

UNIVERSIDAD TÉCNICA DE AMBATO

FACULTAD DE INGENIERÍA EN SISTEMAS, ELECTRÓNICA E INDUSTRIAL

CARRERA DE TECNOLOGÍAS DE LA INFORMACIÓN

Tema:

"METODOLOGÍA ÁGIL A MEDIDA QUE PERMITA OPTIMIZAR EL DESARROLLO DE SOFTWARE EN LA EMPRESA BLACKLABELTECH BLACK LABEL TECHNOLOGY CIA. LTDA."

Trabajo de titulación modalidad Proyecto de Investigación, presentado previo a la obtención del título de Ingeniera en Tecnologías de la Información.

ÁREA: Gestión de Tecnologías de la Información

LÍNEA DE INVESTIGACIÓN: Software, Tecnologías de la Información y Ciencias de Datos

AUTOR: Madeleine Adriana Erazo Sánchez

TUTOR: Ing. Clay Fernando Aldás Flores, Mg.

Ambato - Ecuador

julio – 2026

APROBACIÓN DEL TUTOR

En calidad de tutor del trabajo de titulación con el tema: METODOLOGÍA ÁGIL A MEDIDA QUE PERMITA OPTIMIZAR EL DESARROLLO DE SOFTWARE EN LA EMPRESA BLACKLABELTECH BLACK LABEL TECHNOLOGY CIA. LTDA., desarrollado bajo la modalidad Proyecto de Investigación por la señorita Madeleine Adriana Erazo Sánchez, estudiante de la Carrera de Tecnologías de la Información, de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, de la Universidad Técnica de Ambato, me permito indicar que la estudiante ha sido tutorada durante todo el desarrollo del trabajo hasta su conclusión, de acuerdo a lo dispuesto en el Artículo 17 del Reglamento para la Titulación de Grado en la Universidad Técnica de Ambato y el numeral 6.3 del instructivo del reglamento referido.

Ambato, julio 2026

Validar únicamente en FirmaEC.
Firmado electrónicamente por:
CLAY FERNANDO ALDAS FLORES

Ing. Clay Fernando Aldás Flores, Mg.
TUTOR

AUTORÍA

El presente trabajo de titulación con el tema: METODOLOGÍA ÁGIL A MEDIDA QUE PERMITA OPTIMIZAR EL DESARROLLO DE SOFTWARE EN LA EMPRESA BLACKLABELTECH BLACK LABEL TECHNOLOGY CIA. LTDA. es absolutamente original, auténtico y personal y ha observado los preceptos establecidos en la Disposición General Quinta del Reglamento para la Titulación de Grado en la Universidad Técnica de Ambato. En tal virtud, el contenido, efectos legales y académicos que se desprenden del mismo son de exclusiva responsabilidad del autor.

Ambato, julio 2026



Madeleine Adriana Erazo Sánchez

C.C 1804397915

AUTOR

DERECHOS DE AUTOR

Autorizo a la Universidad Técnica de Ambato para que reproduzca total o parcialmente este trabajo de titulación dentro de las regulaciones legales e institucionales correspondientes. Además, cedo todos mis derechos de autor a favor de la institución con el propósito de su difusión pública, por lo tanto, autorizo su publicación en el repositorio virtual institucional como un documento disponible para la lectura y uso con fines académicos e investigativos de acuerdo con la Disposición General Cuarta del Reglamento para la Titulación de Grado en la Universidad Técnica de Ambato.

Ambato, julio 2026



Madeleine Adriana Erazo Sánchez

C.C 1804397915

AUTOR

APROBACIÓN DEL TRIBUNAL DE GRADO

En calidad de par calificador del informe final del trabajo de titulación presentado por la señorita Madeleine Adriana Erazo Sánchez, estudiante de la Carrera de Tecnologías de la Información, de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, bajo la Modalidad Proyecto de Investigación, titulado METODOLOGÍA ÁGIL A MEDIDA QUE PERMITA OPTIMIZAR EL DESARROLLO DE SOFTWARE EN LA EMPRESA BLACKLABELTECH BLACK LABEL TECHNOLOGY CIA. LTDA., nos permitimos informar que el trabajo ha sido revisado y calificado de acuerdo con el Artículo 19 del Reglamento para la Titulación de Grado en la Universidad Técnica de Ambato y el numeral 6.4 del instructivo del reglamento referido. Para cuya constancia suscribimos, conjuntamente con el señor Presidente del Tribunal.

Ambato, julio 2026

Validar únicamente en FirmaEC.
Firmado electrónicamente por:
FRANKLIN OSWALDO MAYORGA MAYORGA

Ing. Franklin Oswaldo Mayorga Mayorga, Mg.
PRESIDENTE DEL TRIBUNAL

Validar únicamente en FirmaEC.
Firmado electrónicamente por:
JULIO ENRIQUE BALAREZO LOPEZ

Ing. Julio Enrique Balarezo López PhD.
PROFESOR CALIFICADOR

Validar únicamente en FirmaEC.
Firmado electrónicamente por:
PAULO CESAR TORRES ABRIL

Ing. Paulo Cesar Torres Abril Mg.
PROFESOR CALIFICADOR

DEDICATORIA

Dedico este trabajo, en primer lugar, a Dios, por darme la fortaleza, la sabiduría y la perseverancia necesarias para superar cada obstáculo y permitirme alcanzar una de las metas más importantes de mi vida.

A mi madre, Gloria Elisa Sánchez, quien hoy me acompaña desde el cielo. Gracias por creer siempre en mí y por enseñarme el valor del esfuerzo. Fuiste mi mayor inspiración y la principal motivación para culminar esta carrera.

Este logro representa una de las metas que alguna vez soñamos juntas, y aunque hoy ya no estés físicamente a mi lado para compartir este momento, cada paso que di para llegar hasta aquí estuvo impulsado por tu amor, tus enseñanzas y el deseo de hacerte sentir orgullosa. Tu recuerdo me sigue permitiendo mantenerme de pie y encontrar fuerzas para seguir adelante.

Por eso, este trabajo lleva grabado tu nombre en cada una de sus páginas, porque siempre fue por ti y para ti. Sé que, donde quiera que estés, compartes conmigo la alegría de haber alcanzado este sueño. Este logro es tan tuyo como mío, y estará siempre dedicado a tu memoria, con todo mi amor, gratitud y admiración.

A mi hermano, Henry David Erazo, quien más que un hermano ha sido mi pilar, mi apoyo incondicional y mi compañero en los momentos más difíciles. Gracias por estar presente, por impulsarme a seguir adelante y por creer en mí incluso cuando yo dudaba de mis propias capacidades.

Y, finalmente, a mis perritos, quienes con su amor incondicional y compañía hicieron más llevadero este proceso. Gracias por ser mi refugio y mi alegría en los días más difíciles.

Con profundo cariño, dedico este logro a quienes fueron y son parte fundamental de mi vida.

AGRADECIMIENTO

Agradezco, en primer lugar, a Dios, por brindarme la fortaleza, la sabiduría y la perseverancia necesarias para alcanzar esta meta.

