta**. Esto redujo los tiempos de carga de 2 segundos a 1.4 segundos en promedio, mejorando la capacidad de Anfix para manejar un 25 % más de usuarios simultáneos y disminuyendo en un **15 % las incidencias de soporte relacionadas con el rendimiento**.
2. **Optimización de la infraestructura:** La solución de DT en la arquitectura y la infraestructura de **AWS** permitió una mejor escalabilidad del sistema, reduciendo los costos operativos en un **10 %**. Esto se logró mediante la optimización de instancias y el uso eficiente de recursos en la nube.
3. **Experiencia del usuario:** La resolución de problemas de usabilidad en el front-end y en las aplicaciones móviles mejoró la tasa de conversión en un **5 %**, resultando en un aumento del **8 % en la retención de usuarios** y una reducción del **10 % en la tasa de abandono**.
4. **Seguridad del sistema:** Las correcciones en áreas de seguridad resultaron en una reducción del **40 % en incidentes de seguridad** reportados, incrementando la confianza del cliente en la plataforma.
5. **Eficiencia operativa:** El enfoque en problemas de mantenimiento permitió una reducción del **20 % en el tiempo invertido por el equipo de desarrollo en tareas de corrección de errores**, liberando recursos para proyectos estratégicos de mayor valor añadido, como el desarrollo de nuevas funcionalidades.

Impacto cualitativo

Alineación con los Objetivos Estratégicos

La priorización de la DT permitió alinear las decisiones técnicas con la **estrategia de crecimiento** de Anfix. Al enfocarse en las incidencias que afectaban los procesos core, se garantizó que las áreas críticas como producción y ventas se mantuvieran ágiles y adaptables. Esto facilitó el lanzamiento de nuevas funcionalidades en respuesta a las necesidades del mercado.

Experiencia del Cliente y Competitividad

La mejora en el rendimiento y la usabilidad del sistema elevó la **satisfacción del cliente**. Los usuarios reportaron una experiencia más fluida y eficiente, lo que fortaleció la posición competitiva de Anfix. En un mercado con alta competencia, ofrecer un producto estable y de alta calidad es esencial para la retención y adquisición de clientes.

Gestión del Riesgo y Seguridad

Abordar la DT relacionada con la seguridad fue fundamental para mitigar **riesgos operacionales** y garantizar la confiabilidad del sistema. Esto fue especialmente importante dado que Anfix maneja información financiera sensible. La mejora en la seguridad también facilitó el cumplimiento de regulaciones y estándares de la industria.

Percepción de los Stakeholders

Las entrevistas realizadas con los stakeholders clave revelaron percepciones positivas sobre el manejo de la DT:

- **CEO:** “La priorización de la deuda técnica nos ha permitido enfocar recursos en áreas que realmente impulsan nuestro crecimiento y mejoran la satisfacción del cliente.”
- **CPO:** “Ahora tenemos una visión clara de cómo nuestras decisiones técnicas impactan directamente en los objetivos de negocio. Esto ha mejorado la colaboración entre equipos.”
- **Developer Manager:** “El equipo de desarrollo trabaja de manera más eficiente, centrado en resolver problemas críticos y contribuyendo a proyectos innovadores.”
- **Product Manager:** “La reducción de la DT ha acelerado el lanzamiento de nuevas funcionalidades, manteniéndonos competitivos en el mercado.”

5.4. Discusión

Beneficios y Desafíos de la Priorización

La priorización de la DT en Anfix ha sido un proceso esencial para gestionar de manera eficiente las incidencias identificadas, alineando las acciones técnicas con los objetivos estratégicos de la empresa. A continuación, se analizan en profundidad los beneficios y desafíos asociados a este proceso de priorización.

Beneficios

- **Enfoque Estratégico:** La priorización permite a Anfix centrar sus esfuerzos en las partidas de DT que tienen un mayor impacto en los procesos de negocio y en los activos de TI. Esto asegura que los recursos limitados se asignen a las áreas que realmente impulsan el valor empresarial, mejorando la alineación entre las acciones técnicas y los objetivos estratégicos de la empresa.
- **Optimización de Recursos:** Al priorizar las partidas de DT, se garantiza una utilización más eficiente de los recursos humanos y temporales. Las partidas de mayor prioridad reciben atención inmediata, lo que evita la dispersión de esfuerzos en áreas de menor impacto y mejora la productividad del equipo de desarrollo.
- **Mejora en la Toma de Decisiones:** La priorización proporciona una base objetiva para la toma de decisiones, basada en datos cuantitativos y cualitativos recopilados a través de herramientas como JIRA. Esto reduce la subjetividad y facilita la justificación de las prioridades establecidas ante los diferentes stakeholders.
- **Reducción de Riesgos:** Al identificar y priorizar las partidas de DT que representan mayores riesgos operacionales o de seguridad, Anfix puede mitigar proactivamente estos riesgos, evitando potenciales interrupciones en el servicio y protegiendo la reputación de la empresa.
- **Planificación a Largo Plazo:** La priorización permite establecer un roadmap claro para el abordaje de las partidas DT, facilitando la planificación de proyectos futuros y asegurando que las mejoras tecnológicas se implementen de manera ordenada y sostenible.
- **Transparencia y Comunicación:** Utilizando herramientas como JIRA para gestionar y priorizar la DT, se mejora la transparencia del proceso, facilitando la comunicación entre los equipos técnicos y los stakeholders de negocio. Esto fomenta una comprensión compartida de las prioridades y promueve la colaboración interdepartamental.

Desafíos

- **Definición de Criterios de Priorización:** Establecer criterios claros y consensuados para la priorización puede ser complejo, ya que requiere la alineación de diferentes perspectivas y objetivos dentro de la organización. Es fundamental definir métricas que reflejen adecuadamente el impacto de cada DT en el negocio.
- **Consenso entre Stakeholders:** Lograr que todos los stakeholders estén de acuerdo con las prioridades establecidas puede ser un desafío, especialmente cuando existen diferencias en la percepción del impacto de cada DT. Es necesario facilitar la comunicación y negociar prioridades de manera efectiva.

- **Actualización Continua:** La DT es dinámica y puede evolucionar con el tiempo. Mantener la priorización actualizada requiere un monitoreo constante y la capacidad de adaptarse a cambios en el entorno empresarial y tecnológico, lo que implica un esfuerzo continuo por parte del equipo.
- **Cuantificación Precisa del Impacto:** Asociar de manera precisa cada DT con su impacto en los KPI empresariales puede ser complicado debido a la interdependencia de los sistemas y procesos. Es esencial desarrollar métodos robustos para medir y valorar el impacto de cada DT.
- **Resistencia al Cambio:** Implementar un proceso de priorización estructurado puede encontrar resistencia por parte de equipos acostumbrados a enfoques menos sistemáticos. Es necesario gestionar el cambio organizacional mediante capacitación y demostrando los beneficios tangibles de la priorización.
- **Limitaciones de Herramientas de Gestión:** Aunque herramientas como JIRA facilitan la identificación y clasificación de partidas de DT, la priorización efectiva puede requerir complementos o metodologías adicionales para integrar datos cuantitativos y cualitativos de manera más precisa.

Valoración y Cuantificación de la Priorización

La priorización de la DT en Anfix ha permitido una valoración y cuantificación más precisa de cada partida identificada. Utilizando las incidencias registradas en JIRA, se ha podido asociar cada DT con métricas empresariales específicas, lo que facilita una evaluación basada en datos objetivos. Este enfoque ha permitido:

- **Asignación de Prioridades Basada en Impacto:** Cada partida de DT ha sido evaluada en función de su impacto en los procesos de negocio y en los activos de TI, asignándole una prioridad en una escala del 1 al 6. Esta asignación se ha realizado considerando factores como la experiencia del cliente, riesgos de seguridad, valor de negocio y escalabilidad.
- **Estimación de Recursos y Tiempos:** Al priorizar las DT, se ha facilitado la estimación de los recursos y tiempos necesarios para abordar cada deuda. Esto permite una planificación más eficiente y la asignación de personal de manera óptima, asegurando que las partidas de mayor prioridad reciban la atención necesaria sin comprometer otros proyectos.
- **Monitoreo y Seguimiento:** La utilización de JIRA ha permitido un seguimiento detallado del progreso en la priorización y resolución de las DT. Se ha podido monitorear el avance de cada prioridad, ajustando las estrategias según sea necesario para mantener el enfoque en las áreas más críticas.
- **Evaluación de Impacto:** La priorización ha permitido evaluar de manera continua el impacto de las acciones tomadas sobre los KPI empresariales, facilitando ajustes

en la estrategia de gestión de DT para maximizar los beneficios y minimizar los riesgos.

Capítulo 6

Conclusiones y trabajos futuros

6.1. Conclusiones generales del trabajo

El desarrollo de este Trabajo de Fin de Máster ha permitido aplicar con éxito una metodología para la priorización de la DT en la empresa Anfix, cumpliendo así con los objetivos propuestos inicialmente. La aplicación de esta metodología ha sido esencial para mejorar la eficiencia operativa y la capacidad de la empresa para alinear sus decisiones técnicas con sus estrategias comerciales. Además, el proceso de priorización ha tenido un impacto positivo en la colaboración entre los stakeholders, promoviendo una mayor cohesión entre los equipos técnicos y de negocio. Esto ha facilitado una toma de decisiones más consensuada y estratégica, lo que contribuyó de manera significativa a la sostenibilidad, competitividad e innovación de la empresa.

6.2. Revisión de la literatura

La revisión de la literatura fue un componente crucial de este trabajo, ya que permitió identificar, analizar y clasificar diversas metodologías de priorización de la DT. Se detectó una evolución en las propuestas metodológicas, pasando de enfoques puramente técnicos hacia metodologías más integradas que consideran tanto aspectos de negocio como necesidades técnicas. En este sentido, se revisaron enfoques basados en modelos de decisión multicriterio, algoritmos evolutivos y técnicas orientadas a la gestión del negocio.

La literatura revisada destacó la importancia de que las empresas tecnológicas no solo consideren la criticidad técnica de su DT, sino que también integren en sus decisiones factores como el impacto financiero, la capacidad de innovación y la estabilidad a largo plazo. Este análisis permitió sentar las bases teóricas para la selección de una metodología adecuada para Anfix, priorizando aquellas que ofrecieran un balance adecuado entre la resolución de problemas técnicos y el aporte de valor comercial.

6.3. Análisis y comparación de metodologías

A lo largo del trabajo, se realizó un análisis detallado de diversas metodologías de priorización de la DT, con el objetivo de identificar la que mejor se ajustara a las necesidades específicas de Anfix. Se evaluaron enfoques desde los más técnicos, enfocados en la calidad del código y la refactorización, hasta metodologías más holísticas que integraban criterios comerciales, como el Framework Tracy.