A mi madre, Gloria Elisa Sánchez, y a mi hermano, Henry David Erazo, por su amor, apoyo incondicional y por ser una fuente constante de inspiración a lo largo de mi vida y formación profesional.

Mi sincero agradecimiento a la empresa Black Label Technology Cía. Ltda. por abrirme sus puertas para el desarrollo de este trabajo de titulación, y de manera especial a la Ing. Stefhany Cantarero Cordón y al Ing. Adrián Zurita Amores, por su apoyo, orientación y colaboración durante este proceso.

A la Universidad Técnica de Ambato, por contribuir a mi formación académica y profesional. Asimismo, expreso mi gratitud a mi tutor, Ing. Clay Aldás Flores, y al Ing. Renato Urvina, Mg., por sus conocimientos, guía y valioso acompañamiento en el desarrollo de esta investigación.

Finalmente, agradezco a mis amigos por su apoyo, confianza y palabras de aliento durante esta etapa tan importante de mi vida.

A todos quienes formaron parte de este logro, mi más sincero agradecimiento.

ÍNDICE GENERAL DE CONTENIDOS

PORTADA ..... **i**

APROBACIÓN DEL TUTOR ..... **ii**

AUTORÍA ..... **iii**

DERECHOS DE AUTOR ..... **iv**

APROBACIÓN DEL TRIBUNAL DE GRADO ..... **v**

DEDICATORIA ..... **vi**

AGRADECIMIENTO ..... **vii**

ÍNDICE GENERAL DE CONTENIDOS ..... **viii**

ÍNDICE DE TABLAS ..... **xiii**

ÍNDICE DE FIGURAS ..... **xvi**

ÍNDICE DE ANEXOS ..... **xviii**

RESUMEN EJECUTIVO ..... **xix**

ABSTRACT ..... **xx**

CAPÍTULO I. MARCO TEÓRICO **1**

1.1 Tema de investigación 1

1.1.1 Planteamiento del problema 1

1.2 Antecedentes investigativos 3

1.3 Fundamentación teórica 7

1.3.1 Ingeniería de software 7

1.3.2 Gestión de proyectos de software 8

1.3.3 Metodología ágil 8

1.3.4 Satisfacción del cliente 9

1.3.5 Calidad del producto 9

1.3.6 Productividad del equipo 9

1.3.7 Optimización del desarrollo de software 10

1.3.8 Metodologías Ágiles Híbridas 10

1.3.9 Métricas Ágiles 11

1.4 Objetivos 12

1.4.1 Objetivo general 12

1.4.2 Objetivos específicos 12

CAPÍTULO II. METODOLOGÍA 14

2.1 Materiales 14

2.2 Métodos 14

2.2.1 Modalidad de la investigación 14

a. Investigación de campo 14

b. Investigación documental 15

2.2.2 Población y muestra 15

2.2.3 Recolección de información 16

2.2.4 Procesamiento y análisis de datos 30

CAPÍTULO III. RESULTADOS Y DISCUSIÓN 32

3.1 Diagnóstico de la situación actual del proceso de desarrollo de software 32

3.1.1 Análisis FODA del proceso de desarrollo de software 34

3.2 Análisis de metodologías ágiles y adaptación a la realidad de la empresa 39

3.2.1 Análisis comparativo de metodologías ágiles 40

3.2.2 Diagnóstico de las características del equipo y los proyectos 42

3.2.3 Criterios de selección de prácticas metodológicas 43

3.2.4 Adaptación de roles a la realidad de la empresa 45

3.3 Propuesta de metodología ágil a medida para Blacklabeltech Black Label Technology Cía. Ltda. 47

3.3.1 Componentes de la metodología BLT-Agile 47

3.3.2 Trazabilidad entre debilidades diagnosticadas y soluciones de BLT-Agile 48

3.3.3 Definición de roles adaptados 49

3.3.4 Fases del ciclo de desarrollo BLT-Agile 50

3.3.5 Flujo de trabajo visual (Tablero Kanban) 50

3.3.6 Definición de Hecho (Definition of Done – DoD) 51

3.3.7 Métricas de seguimiento 52

3.4 Resultados del diagnóstico: Limitaciones y oportunidades de mejora detectadas 52

3.5 Resultados del análisis de metodologías ágiles y adaptación a la realidad de la empresa 54

3.6 Implementación de la metodología BLT-Agile 55

3.6.1 Situación antes de la implementación (línea base) 56

3.6.2 Fase 1: Sprint Planning del proyecto PCD00021 57

3.6.3 Gestión visual: Tablero Kanban en Trello 59

3.6.4 Evidencia de Daily Stand-ups 63

3.6.5 Evidencia de revisión de código en pareja 64

3.6.6 Sprint Review y Retrospectiva 66

3.6.7 Resultados comparativos: Antes vs. Después de BLT-Agile 67

3.6.8 Resumen cuantitativo de la optimización 68

3.6.9 Guía de configuración del tablero Trello para BLT-Agile 69

3.7 Discusión 71

CAPÍTULO IV. CONCLUSIONES Y RECOMENDACIONES 73

4.1 Conclusiones 73

4.2 Recomendaciones 75

REFERENCIAS BIBLIOGRÁFICAS 78

ANEXOS 81

ÍNDICE DE TABLAS

Tabla 1. Cadenas de búsqueda en bases de datos científicas. 3

Tabla 2. Criterios de inclusión y exclusión. 5

Tabla 3. Formato de ficha de contenido. 14

Tabla 4. Población de estudio. 15

Tabla 5. Entrevista al presidente ejecutivo de la empresa. 16

Tabla 6. Estadística de fiabilidad. 18

Tabla 7. Escala de la estadística de fiabilidad. 18

Tabla 8. Tabulación de la pregunta 1. 19

Tabla 9. Tabulación de la pregunta 2. 20

Tabla 10. Tabulación de la pregunta 3. 21

Tabla 11. Tabulación de la pregunta 4. 22

Tabla 12. Tabulación de la pregunta 5. 23

Tabla 13. Tabulación de la pregunta 6. 24

Tabla 14. Tabulación de la pregunta 7. 25

Tabla 15. Tabulación de la pregunta 8. 26

Tabla 16. Tabulación de la pregunta 9. 27

Tabla 17. Tabulación de la pregunta 10. 28

Tabla 18. Ficha de contenido N° 1. 29

Tabla 19. Ficha de Contenido N° 2. 29

Tabla 20. Ficha de Contenido N° 3. 30

Tabla 21. Análisis FODA — Blacklabeltech Black Label Technology Cía. Ltda. ... 35

Tabla 22. Limitaciones identificadas y acciones correctivas para su mitigación. 37

Tabla 23. Oportunidades identificadas y acciones estratégicas para su aprovechamiento. 38

Tabla 24. Diagnóstico comparativo de la situación actual vs. oportunidades de mejora. 38

Tabla 25. Análisis comparativo de metodologías ágiles — Scrum, Kanban y Extreme Programming. 40

Tabla 26. Diagnóstico del equipo y los proyectos de Blacklabeltech y su implicación metodológica. 42

Tabla 27. Criterios de selección ponderados de prácticas metodológicas para la metodología propuesta. 44

Tabla 28. Propuesta de adaptación de roles ágiles a la realidad de Blacklabeltech. .. 45

Tabla 29. Componentes de la metodología BLT-Agile. 47

Tabla 30. Matriz de trazabilidad: debilidades diagnosticadas, componentes de BLT-Agile y métricas de verificación. 48

Tabla 31. Roles definidos en la metodología BLT-Agile. 49

Tabla 32. Fases del ciclo de desarrollo BLT-Agile. 50

Tabla 33. Estructura del tablero Kanban en BLT-Agile. 51

Tabla 34. Definición de Hecho (DoD) en BLT-Agile. 51

Tabla 35. Métricas de seguimiento en BLT-Agile. 52

Tabla 36. Línea base del proceso de desarrollo antes de BLT-Agile. 56

Tabla 37. Product Backlog del proyecto PCD00021. 57

Tabla 38. Sprint Backlog del proyecto PCD00021. 58

Tabla 39. Registro de Daily Stand-ups del Sprint 1 – PCD00021. 63

Tabla 40. Registro de revisiones de código – Sprint 1 64

Tabla 41. Retrospectiva del Sprint 1 – PCD00021. 67

Tabla 42. Comparación de métricas: Antes vs. Después de BLT-Agile. 67

Tabla 43. Indicadores clave de optimización con BLT-Agile. 68

Tabla 44. Configuración del tablero Trello. 69

Tabla 45. Comparativa entre antecedentes investigativos y resultados de BLT-Agile en el proyecto. 71

ÍNDICE DE FIGURAS

Figura 1. Diagrama de flujo PRISMA del proceso de selección de literatura. 6

Figura 2. Datos de la pregunta 1. 19

Figura 3. Datos de la pregunta 2. 20

Figura 4. Datos de la pregunta 3. 21

Figura 5. Datos de la pregunta 4. 22

Figura 6. Datos de la pregunta 5. 23

Figura 7. Datos de la pregunta 6. 24

Figura 8. Datos de la pregunta 7. 25

Figura 9. Datos de la pregunta 8. 26

Figura 10. Datos de la pregunta 9. 27

Figura 11. Datos de la pregunta 10. 28

Figura 12. Diagrama de flujo del proceso actual de desarrollo de software con tiempos estimados en la empresa Black Label Technology Cía. Ltda. 33

Figura 13. Diagrama de Gantt del proyecto de desarrollo de mejoras en la administración de la banca (10 días laborables). 34

Figura 14. Tablero de gestión del proyecto en Trello para el seguimiento del Sprint 1 en el desarrollo de Banca Móvil e integración DeUna. 60

Figura 15. Tablero de gestión del proyecto en Trello para el seguimiento del Sprint 1 en el desarrollo de funcionalidades de Banca Móvil e integración con DeUna... 61

Figura 16. Estado de avance del Sprint 1 mediante el tablero de gestión ágil en Trello. 62

Figura 17. Estado final del Sprint 1 en el tablero Trello posterior a la ejecución de las tareas del proyecto..... 63

Figura 18. Registro de revisión y validación técnica de historias de usuario en Trello. 65

Figura 19. Validación final y aprobación de historia de usuario en el proceso de revisión del Sprint 1. 66

Figura 20. Uso de etiquetas en Trello para la clasificación y control de tareas del Sprint 1. 70

ÍNDICE DE ANEXOS

Anexo A. Formato de cuestionario. 81

Anexo B. Guía de entrevista con preguntas abiertas..... 83

RESUMEN EJECUTIVO

La presente investigación tuvo como objetivo implementar una metodología ágil híbrida denominada BLT-Agile para optimizar el proceso de desarrollo de software en la empresa Blacklabeltech Black Label Technology Cía. Ltda. El estudio surgió debido a problemas identificados en la gestión de proyectos, como retrasos en las entregas, limitada identificación de errores, falta de control de actividades y dificultades de adaptación a cambios en los requerimientos del cliente. La metodología empleada tuvo un enfoque mixto, apoyándose en técnicas como encuestas, entrevistas y análisis documental para diagnosticar la situación actual de la empresa y determinar las principales debilidades como la ausencia de herramientas de gestión visual del trabajo, la falta de procesos formales de revisión de código y pruebas de software. Se realizó un análisis comparativo de las metodologías Scrum, Kanban y Extreme Programming, identificando ventajas como la planificación iterativa, la gestión visual de tareas y las prácticas orientadas a la calidad del software; así como limitaciones relacionadas con la falta de flexibilidad de los enfoques individuales para cubrir todas las necesidades de la organización, lo que permitió diseñar una metodología híbrida adaptada al contexto empresarial. La propuesta fue implementada en un proyecto piloto, obteniendo resultados positivos reflejados en la reducción del 37,5 por ciento en los tiempos de entrega y una disminución del 83,3 por ciento en los bugs reportados por el cliente. Esto permitió mejorar la organización del trabajo, la comunicación del equipo, el seguimiento de tareas y el control de calidad mediante métricas ágiles y herramientas visuales.

Palabras clave: Metodologías híbridas, desarrollo de software, Scrum, Kanban, Extreme Programming, Optimización de procesos, Black Label Tech.

ABSTRACT

This research aimed to implement a hybrid agile methodology called BLT-Agile to optimize the software development process at Blacklabeltech Black Label Technology Cía. Ltda. The study was motivated by issues identified in project management, including delivery delays, limited error detection, insufficient activity control, and difficulties in adapting to changes in customer requirements. A mixed-methods approach was employed, supported by surveys, interviews, and document analysis to assess the company's current situation and identify key weaknesses, such as the absence of visual work management tools and the lack of formal code review and software testing processes. A comparative analysis of Scrum, Kanban, and Extreme Programming methodologies was conducted, identifying strengths such as iterative planning, visual task management, and software quality-oriented practices, as well as limitations associated with the inability of individual approaches to fully address all organizational needs. Based on these findings, a hybrid methodology adapted to the company's context was designed. The proposed methodology was implemented in a pilot project and produced positive results, including a 37.5 percent reduction in delivery times and an 83.3 percent decrease in software defects reported by clients. Consequently, improvements were achieved in work organization, team communication, task monitoring, and quality control through the use of agile metrics and visual management tools.

Keywords: Hybrid Methodologies, Software Development, Scrum, Kanban, Extreme Programming, Process Optimization, Black Label Tech.

CAPÍTULO I. MARCO TEÓRICO

1.1 Tema de investigación

"METODOLOGÍA ÁGIL A MEDIDA QUE PERMITA OPTIMIZAR EL DESARROLLO DE SOFTWARE EN LA EMPRESA BLACKLABELTECH BLACK LABEL TECHNOLOGY CIA. LTDA."