La selección final de Framework Tracy se basó en su capacidad para integrar las perspectivas técnica y comercial, logrando un enfoque equilibrado que permite priorizar la DT no solo en función de su impacto técnico, sino también considerando el valor que aporta al negocio. Esta metodología permitió identificar las áreas críticas de la empresa, priorizando aquellas que tuvieran el mayor impacto en los activos de TI y en los procesos esenciales del negocio. Esto fue clave para asegurar que los recursos destinados a la reducción de la DT fueran empleados de manera eficiente y alineados con los objetivos estratégicos de la empresa.

Además, se evidenció que la flexibilidad de adaptación del framework fue fundamental para su éxito, ya que permitió a Anfix ajustarse a las demandas del mercado sin comprometer la estabilidad de su software. El enfoque colaborativo que promueve este framework, integrando a los distintos actores del negocio en la toma de decisiones, facilitó un proceso de priorización más consensuado y orientado hacia el crecimiento a largo plazo.

6.4. Resultados obtenidos en Anfix

La implementación del Framework Tracy en Anfix se centró en un enfoque estructurado para la priorización de la DT, alineando las decisiones técnicas con los objetivos estratégicos de la empresa. El proceso de priorización se llevó a cabo mediante un análisis detallado de las partidas de DT, clasificándolas en función de su impacto tanto en los activos de TI como en los procesos de negocio. Para realizar la priorización de la DT, se establecieron métricas clave que incluían factores como el impacto en la experiencia del cliente, costes operativos, valor comercial, y la escalabilidad del sistema. Estas métricas fueron definidas en colaboración con los diferentes stakeholders, incluidos los equipos técnicos, los gerentes de producto, y los directivos de Anfix. La participación activa de los stakeholders en la definición de estas métricas fue crucial para asegurar que la priorización reflejara tanto las necesidades técnicas como los objetivos comerciales de la empresa. Uno de los aspectos más importantes de este proceso fue la cohesión entre los stakeholders, quienes colaboraron activamente en la evaluación y priorización de la DT. Este enfoque colaborativo permitió a los equipos técnicos y de negocio trabajar juntos en la alineación de las decisiones técnicas con los objetivos estratégicos de la empresa. La creación de esta visión común aseguró que la priorización no solo respondiera a problemas técnicos urgentes, sino también a la necesidad de mantener la competitividad de Anfix en el mercado. Otro de los resultados clave fue la mejora en la eficiencia operativa. Al gestionar de manera proactiva la DT, Anfix logró optimizar la asignación de recursos, lo que no solo redujo los costos asociados

a la acumulación de DT, sino que también liberó tiempo y esfuerzo para invertir en nuevas oportunidades de crecimiento.

6.5. Reflexiones finales

Con este Trabajo de Fin de Máster se muestra que una adecuada priorización de la DT no es simplemente una cuestión técnica, sino una estrategia fundamental para asegurar la competitividad y la sostenibilidad de una empresa en un entorno tecnológico. El caso de Anfix ilustra cómo la correcta gestión de la DT, mediante la aplicación de una metodología adecuada, permite alinear las decisiones técnicas con los objetivos estratégicos del negocio. La colaboración entre los distintos stakeholders fue esencial para el éxito de este proceso, logrando una cohesión que permitió definir y priorizar las tareas de acuerdo con el impacto técnico y comercial. Este enfoque colaborativo no solo optimizó la eficiencia operativa de la empresa, sino que también facilitó la planificación a largo plazo, asegurando que las decisiones de negocio y las prioridades técnicas estuvieran en sincronía. Además, la implementación del Framework Tracy permitió a Anfix establecer una base sólida para gestionar la DT de forma proactiva, evitando su acumulación futura. Este proceso de priorización ha sido clave para mejorar la capacidad de la empresa para innovar sin comprometer la estabilidad de su sistema, mostrando que la gestión de la DT debe ser una prioridad constante para cualquier empresa que desee mantenerse competitiva en un mercado tan dinámico.

6.6. Trabajo futuro

A partir de los hallazgos y resultados de este estudio, se identifican varias oportunidades para trabajos futuros que podrían ampliar el enfoque y el impacto de la priorización de la DT en diferentes contextos. Algunas de las líneas de investigación y desarrollo futuro incluyen:

1. **Automatización del proceso de priorización:** Aunque el proceso de priorización de la DT en este trabajo se realizó con un alto grado de colaboración entre los stakeholders, una futura línea de investigación podría enfocarse en la automatización parcial o total de este proceso. Desarrollar herramientas o algoritmos que integren las métricas clave y generen automáticamente recomendaciones de priorización podría optimizar aún más la asignación de recursos.
2. **Ampliación del Framework Tracy a otros contextos empresariales:** Si bien la metodología utilizada mostró ser efectiva para Anfix, sería interesante probar su aplicabilidad en otras empresas de distintos sectores, especialmente aquellas con estructuras tecnológicas más complejas o con requerimientos normativos más estrictos. Esto permitiría evaluar la adaptabilidad y eficacia del Framework Tracy en otros entornos de negocio.