1.1.1 Planteamiento del problema

En la actualidad, la industria del desarrollo de software se encuentra en constante evolución debido a la rápida transformación digital y a la creciente demanda de soluciones tecnológicas innovadoras. Esta dinámica obliga a las empresas a mejorar sus procesos de trabajo para responder de forma ágil y eficiente a los cambios del mercado, a las necesidades de los clientes y a los avances tecnológicos.

En este contexto, las metodologías ágiles surgieron como una alternativa a los enfoques tradicionales de desarrollo, como el modelo en cascada, los cuales resultaban poco flexibles ante los cambios de requisitos [1]. A diferencia de los modelos secuenciales, las metodologías ágiles permiten trabajar en ciclos iterativos e incrementales, priorizando la colaboración, la retroalimentación continua y la entrega temprana de valor al cliente [2]. Estas características han convertido a la agilidad en un estándar de facto en las empresas de software de todo el mundo.

Entre las metodologías ágiles más utilizadas destacan Scrum, Kanban y Extreme Programming (XP). Scrum es ampliamente adoptada por su estructura clara de roles, artefactos y eventos [4]; Kanban se centra en la gestión visual del flujo de trabajo y la reducción de cuellos de botella; mientras que XP enfatiza la calidad del código a través de prácticas como la programación en pareja y el desarrollo guiado por pruebas. Sin embargo, muchas organizaciones no aplican estas metodologías de manera estricta, sino que adaptan sus prácticas a su realidad particular. Este enfoque se conoce como metodología ágil a medida, en la que se seleccionan, combinan y personalizan técnicas según el tamaño del equipo, la complejidad de los proyectos, la cultura organizacional y los recursos disponibles [2] [3].

En el caso de Ecuador, la adopción de metodologías ágiles en las empresas de desarrollo de software ha ido en aumento en los últimos años, especialmente en Pequeñas y Medianas Empresas (PYMES), que representan la mayor parte del sector tecnológico. Estas organizaciones suelen enfrentar limitaciones de personal, tiempo y presupuesto, lo que hace necesario optimizar sus procesos internos para mejorar la productividad y garantizar productos de calidad. Sin embargo, uno de los principales retos identificados es que, aunque muchas empresas afirman aplicar metodologías ágiles, en la práctica suelen hacerlo de manera incompleta o inadecuada, limitando así los beneficios reales que estas ofrecen [1].

La empresa Blacklabeltech Black Label Technology Cía. Ltda., ubicada en la ciudad de Ambato, se dedica al desarrollo de software y, como muchas pymes tecnológicas locales, actualmente presenta desafíos en la gestión de proyectos: retrasos en las entregas, baja estandarización de procesos, comunicación limitada entre los miembros del equipo y dificultad para adaptarse a cambios de requisitos en los proyectos.

Frente a esta realidad, implementar una metodología ágil a medida que se adapte específicamente a las características y necesidades de Black Label Tech representa una oportunidad clave para optimizar sus procesos de desarrollo de software. Por lo que, en la empresa se ha identificado que, a pesar de utilizar metodologías ágiles, no se dispone de una metodología ágil estructurada y adaptada específicamente a las necesidades de la organización. Esta situación puede generar dificultades en la gestión de proyectos, así como retrasos en el desarrollo de software y en la entrega de soluciones a los clientes.

Además, la falta de una metodología ágil definida y ajustada al contexto de la empresa puede provocar la pérdida de oportunidades de negocio, considerando que muchas organizaciones demandan procesos de desarrollo claros y bien estructurados como parte de los acuerdos y contratos tecnológicos. Por lo tanto, surge la necesidad de diseñar una metodología ágil a medida que permita optimizar el proceso de desarrollo de software, mejorar la organización del trabajo y fortalecer la competitividad de la empresa en un mercado cada vez más exigente.

1.2 Antecedentes investigativos

Mediante la revisión sistemática de literatura en bases de datos científicas y académicas como Scopus, IEEE Xplore, Google Scholar y ScienceDirect, se identificaron estudios relevantes publicados en los últimos cinco años, de los cuales se obtuvo un total de 4075 artículos, utilizando las cadenas de búsqueda detalladas en la Tabla 1. Los temas centrales abordados fueron:

- Metodologías ágiles y adaptación en entornos específicos.
- Optimización de procesos de desarrollo de software.
- Frameworks ágiles a medida (Scrum, Kanban, XP, híbridos).
- Cultura organizacional y adopción ágil.
- Mejora continua en entornos tecnológicos.

Tabla 1. Cadenas de búsqueda en bases de datos científicas.

Bases de datos	Cadenas de búsqueda	Cantidad artículos
Scopus	TITLE-ABS-KEY(("metodología ágil" OR Agile OR Scrum OR Kanban) AND (desarrollo OR software OR productividad OR calidad)) AND PUBYEAR > 2020	520
	TITLE-ABS-KEY(("tailored agile" OR "custom agile" OR "agile methodology adaptation") AND ("software development optimization" OR "development performance")) AND PUBYEAR > 2020	435
	TITLE-ABS-KEY(("agile methodology" OR "custom agile" OR "tailored framework") AND ("software development" OR "process optimization") AND ("case study" OR "implementation")) AND PUBYEAR > 2020	128
	TITLE-ABS-KEY(("Scrum" OR "Kanban" OR "XP" OR "hybrid agile") AND ("performance" OR "efficiency" OR "productivity") AND "software development")) AND PUBYEAR > 2020	314
	TITLE-ABS-KEY(("customized agile" OR "tailored agile" OR "adapted agile") AND ("software development" OR "software engineering")) AND PUBYEAR > 2020	245
	TITLE-ABS-KEY(("Scrum implementation" AND ("Ecuador" OR "Latin America") AND "case study")) AND PUBYEAR > 2020	60

IEEE Xplore	("Agile methodology" OR Scrum OR Kanban) AND ("software development" OR productividad OR calidad) Refined By: Year: 2019-2025	120
	("custom agile methodology" OR "tailoring agile") AND ("software development optimization" OR "project success") Refined By: Year: 2021-2025	131
	("Agile tailoring" OR "custom agile framework") AND ("software development" OR "process improvement") Refined By: Year: 2021-2025	45
	("Agile metrics" OR "team performance" OR "velocity") AND "software engineering" Refined By: Year: 2021-2025	89
	("Continuous improvement" AND "agile teams" AND "software development") Refined By: Year: 2021-2025	70
	("Hybrid agile frameworks" AND "small development teams") Refined By: Year: 2021-2025	55
Google Scholar	“Kanban en equipos pequeños” AND “eficiencia en desarrollo de software” site:researchgate.net	20
	"metodología ágil a medida" OR "optimización desarrollo software" OR “empresa tecnológica" site:researchgate.net	18
	"agile adoption" OR "SME" OR "software company" OR "case study" site:researchgate.net	22
	“Scrum adaptado” AND “pequeñas empresas de software” site:researchgate.net	17
	“metodologías ágiles” AND “optimización del desarrollo de software” AND “Scrum” site:researchgate.net	31
	“metodología ágil a medida” OR “metodología ágil personalizada” AND “productividad” AND “calidad” site:researchgate.net	14
ScienceDirect	Title, abstract, keywords: ("metodología ágil" OR Scrum OR Kanban) AND (calidad OR productividad) AND "desarrollo de software"; Date Published: 2021-2024	450
	Title, abstract, keywords: ("agile software development" AND "process optimization" AND "tailored"); Date Published: 2021-2025	31
	Title, abstract, keywords: ("agile practices" AND "software productivity" AND "team performance"); Date Published: 2021-2025	450
	Title, abstract, keywords: ("Scrum customization" OR "agile tailoring") AND "software project management"; Date Published: 2021-2024	320
	Title, abstract, keywords: ("agile maturity" AND "software quality improvement"); Date Published: 2021-2025	280