3. **Estudio del impacto a largo plazo de la priorización de la DT:** Un análisis más profundo y a largo plazo sobre cómo la priorización de la DT afecta la capacidad de innovación, la sostenibilidad técnica y la competitividad de las empresas podría proporcionar valiosa información para refinar las metodologías actuales. Este tipo de estudio podría incluir métricas más detalladas de los beneficios económicos y operativos resultantes de una correcta priorización de la DT.
4. **Incorporación de inteligencia artificial (IA):** El uso de IA en la priorización de la DT es otra área prometedora. A través del aprendizaje automático, se podría identificar patrones recurrentes en la DT y predecir el impacto futuro de la acumulación de DT, lo que podría mejorar aún más la precisión en la priorización y la toma de decisiones.

Bibliografía

- [1] ABUHASSAN, A., ALSHAYEB, M., AND GHOUTI, L. Detection of design smells using adaptive neuro-fuzzy approaches. *International Journal of Fuzzy Systems 24* (August 2022), 1361–1376.
- [2] AGHAYI, E., LATOZA, T. D., SURENDRA, P., AND ABOLGHASEMI, S. Crowdsourced behavior-driven development. *Social Science Research Network* (2020).
- [3] ALFAYEZ, R., ALWEHAIBI, W., WINN, R., VENSON, E., AND BOEHM, B. A systematic literature review of technical debt prioritization. In *A systematic literature review of technical debt prioritization* (6 2020), Association for Computing Machinery, Inc, pp. 1–10.
- [4] ALFAYEZ, R., AND BOEHM, B. Technical debt prioritization: A search-based approach. *Information and Software Technology* (10 2019).
- [5] ALFAYEZ, R., WINN, R., ALWEHAIBI, W., VENSON, E., AND BOEHM, B. How sonarqube-identified technical debt is prioritized: An exploratory case study. *Information and Software Technology 156* (4 2023).
- [6] ALMEIDA, R. R. D., KULESZA, U., TREUDE, C., FEITOSA, D. C., AND LIMA, A. H. G. Aligning technical debt prioritization with business objectives: A multiple-case study. In *Aligning technical debt prioritization with business objectives: A multiple-case study* (11 2018), Institute of Electrical and Electronics Engineers Inc., pp. 655–664.
- [7] ALMEIDA, R. R. D., KULESZA, U., TREUDE, C., FEITOSA, D. C., AND LIMA, A. H. G. Business-driven technical debt prioritization. In *Business-Driven Technical Debt Prioritization* (12 2019), Institute of Electrical and Electronics Engineers Inc., pp. 605–609.
- [8] ALMEIDA, R. R. D., KULESZA, U., TREUDE, C., FEITOSA, D. C., AND LIMA, A. H. G. Tracy: A business-driven technical debt prioritization framework. In *Tracy: A Business-Driven Technical Debt Prioritization Framework* (12 2019), Institute of Electrical and Electronics Engineers Inc., pp. 181–185.

- [9] ALMEIDA, R. R. D., KULESZA, U., TREUDE, C., FEITOSA, D. C., AND LIMA, A. H. G. What's behind tight deadlines? business causes of technical debt. In *What's behind tight deadlines? Business causes of technical debt* (7 2023), 2023 IEEE/ACM 16th International Conference on Cooperative and Human Aspects of Software Engineering (CHASE).
- [10] ALMEIDA, R. R. D., RIBEIRO, R. D. N., TREUDE, C., AND KULESZA, U. Business-driven technical debt prioritization: An industrial case study. In *Business-Driven Technical Debt Prioritization: An Industrial Case Study* (5 2021), Institute of Electrical and Electronics Engineers Inc., pp. 74–83.
- [11] Anfix. <https://www.anfix.com>, 2024. Acceso: 12 de julio de 2024.
- [12] BALALAIIE, A., HEYDARNOORI, A., AND JAMSHIDI, P. Microservices architecture enables devops: Migration to a cloud-native architecture. *IEEE Software* (September 2016).
- [13] BIRD, C., RAHMAN, F., AND DEVANBU, P. The promises and perils of mining git. In *The promises and perils of mining git* (September 2010).
- [14] BROWN, N., CAI, Y., GUO, Y., KAZMAN, R., KIM, M., AND KRUCHTEN, P. Managing technical debt in software-reliant systems. In *Managing technical debt in software-reliant systems* (November 2010).
- [15] Cast highlight software intelligence. <https://www.castsoftware.com/products/highlight>, 2024. Acceso: 10 de Mayo de 2024.
- [16] CODABUX, Z., AND WILLIAMS, B. J. Technical debt prioritization using predictive analytics. In *Technical Debt Prioritization Using Predictive Analytics* (3 2017), Institute of Electrical and Electronics Engineers Inc., pp. 704–706.
- [17] Code climate: Code review and quality analysis. <https://codeclimate.com/>, 2023. Acceso: 15 de Abril de 2024.
- [18] Code climate velocity. <https://codeclimate.com/velocity/>, 2023. Acceso: 1 de Junio de 2024.
- [19] Codescene: Visualize, prioritize and improve your codebase. <https://codescene.com>, 2024. Acceso: 28 de Julio de 2024.
- [20] CUNNINGHAM, W. The wycash portfolio management system. In *The WyCash portfolio management system* (March 1992).
- [21] DETOFENO, T., MALUCELLI, A., AND REINEHR, S. Priortd: A method for prioritization technical debt. In *PriorTD: A Method for Prioritization Technical Debt* (10 2022), Association for Computing Machinery, pp. 230–240.
- [22] FOWLER, M. Technical debt quadrant. <https://martinfowler.com/bliki/TechnicalDebtQuadrant.html>, 7 2009. Accessed: 2024-05-29.

- [23] GUO, Y., AND SEAMAN, C. A portfolio approach to technical debt management. In *A portfolio approach to technical debt management* (May 2011).
- [24] ISO. Sistemas e ingeniería de software-requerimientos y evaluación de sistemas y calidad de software (square)-modelos de calidad del sistema y software (iso/iec 25010:2011, idt). <https://www.iso.org/standard/35733.html>, 2014.
- [25] K, Y. R. The case study as a serious research strategy. <https://us.sagepub.com/en-us/nam/the-case-study-anthology/book233403>, 2015.
- [26] KAZMAN, R., AND CHEN, H.-M. The role of architecture in managing technical debt. In *The role of architecture in managing technical debt* (May 2015).
- [27] Kiuwan: Code analysis and software analytics for security and quality. <https://www.kiuwan.com>, 2024. Acceso: 28 de Julio de 2024.
- [28] KRUCHTEN, P., NORD, R. L., AND OZKAYA, I. Technical debt: From metaphor to theory and practice. In *Technical debt: From metaphor to theory and practice* (November 2012).
- [29] LENARDUZZI, V., BESKER, T., TAIBI, D., MARTINI, A., AND FONTANA, F. A. A systematic literature review on technical debt prioritization: Strategies, processes, factors, and tools. *Journal of Systems and Software* 171 (7 2021).
- [30] LI, Z., AVGERIOU, P., AND LIANG, P. A systematic mapping study on technical debt and its management. *Journal of Systems and Software* 101 (3 2015), 193–220.
- [31] MCCONNELL, S. *Code Complete*. Microsoft Press, June 2007.
- [32] MESZAROS, G. *xUnit Test Patterns: Refactoring Test Code*. Addison-Wesley Professional, May 2007.
- [33] Pmd source code analyzer. <https://pmd.github.io/>, 2023. Acceso: 25 de Mayo de 2024.
- [34] RIBEIRO, L. F., ALVES, N. S. R., NETO, M. G. D. M., AND SPINOLA, R. O. A strategy based on multiple decision criteria to support technical debt management. In *A strategy based on multiple decision criteria to support technical debt management* (9 2017), Institute of Electrical and Electronics Engineers Inc., pp. 334–341.
- [35] SEAMAN, C. The role of communication in managing technical debt. In *The role of communication in managing technical debt* (May 2013).
- [36] SHULL, F., SEAMAN, C., AND GODFREY, M. Evaluating software engineering technologies. In *Evaluating software engineering technologies* (May 2002).
- [37] SILVA, F. T. D., SOUZA, E. O. S. D., ALMEIDA, R. R. D., AND SANTOS, W. B. Business-driven technical debt prioritization: A replication study. In *2022 17th Iberian Conference on Information Systems and Technologies (CISTI)* (2022), Institute of Electrical and Electronics Engineers Inc.

- [38] SONARSOURCE. Sonarqube. <https://www.sonarqube.org/>, 2023. Accessed: 2024-06-04.
- [39] Stack overflow developer survey 2022: Methodology and results. <https://survey.stackoverflow.co/2022/#methodology-general>, 2022. Accessed: 28 de Julio de 2024.
- [40] STOCHEL, M. G., CHOLDA, P., AND WAWROWSKI, M. R. Continuous debt valuation approach (codva) for technical debt prioritization. In *Continuous Debt Valuation Approach (CoDVA) for Technical Debt Prioritization* (10 2020), Institute of Electrical and Electronics Engineers Inc.
- [41] STOCHEL, M. G., CHOLDA, P., AND WAWROWSKI, M. R. Adopting devops paradigm in technical debt prioritization and mitigation. In *Adopting DevOps Paradigm in Technical Debt Prioritization and Mitigation* (2022), Institute of Electrical and Electronics Engineers Inc., pp. 306–313.
- [42] STOCHEL, M. G., CHOLDA, P., AND WAWROWSKI, M. R. Business-driven technical debt management using continuous debt valuation approach (codva). In *Business-driven technical debt management using Continuous Debt Valuation Approach (CoDVA)*. (2023), Institute of Electrical and Electronics Engineers Inc.
- [43] STOCHEL, M. G., WAWROWSKI, M. R., AND CHOLDA, P. Technical debt prioritization in telecommunication applications: Why the actual refactoring deviates from the plan and how to remediate it? case study in the covid era. *Applied Sciences* (2022).
- [44] TOLEDO, S. S. D., MARTINI, A., PRZYBYSZEWSKA, A., AND SJOBERG, D. I. Architectural technical debt in microservices: A case study in a large company. In *Architectural technical debt in microservices: A case study in a large company* (5 2019), Institute of Electrical and Electronics Engineers Inc., pp. 78–87.
- [45] TSOUKALAS, D., MITTAS, N., CHATZIGEORGIOU, A., KEHAGIAS, D., AMPATZOGLOU, A., AMANATIDIS, T., AND ANGELIS, L. Machine learning for technical debt identification. *IEEE Transactions on Software Engineering* 47, 12 (2021), 2690–2706.
- [46] XIAO, L., CAI, Y., KAZMAN, R., MO, R., AND FENG, Q. Detecting the locations and predicting the maintenance costs of compound architectural debts. *IEEE Transactions on Software Engineering* 48 (9 2022), 3686–3715.
- [47] ZAZWORKA, N., SHAW, M. A., SHULL, F., AND SEAMAN, C. Investigating the impact of design debt on software quality. In *Investigating the impact of design debt on software quality* (May 2011), pp. 17–23.