	Title, abstract, keywords: ("hybrid agile" AND "process optimization" AND "software engineering"); Date Published: 2021-2024	210
	Total:	**4075**

Luego de un riguroso proceso de revisión de la literatura recolectada, se establecieron los siguientes criterios de inclusión y exclusión (Tabla 2) los cuales permiten analizar la relevancia de cada uno de ellos en el presente proyecto de investigación.

Tabla 2. Criterios de inclusión y exclusión.

Criterios de inclusión	**Criterios de exclusión**
Investigaciones recientes (últimos 5 años).	Investigaciones previas al 2019.
Investigaciones aplicadas en empresas de desarrollo de software.	Estudios teóricos sin aplicación práctica.
Enfoque en adaptación de metodologías ágiles.	Estudios sin métricas de rendimiento.
Contextos similares a PYMES tecnológicas	Contextos no relacionados con TI.

Con el propósito de documentar de manera transparente y rigurosa el proceso de selección de la literatura, se elaboró el diagrama de flujo PRISMA (Preferred Reporting Items for Systematic Reviews and Meta-Analyses), el cual permite visualizar el número de artículos identificados, filtrados, evaluados e incluidos en la revisión sistemática.

Como se evidencia en la Figura 1, de los 4.075 artículos identificados inicialmente en las bases de datos consultadas, se eliminaron 875 por duplicación entre fuentes. Los 3.200 artículos restantes fueron cribados por título y resumen, descartando 3.160 que no correspondían al área de metodologías ágiles aplicadas en empresas de software. Los 40 artículos restantes fueron evaluados a texto completo, de los cuales 25 fueron excluidos por no cumplir con los criterios de inclusión establecidos en la Tabla 2. Finalmente, 15 estudios fueron seleccionados e incluidos en la revisión sistemática como fundamento de los antecedentes investigativos de la presente investigación.

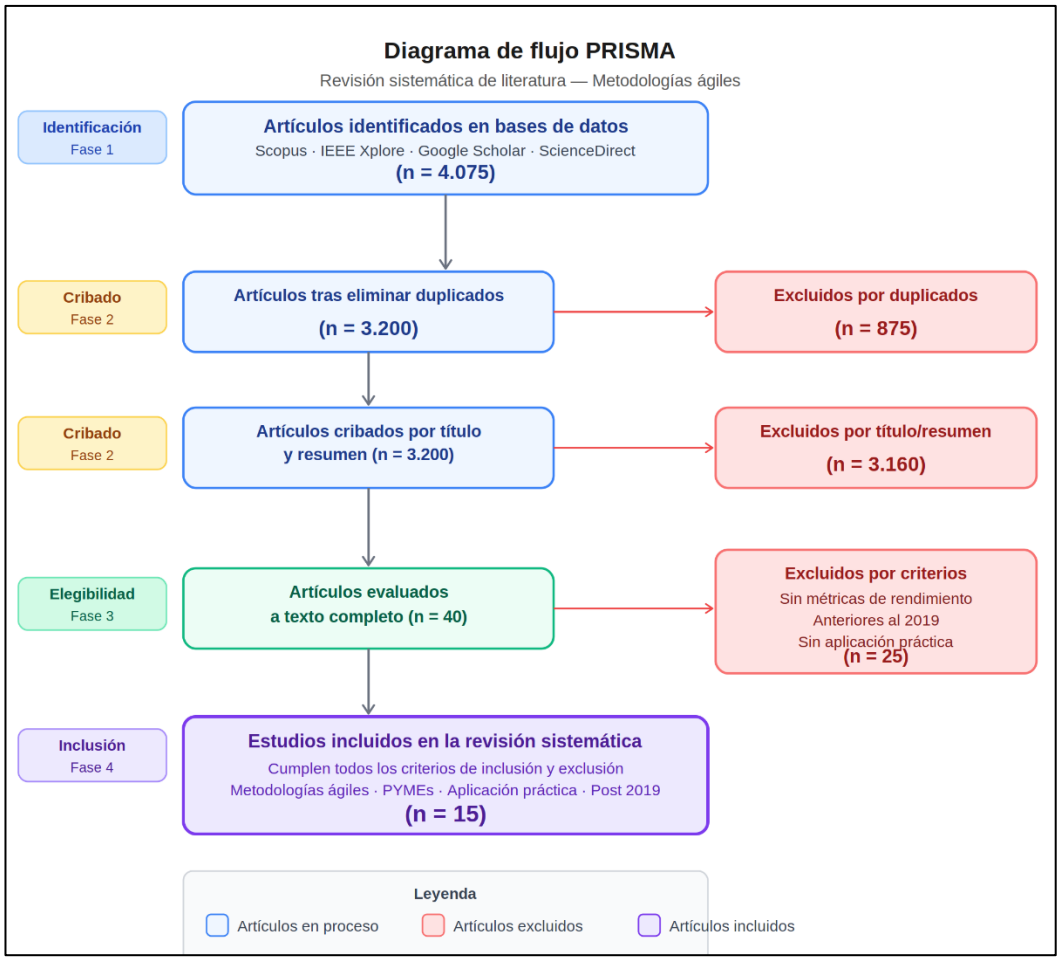


Figura 1. Diagrama de flujo PRISMA del proceso de selección de literatura.

Fuente: Elaboración propia (2025).

La creciente competitividad en el sector tecnológico ha impulsado la adopción de metodologías ágiles para mejorar la flexibilidad, la entrega continua y la calidad del software. Sin embargo, la aplicación genérica de frameworks como Scrum o Kanban no siempre se ajusta a las necesidades específicas de cada organización, especialmente en empresas de tamaño medio como Blacklabeltech Black Label Technology Cía. Ltda., donde la escalabilidad y la cultura interna juegan un papel determinante.

Según Schmidt & López (2021), la adaptación de metodologías ágiles en PYMES tecnológicas incrementa la eficiencia en hasta un 40%, siempre que se alinee con los procesos internos y la capacidad del equipo [7]. Por su parte, Rivera et al. (2022) demostraron que la implementación de un modelo híbrido (Scrum-Kanban) mejoró la

predictibilidad de entregas y la gestión de cambios en proyectos con requisitos volátiles [8].