Apéndices

Apéndice A

Documentación de la deuda técnica identificada

Este apéndice presentará la documentación de la deuda técnica identificada, eliminando los datos que son confidenciales de la empresa. Por tanto, se incluye en esta documentación un código único asignado a cada partida de DT identificada y su caracterización, sin entrar en el detalle preciso de su descripción que se considera información interna y confidencial de la empresa.

Número DT	Código	Características
DT-1	AF-3481	<ul style="list-style-type: none"> ■ Causa: Deuda no intencional. ■ Impacto: Alto. ■ Urgencia: Media. ■ Esfuerzo: Alto. ■ Clasificación: Prudente Involuntaria ■ Tipos de DT afectadas: Arquitectura, Mantenimiento, Operaciones.
DT-2	AF-1566	<ul style="list-style-type: none"> ■ Causa: Deuda intencional. ■ Impacto: Alto. ■ Urgencia: Alto. ■ Esfuerzo: Alto. ■ Clasificación: Prudente Involuntaria ■ Tipos de DT afectadas: Arquitectura, Mantenimiento, Rendimiento.
DT-3	AF-7518	<ul style="list-style-type: none"> ■ Causa: Deuda no intencional. ■ Impacto: Bajo. ■ Urgencia: Media. ■ Esfuerzo: Bajo (limpieza de código). ■ Clasificación: Prudente Involuntaria ■ Tipos de DT afectadas: Mantenimiento, Código, Rendimiento.

Número DT	Código	Características
DT-4	AF-10283	<ul style="list-style-type: none"> ■ Causa: Deuda no intencional. ■ Impacto: Bajo. ■ Urgencia: Media. ■ Esfuerzo: Bajo. ■ Clasificación: Prudente Involuntaria. ■ Tipos de DT afectadas: Mantenimiento, Código, Rendimiento.
DT-5	AF-10575	<ul style="list-style-type: none"> ■ Causa: Deuda no intencional. ■ Impacto: Alto. ■ Urgencia: Alta. ■ Esfuerzo: Alto. ■ Clasificación: Prudente Involuntaria ■ Tipos de DT afectadas: Dependencia, Mantenimiento, Funcionalidad, Usabilidad.
DT-6	AF-7587	<ul style="list-style-type: none"> ■ Causa: Deuda no intencional. ■ Impacto: Alto. ■ Urgencia: Alta. ■ Esfuerzo: Alto. ■ Clasificación: Prudente Involuntaria ■ Tipos de DT afectadas: Rendimiento, Mantenimiento, Arquitectura.

Número DT	Código	Características
DT-7	AF-1567	<ul style="list-style-type: none"> ■ Causa: Deuda intencional. ■ Impacto: Alto. ■ Urgencia: Media. ■ Esfuerzo: Alto. ■ Clasificación: Prudente Involuntaria ■ Tipos de DT afectadas: Arquitectura, Mantenimiento, Usabilidad.
DT-8	AF-1569	<ul style="list-style-type: none"> ■ Causa: Deuda no intencional. ■ Impacto: Alto. ■ Urgencia: Alta. ■ Esfuerzo: Alto. ■ Clasificación: Prudente Involuntaria ■ Tipos de DT afectadas: Arquitectura, Rendimiento, Operaciones.
DT-9	AF-1236	<ul style="list-style-type: none"> ■ Causa: Deuda no intencional. ■ Impacto: Medio. ■ Urgencia: Media. ■ Esfuerzo: Medio. ■ Clasificación: Prudente Involuntaria ■ Tipos de DT afectadas: Mantenimiento, Funcionalidad, Arquitectura.

Número DT	Código	Características
DT-10	AF-370	<ul style="list-style-type: none"> ■ Causa: Deuda no intencional. ■ Impacto: Medio. ■ Urgencia: Media. ■ Esfuerzo: Alto. ■ Clasificación: Imprudente Involuntaria ■ Tipos de DT afectadas: Mantenimiento, Usabilidad, Rendimiento, Arquitectura.
DT-11	AF-371	<ul style="list-style-type: none"> ■ Causa: Deuda no intencional. ■ Impacto: Medio. ■ Urgencia: Media. ■ Esfuerzo: Medio. ■ Clasificación: Prudente Involuntaria ■ Tipos de DT afectadas: Arquitectura, Usabilidad, Rendimiento, Mantenimiento.
DT-12	AF-372	<ul style="list-style-type: none"> ■ Causa: Deuda no intencional. ■ Impacto: Bajo. ■ Urgencia: Media. ■ Esfuerzo: Bajo. ■ Clasificación: Prudente Involuntaria ■ Tipos de DT afectadas: Mantenimiento, Rendimiento, Arquitectura.

Número DT	Código	Características
DT-13	AF-385	<ul style="list-style-type: none"> ■ Causa: Deuda no intencional. ■ Impacto: Medio. ■ Urgencia: Media. ■ Esfuerzo: Medio. ■ Clasificación: Prudente Deliberada ■ Tipos de DT afectadas: Documentación, Mantenimiento, Rendimiento.
DT-14	AF-386	<ul style="list-style-type: none"> ■ Causa: Deuda no intencional. ■ Impacto: Medio. ■ Urgencia: Media. ■ Esfuerzo: Medio. ■ Clasificación: Prudente Deliberada ■ Tipos de DT afectadas: Funcionalidad, Mantenimiento.
DT-15	AF-387	<ul style="list-style-type: none"> ■ Causa: Deuda no intencional. ■ Impacto: Alto. ■ Urgencia: Alta. ■ Esfuerzo: Medio. ■ Clasificación: Prudente Involuntaria ■ Tipos de DT afectadas: Arquitectura, Mantenimiento, Rendimiento, Funcionalidad.

Número DT	Código	Características
DT-16	AF-389	<ul style="list-style-type: none"> ■ Causa: Deuda no intencional. ■ Impacto: Alto. ■ Urgencia: Alta. ■ Esfuerzo: Alto . ■ Clasificación: Imprudente Involuntaria ■ Tipos de DT afectadas: Arquitectura, Mantenimiento.
DT-17	AF-392	<ul style="list-style-type: none"> ■ Causa: Deuda no intencional. ■ Impacto: Medio. ■ Urgencia: Media. ■ Esfuerzo: Medio. ■ Clasificación: Prudente Involuntaria ■ Tipos de DT afectadas: Arquitectura, Mantenimiento, Código.
DT-18	AF-393	<ul style="list-style-type: none"> ■ Causa: Deuda intencional. ■ Impacto: Medio. ■ Urgencia: Media. ■ Esfuerzo: Medio. ■ Clasificación: Prudente Deliberada ■ Tipos de DT afectadas: Arquitectura, Mantenimiento, Integración.

Número DT	Código	Características
DT-19	AF-373	<ul style="list-style-type: none"> ■ Causa: Deuda no intencional. ■ Impacto: Medio. ■ Urgencia: Media. ■ Esfuerzo: Bajo. ■ Clasificación: Prudente Involuntaria ■ Tipos de DT afectadas: Test, Mantenimiento, Rendimiento.
DT-20	AF-379	<ul style="list-style-type: none"> ■ Causa: Deuda no intencional. ■ Impacto: Bajo. ■ Urgencia: Media. ■ Esfuerzo: Bajo. ■ Clasificación: Prudente Involuntaria ■ Tipos de DT afectadas: Mantenimiento, Funcionalidad, Rendimiento.
DT-21	AF-380	<ul style="list-style-type: none"> ■ Causa: Deuda no intencional. ■ Impacto: Bajo. ■ Urgencia: Media. ■ Esfuerzo: Bajo. ■ Clasificación: Prudente Involuntaria ■ Tipos de DT afectadas: Mantenimiento, Arquitectura, Funcionalidad.

Número DT	Código	Características
DT-22	AF-381	<ul style="list-style-type: none"> ■ Causa: Deuda no intencional. ■ Impacto: Medio. ■ Urgencia: Media. ■ Esfuerzo: Bajo. ■ Clasificación: Imprudente Involuntaria ■ Tipos de DT afectadas: Mantenimiento, Funcionalidad.
DT-23	AF-382	<ul style="list-style-type: none"> ■ Causa: Deuda intencional. ■ Impacto: Medio. ■ Urgencia: Media. ■ Esfuerzo: Medio. ■ Clasificación: Prudente Deliberada ■ Tipos de DT afectadas: Mantenimiento, Funcionalidad, Arquitectura, Rendimiento.
DT-24	AF-374	<ul style="list-style-type: none"> ■ Causa: Deuda no intencional. ■ Impacto: Alto. ■ Urgencia: Alta. ■ Esfuerzo: Medio. ■ Clasificación: Imprudente Involuntaria ■ Tipos de DT afectadas: Funcionalidad, Mantenimiento.

Número DT	Código	Características
DT-25	AF-375	<ul style="list-style-type: none"> ■ Causa: Deuda no intencional. ■ Impacto: Medio. ■ Urgencia: Media. ■ Esfuerzo: Medio. ■ Clasificación: Prudente Involuntaria. ■ Tipos de DT afectadas: Rendimiento, Mantenimiento, Dependencia.
DT-26	AF-376	<ul style="list-style-type: none"> ■ Causa: Deuda no intencional. ■ Impacto: Alto. ■ Urgencia: Alta. ■ Esfuerzo: Alto (requiere desacoplar, refactorizar y migrar a AWS Lambda). ■ Clasificación: Prudente Deliberada ■ Tipos de DT afectadas: Rendimiento, Mantenimiento, Funcionalidad.
DT-27	AF-2423	<ul style="list-style-type: none"> ■ Causa: Deuda no intencional. ■ Impacto: Medio. ■ Urgencia: Media. ■ Esfuerzo: Medio. ■ Clasificación: Prudente Involuntaria ■ Tipos de DT afectadas: Mantenimiento, Arquitectura, Rendimiento.

Número DT	Código	Características
DT-28	AF-367	<ul style="list-style-type: none"> ■ Causa: Deuda no intencional. ■ Impacto: Medio. ■ Urgencia: Media. ■ Esfuerzo: Medio. ■ Clasificación: Prudente Involuntaria. ■ Tipos de DT afectadas: Mantenimiento, Operaciones.
DT-29	AF-368	<ul style="list-style-type: none"> ■ Causa: Deuda no intencional. ■ Impacto: Alto. ■ Urgencia: Alta. ■ Esfuerzo: Alto. ■ Clasificación: Prudente Deliberada ■ Tipos de DT afectadas: Mantenimiento, Arquitectura, Rendimiento.
DT-30	AF-901	<ul style="list-style-type: none"> ■ Causa: Deuda no intencional. ■ Impacto: medio. ■ Urgencia: Media. ■ Esfuerzo: medio. ■ Clasificación: Prudente Involuntaria ■ Tipos de DT afectadas: Rendimiento y Arquitectura.