Scrum, XP y Kanban han demostrado beneficios significativos en términos de productividad del equipo y calidad del software [11]. En el caso de Ecuador, diversos estudios evidencian un crecimiento en la adopción de metodologías ágiles dentro de las empresas de desarrollo de software, siendo Scrum una de las más utilizadas debido a su enfoque colaborativo y su capacidad para adaptarse a entornos cambiantes [1]. Asimismo, Extreme Programming (XP) enfatiza la calidad del software mediante prácticas como las pruebas continuas, la programación en parejas y las entregas frecuentes, contribuyendo a la reducción de errores y a la mejora continua del producto.

Según Gómez y Torres (2020), una empresa de desarrollo de software en Colombia enfrentó problemas de sobrecarga administrativa y resistencia al cambio debido a la falta de personalización en la metodología ágil. Con la aplicación de mejoras como ciclos de retroalimentación más cortos y una mejor definición de roles, se logró una reducción del 25% en los tiempos de entrega [9].

Asimismo, Navarro (2023) destaca que la optimización del desarrollo de software no depende únicamente de la metodología, sino también de factores como la madurez organizacional, las herramientas de apoyo y la formación continua del equipo [10].

Estos antecedentes reflejan la necesidad de diseñar metodologías ágiles a medida que respondan a las particularidades de cada organización, integrando mejores prácticas, herramientas de medición y un enfoque en la mejora continua.

1.3 Fundamentación teórica

1.3.1 Ingeniería de software

La ingeniería de software es una disciplina que aplica principios científicos, matemáticos y de ingeniería para el diseño, desarrollo, implementación, mantenimiento y gestión de sistemas de software de alta calidad. Su objetivo es

garantizar que los productos desarrollados sean confiables, eficientes, seguros y cumplan con las necesidades del usuario [18].

De acuerdo con Sommerville (2020), la ingeniería de software combina métodos técnicos, herramientas y prácticas de gestión que permiten controlar la complejidad de los proyectos, reducir errores y asegurar la calidad del producto final. En el contexto ágil, esta disciplina se adapta mediante prácticas como la integración continua, las pruebas automatizadas y el desarrollo iterativo [18].

1.3.2 Gestión de proyectos de software

La gestión de proyectos de software es el proceso de planificación, organización, ejecución y control de las actividades necesarias para desarrollar un producto de software que cumpla con los objetivos definidos en alcance, tiempo, costo y calidad [16].

Incluye la coordinación del equipo, la asignación de recursos, la gestión de riesgos y la comunicación efectiva con las partes interesadas.

Según Pressman y Maxim (2021), la gestión de proyectos en entornos ágiles busca equilibrar la estructura con la flexibilidad, promoviendo ciclos iterativos de desarrollo y entregas incrementales que garanticen una respuesta oportuna ante cambios de requerimientos [16].

1.3.3 Metodología ágil

Las metodologías ágiles son enfoques de gestión y desarrollo de software que se basan en principios de flexibilidad, colaboración, entrega continua de valor y adaptación al cambio. Su propósito es responder de manera eficiente a las necesidades del cliente mediante iteraciones cortas, retroalimentación constante y mejora continua. Estas metodologías priorizan a las personas y la comunicación por encima de los procesos rígidos, fomentando la autoorganización de los equipos y la generación de valor en cada ciclo de trabajo [14].

Autores como Beck et al. (2021) y Schwaber & Sutherland (2020) señalan que las metodologías ágiles surgen como una alternativa a los enfoques tradicionales de desarrollo, permitiendo obtener resultados de mayor calidad en menor tiempo [14].

1.3.4 Satisfacción del cliente

Es el grado de conformidad del cliente o usuario final con el producto y con el proceso de desarrollo. En entornos ágiles, el cliente forma parte activa del equipo mediante la revisión constante de entregables, lo que permite alinear expectativas y mejorar la percepción de valor.

De acuerdo con Moe, Šmite y Ågerfalk (2020), la colaboración continua entre cliente y equipo de desarrollo es el factor más determinante para garantizar la satisfacción y fidelización a largo plazo [17].

1.3.5 Calidad del producto

Esta categoría mide el grado en que el software desarrollado cumple con los requisitos funcionales y no funcionales establecidos. La calidad abarca aspectos como la confiabilidad, usabilidad, mantenibilidad y rendimiento.

El modelo ISO/IEC 25010 (2011) establece ocho características esenciales de calidad de software, y las metodologías ágiles contribuyen a su cumplimiento mediante entregas iterativas, revisiones constantes y pruebas continuas [16].

1.3.6 Productividad del equipo

Hace referencia al rendimiento y capacidad de entrega del equipo de desarrollo en función del tiempo y los recursos disponibles. En contextos ágiles, la productividad se mide mediante métricas como la velocidad del sprint, la cantidad de tareas completadas o el tiempo promedio de entrega.

Según Rico & Sayani (2021), las metodologías ágiles incrementan la productividad en más del 20 % respecto a los métodos tradicionales, gracias a la priorización dinámica de tareas y la comunicación constante dentro del equipo [15].

1.3.7 Optimización del desarrollo de software

La optimización del desarrollo de software se refiere al proceso sistemático de mejorar la eficiencia, calidad y productividad en todas las etapas del ciclo de vida del software, desde el análisis de requerimientos hasta la implementación y mantenimiento. Este enfoque busca reducir tiempos de desarrollo, minimizar errores, aprovechar adecuadamente los recursos disponibles y garantizar la entrega de productos que cumplan con las necesidades del usuario y los estándares de calidad establecidos [18].

De acuerdo con Sommerville (2019) y Pressman (2020), la optimización del desarrollo de software no solo implica la mejora de herramientas tecnológicas, sino también la adopción de buenas prácticas de ingeniería, metodologías adecuadas y una gestión eficiente del equipo de trabajo. En este sentido, factores como la planificación adecuada, la comunicación efectiva, la automatización de procesos, el control de versiones y la integración continua juegan un papel fundamental en la mejora del rendimiento del proceso de desarrollo [18] [19].

Asimismo, la implementación de enfoques ágiles permite optimizar el desarrollo de software mediante ciclos iterativos e incrementales, facilitando la adaptación a cambios, la retroalimentación constante del cliente y la entrega continua de valor. Esto contribuye a incrementar la productividad del equipo, mejorar la calidad del producto final y elevar el nivel de satisfacción del cliente, convirtiéndose en un elemento clave para la competitividad de las organizaciones en el ámbito tecnológico [19].

1.3.8 Metodologías Ágiles Híbridas

Las metodologías ágiles híbridas son enfoques de gestión de proyectos de software que combinan elementos de distintos frameworks ágiles, como Scrum, Kanban y Extreme Programming, con el propósito de adaptarse a las necesidades específicas de cada organización. A diferencia de aplicar un único framework de manera aislada, estos modelos permiten integrar prácticas seleccionadas estratégicamente para optimizar los procesos de desarrollo y responder mejor a diferentes contextos empresariales [20]. Según Schmidt y López (2021), este enfoque se basa en la teoría de contingencia organizacional, la cual sostiene que no existe un modelo universal de

gestión, sino que la efectividad depende del grado de adaptación entre la organización y la metodología implementada. En este sentido, Scrum aporta estructura y planificación, Kanban facilita la visualización y gestión del flujo de trabajo, mientras que Extreme Programming fortalece la calidad técnica del software [20].