Número DT	Código	Características
DT-31	AF-911	<ul style="list-style-type: none"> ■ Causa: Deuda intencional. ■ Impacto: Bajo. ■ Urgencia: Media. ■ Esfuerzo: Bajo. ■ Clasificación: Prudente Involuntaria ■ Tipos de DT afectadas: Mantenimiento, Seguridad, Código.
DT-32	AF-1565	<ul style="list-style-type: none"> ■ Causa: Deuda intencional. ■ Impacto: Medio. ■ Urgencia: Media. ■ Esfuerzo: Alto. ■ Clasificación: Prudente Involuntaria ■ Tipos de DT afectadas: Mantenimiento, Arquitectura, Rendimiento.
DT-33	AF-883	<ul style="list-style-type: none"> ■ Causa: Deuda no intencional. ■ Impacto: Medio. ■ Urgencia: Media. ■ Esfuerzo: Alto. ■ Clasificación: Prudente Involuntaria. ■ Tipos de DT afectadas: Mantenimiento, Rendimiento, Arquitectura.

Número DT	Código	Características
DT-34	AF-1564	<ul style="list-style-type: none"> ■ Causa: Deuda no intencional. ■ Impacto: Alto. ■ Urgencia: Alta. ■ Esfuerzo: Alto. ■ Clasificación: Prudente Involuntaria ■ Tipos de DT afectadas: Arquitectura, Mantenimiento, Rendimiento.
DT-35	AF-1678	<ul style="list-style-type: none"> ■ Causa: Deuda no intencional. ■ Impacto: Medio. ■ Urgencia: Media. ■ Esfuerzo: Bajo. ■ Clasificación: Prudente Involuntaria. ■ Tipos de DT afectadas: Mantenimiento, Rendimiento, Funcionalidad, Operaciones.
DT-36	AF-2512	<ul style="list-style-type: none"> ■ Causa: Deuda no intencional. ■ Impacto: Medio. ■ Urgencia: Media. ■ Esfuerzo: Medio. ■ Clasificación: Prudente Involuntaria. ■ Tipos de DT afectadas: Rendimiento, Funcionalidad, Usabilidad.

Número DT	Código	Características
DT-37	AF-2962	<ul style="list-style-type: none"> ■ Causa: Deuda no intencional. ■ Impacto: Alto. ■ Urgencia: Alta. ■ Esfuerzo: Alto. ■ Clasificación: Prudente Involuntaria ■ Tipos de DT afectadas: Rendimiento, Arquitectura.
DT-38	AF-3660	<ul style="list-style-type: none"> ■ Causa: Deuda no intencional. ■ Impacto: Medio. ■ Urgencia: Media. ■ Esfuerzo: Alto. ■ Clasificación: Prudente Involuntaria. ■ Tipos de DT afectadas: Arquitectura, Operaciones.
DT-39	AF-3706	<ul style="list-style-type: none"> ■ Causa: Deuda no intencional. ■ Impacto: Bajo. ■ Urgencia: Media. ■ Esfuerzo: Bajo. ■ Clasificación: Prudente Involuntaria. ■ Tipos de DT afectadas: Mantenimiento, Rendimiento, Usabilidad.

Número DT	Código	Características
DT-40	AF-4914	<ul style="list-style-type: none"> ■ Causa: Deuda no intencional. ■ Impacto: Medio. ■ Urgencia: Media. ■ Esfuerzo: Bajo. ■ Clasificación: Prudente Involuntaria. ■ Tipos de DT afectadas: Mantenimiento, Funcionalidad.
DT-41	AF-5020	<ul style="list-style-type: none"> ■ Causa: Deuda no intencional. ■ Impacto: Medio. ■ Urgencia: Media. ■ Esfuerzo: Medio. ■ Clasificación: Prudente Involuntaria. ■ Tipos de DT afectadas: Mantenimiento, Rendimiento, Dependencia, Funcionalidad.
DT-42	AF-5627	<ul style="list-style-type: none"> ■ Causa: Deuda no intencional. ■ Impacto: Bajo. ■ Urgencia: Media. ■ Esfuerzo: Bajo. ■ Clasificación: Prudente Deliberada ■ Tipos de DT afectadas: Mantenimiento, Código.

Número DT	Código	Características
DT-43	AF-5836	<ul style="list-style-type: none"> ■ Causa: Deuda no intencional. ■ Impacto: Alto. ■ Urgencia: Alta. ■ Esfuerzo: Medio. ■ Clasificación: Imprudente Involuntaria ■ Tipos de DT afectadas: Mantenimiento, Seguridad.
DT-44	AF-6904	<ul style="list-style-type: none"> ■ Causa: Deuda intencional. ■ Impacto: Alto. ■ Urgencia: Alta. ■ Esfuerzo: Alto. ■ Clasificación: Prudente Deliberada ■ Tipos de DT afectadas: Mantenimiento, Arquitectura, Rendimiento.
DT-45	AF-7733	<ul style="list-style-type: none"> ■ Causa: Deuda no intencional. ■ Impacto: Alto. ■ Urgencia: Alta. ■ Esfuerzo: Alto. ■ Clasificación: Prudente Involuntaria ■ Tipos de DT afectadas: Arquitectura, Rendimiento y Dependencia.

Tabla A.1: Listado de incidencias registradas