Diversos estudios evidencian beneficios importantes de las metodologías híbridas. Rivera et al. (2022) demostraron que la combinación de Scrum y Kanban mejora la predictibilidad de las entregas y la capacidad de respuesta frente a cambios en los requisitos, logrando un equilibrio entre control y flexibilidad [20]. Entre sus principales características destacan la adaptabilidad selectiva, la coherencia metodológica y la trazabilidad de decisiones, lo que facilita la comprensión y adopción del modelo por parte de los equipos [20]. Asimismo, estas metodologías ofrecen ventajas como una mejor adecuación al contexto organizacional, reducción de costos de implementación y posibilidades de mejora continua; sin embargo, también presentan limitaciones relacionadas con la complejidad de diseño, la falta de estándares de certificación ampliamente reconocidos y las dificultades de escalabilidad en equipos de mayor tamaño [20].

1.3.9 Métricas Ágiles

Las métricas ágiles son herramientas cuantitativas utilizadas para evaluar y mejorar el desarrollo de software dentro de entornos ágiles. A diferencia de las métricas tradicionales, que se enfocan principalmente en el cumplimiento de planes y esfuerzo invertido, las métricas ágiles priorizan aspectos como la entrega continua de valor, la calidad del producto y la capacidad de adaptación frente a cambios [21]. Según Moe et al. (2020), estas métricas son fundamentales debido al principio de visibilidad radical, el cual establece que los problemas del proceso solo pueden identificarse y solucionarse cuando se hacen visibles mediante medición y análisis. Además, facilitan la toma de decisiones basada en datos, la retroalimentación continua y el aprendizaje organizacional [21].

Pressman y Maxim (2021) señalaron que los equipos ágiles que aplican métricas de manera disciplinada logran velocidades de entrega hasta un 25% mayores, ya que pueden detectar y eliminar cuellos de botella de forma continua [21]. Las métricas

ágiles se clasifican en diferentes categorías, entre ellas las métricas de flujo de trabajo, como la velocidad del sprint y el tiempo de ciclo; métricas de calidad, como la densidad de defectos y cobertura de pruebas; métricas de predictibilidad relacionadas con el cumplimiento de compromisos; y métricas de satisfacción orientadas a clientes y stakeholders [21]. Entre las más utilizadas destacan el burndown chart, la tasa de defectos post-entrega y el índice de bloqueos [21].

La implementación de métricas ágiles aporta beneficios importantes, como la detección temprana de problemas, la mejora continua basada en evidencia y una comunicación más transparente con los stakeholders [21]. Asimismo, incrementa la predictibilidad del proyecto, permitiendo comprometerse con plazos de entrega de manera más confiable. Sin embargo, también existen limitaciones, como el riesgo de enfocarse excesivamente en ciertas métricas, la carga administrativa asociada al registro de datos y la posibilidad de interpretar incorrectamente la información obtenida [21]. Por ello, cada organización debe seleccionar las métricas que mejor se adapten a sus necesidades y contexto específico.

1.4 Objetivos

1.4.1 Objetivo general

Implementar una metodología ágil a medida que permita optimizar el desarrollo de software en la empresa Blacklabeltech Black Label Technology Cía. Ltda.

1.4.2 Objetivos específicos

- Determinar las prácticas actuales aplicadas en el desarrollo de software de la empresa Blacklabeltech Black Label Technology Cía. Ltda., para detectar limitaciones y oportunidades de mejora que contribuyan a su proceso productivo.

- Analizar metodologías ágiles conforme a las características y necesidades particulares del equipo de desarrollo y los proyectos, adaptando los roles a la realidad de la empresa.

- Diseñar una metodología ágil que integre las prácticas analizadas y personalizadas, con el fin de optimizar el proceso de desarrollo de software en la empresa.

CAPÍTULO II. METODOLOGÍA

2.1 Materiales

Para el desarrollo de la investigación se utilizaron diversos materiales e instrumentos que permitieron la recopilación y organización de la información necesaria para el estudio. Entre los materiales empleados se encuentran las fichas documentales (Tabla 3) utilizadas en la revisión bibliográfica para el análisis de fuentes académicas relacionadas con metodologías ágiles de desarrollo de software.

Tabla 3. Formato de ficha de contenido.

Ficha de contenido N° #	
Tema	
Año de publicación	
Tesis	
Ideas centrales	
Conceptos clave	
Conclusión	
Aporte al tema	
Observaciones	

Asimismo, se utilizaron instrumentos de recolección de datos como el cuestionario dirigido a los desarrolladores y personal de soporte de la empresa (Anexo A), así como la guía de entrevista aplicada al presidente ejecutivo de la empresa (Anexo B).

2.2 Métodos

2.2.1 Modalidad de la investigación

a. Investigación de campo

La investigación se enmarca en la modalidad de investigación de campo, debido a que permitió obtener información directa, real y específica del entorno de la empresa Blacklabeltech Black Label Technology Cía. Ltda.; para ello se utilizaron encuestas, con el propósito de recopilar de manera cuantificable la percepción de los

desarrolladores sobre la organización del trabajo, la productividad y el uso de metodologías ágiles, y la entrevista dirigida al presidente ejecutivo, con la finalidad de obtener información cualitativa más profunda sobre la gestión de proyectos, la toma de decisiones y las problemáticas existentes en el proceso de desarrollo de software.

b. Investigación documental

Se realizó una investigación bibliográfica-documental que permitiera fundamentar teóricamente el proyecto antes de la implementación, en la cual se emplearon fichas de contenido, debido a que estas permitieron recopilar, organizar y analizar información relevante de fuentes académicas, facilitando la estructura del marco teórico y proporcionando el sustento necesario para el diseño de una metodología ágil a medida orientada a optimizar el desarrollo de software de la empresa.

2.2.2 Población y muestra

La población está constituida por el equipo de desarrollo de software de la empresa Blacklabeltech Black Label Technology Cía. Ltda. Este grupo está conformado por los colaboradores que participan directamente en las actividades de análisis, diseño, programación, pruebas y gestión de proyectos de software.

Dado que la población de la presente investigación no supera las diez personas, se consideró a toda la población como muestra de estudio.

Esta selección garantiza información representativa y confiable, ya que se incluyó a todos los participantes que intervienen directamente en los procesos de desarrollo y gestión de proyectos.

Tabla 4. Población de estudio.

Población	Número	Porcentaje
Gerente	1	12.50 %
Presidente ejecutivo	1	12.50 %
Desarrolladores de software	5	62.50 %
Analista de procesos	1	12.50 %
Total	8	100.00 %

2.2.3 Recolección de información

El presente proyecto se fundamenta principalmente en una investigación de carácter bibliográfico-documental y aplicada, por lo cual se emplearon fichas de contenido y técnicas de recolección de datos como la encuesta y la entrevista, dirigidas a los desarrolladores y al presidente ejecutivo de la empresa Blacklabeltech Black Label Technology Cía. Ltda., por lo tanto, este procedimiento permitió contar con una base estructurada de información teórica y práctica que sirvió de apoyo para el análisis de la situación actual y para el diseño de la metodología ágil híbrida propuesta orientada a optimizar el desarrollo de software en la empresa.

a. Entrevista al presidente ejecutivo de la empresa

Para el análisis de datos, se inició con la información obtenida mediante la entrevista aplicada al presidente ejecutivo de la empresa Blacklabeltech Black Label Technology Cía. Ltda., la cual fue realizada el miércoles 25 de marzo del presente año. Esta técnica fue empleada con el propósito de obtener información cualitativa detallada desde una perspectiva estratégica, permitiendo conocer la forma en que se gestionan los proyectos, la aplicación de metodologías ágiles y las principales problemáticas presentes en el proceso de desarrollo de software.

Tabla 5. Entrevista al presidente ejecutivo de la empresa.

Número	Pregunta	Respuesta
1	¿Cómo se implementó la metodología ágil Scrum dentro de la empresa Black Label Technology Cía. Ltda.?	La metodología Scrum se implementó de manera progresiva como una alternativa para mejorar la organización de los proyectos de desarrollo de software. Sin embargo, su aplicación no se encuentra completamente estandarizada, ya que depende en gran medida de la experiencia del equipo y no se sigue una estructura formal en todos los proyectos.
Conclusión		
Se evidencia que la implementación de Scrum no está completamente estructurada ni adaptada a la empresa, lo que genera inconsistencias en su aplicación y limita su efectividad en el desarrollo de software.		
2	¿Cuáles han sido los principales beneficios que ha observado desde la implementación de Scrum en la gestión de proyectos?	Entre los principales beneficios se destaca una mejor organización de las tareas y una mayor comunicación entre los miembros del equipo. Además, se

Número	Pregunta	Respuesta
		ha logrado una mejor distribución del trabajo en algunos proyectos.
Conclusión		
Scrum ha generado mejoras en la organización y comunicación del equipo, lo que evidencia que la empresa cuenta con una base ágil, aunque aún existen oportunidades para optimizar su aplicación.		
3	¿Qué dificultades o desafíos se han presentado al aplicar la metodología Scrum en el equipo de trabajo?	Se han presentado dificultades en la planificación de los sprints, en la identificación oportuna de problemas y en la adaptación a cambios durante el desarrollo. Además, no siempre se cumplen todas las prácticas de Scrum debido a la carga de trabajo y la dinámica del equipo.
Conclusión		
Existen limitaciones en la implementación de Scrum, principalmente en la planificación, control y adaptación, lo que evidencia la necesidad de mejorar su aplicación mediante una metodología más estructurada.		
4	Desde su perspectiva, ¿Cómo ha influido Scrum en la productividad y organización del equipo?	Scrum ha contribuido parcialmente a mejorar la organización del equipo, pero no ha tenido un impacto significativo en la productividad, ya que aún se presentan retrasos y dificultades en el cumplimiento de tareas.
Conclusión		
Aunque Scrum ha mejorado la organización del equipo, no ha logrado optimizar completamente la productividad, lo que evidencia la necesidad de fortalecer el proceso de desarrollo.		
5	¿Qué mejoras considera necesarias para optimizar la aplicación de la metodología Scrum en la empresa?	Se considera necesario contar con una metodología más estructurada y adaptada a la realidad de la empresa, que permita mejorar la planificación, el seguimiento de tareas y la calidad del software, además de integrar herramientas que faciliten el control del trabajo.
Conclusión		
Se evidencia la necesidad de diseñar e implementar una metodología ágil a medida que permita optimizar la gestión de proyectos y mejorar el desarrollo de software en la empresa.		

b. Encuesta realizada a desarrolladores de la empresa

Validación de encuesta

La confiabilidad del instrumento de recolección de datos fue evaluada mediante el coeficiente Alfa de Cronbach, el cual es una medida estadística utilizada para determinar el grado de consistencia interna de un cuestionario, es decir, permite verificar qué tan coherentes y relacionados están los ítems que lo conforman en función de medir una misma variable. Esta validación es importante porque garantiza

que las preguntas de la encuesta sean fiables y que los resultados obtenidos reflejen de manera precisa la percepción de los encuestados.

En el presente estudio, la aplicación del Alfa de Cronbach permitió comprobar que el instrumento posee un nivel adecuado de confiabilidad, lo que respalda la calidad de los datos recopilados y brinda mayor validez al análisis e interpretación de los resultados. Por tanto, el uso de este método asegura que la información obtenida sea consistente y útil para sustentar el diagnóstico de la situación actual y el diseño de la propuesta de una metodología ágil a medida para optimizar el desarrollo de software en la empresa.

Tabla 6. Estadística de fiabilidad.

Estadísticas de fiabilidad de elemento	
	Alfa de Cronbach
P1	0.919
P2	0.877
P3	0.896
P4	0.903
P5	0.871
P6	0.886
P7	0.885
P8	0.895
P9	0.891
P10	0.897

El análisis de fiabilidad del instrumento aplicado se realizó mediante el coeficiente Alfa de Cronbach, obteniéndose un valor de 0.903 para la escala general, lo cual indica un nivel de confiabilidad excelente. Este resultado evidencia que los ítems del cuestionario presentan una alta consistencia interna y están adecuadamente correlacionados para medir la variable de estudio. Asimismo, el análisis individual de los ítems muestra que, en caso de eliminar alguno de ellos, los valores del Alfa de Cronbach se mantienen en rangos similares (entre 0.871 y 0.919), lo que confirma que ninguna pregunta afecta negativamente la fiabilidad del instrumento.

Tabla 7. Escala de la estadística de fiabilidad.

Estadísticas de Fiabilidad de Escala	
	Alfa de Cronbach
Escala	0.903

En consecuencia, se determina que la encuesta es confiable y adecuada para la recolección de datos, garantizando la validez de los resultados obtenidos y su utilidad

para el análisis del proceso de desarrollo de software y la propuesta de mejora planteada en la investigación.

Se aplicó una encuesta dirigida a los desarrolladores de la empresa con el objetivo de identificar su percepción sobre el uso de la metodología Scrum, la organización del trabajo, la comunicación del equipo y la eficiencia en la gestión de proyectos.

Pregunta 1: ¿La metodología Scrum facilita la organización de las tareas dentro del equipo de trabajo?

Tabla 8. Tabulación de la pregunta 1.

Respuesta	Frecuencia	Porcentaje
Totalmente en desacuerdo	0	0%
En desacuerdo	0	0%
Ni de acuerdo, ni en desacuerdo	2	25%
De acuerdo	6	75%
Totalmente de acuerdo	0	0%
Total	**8**	**100%**

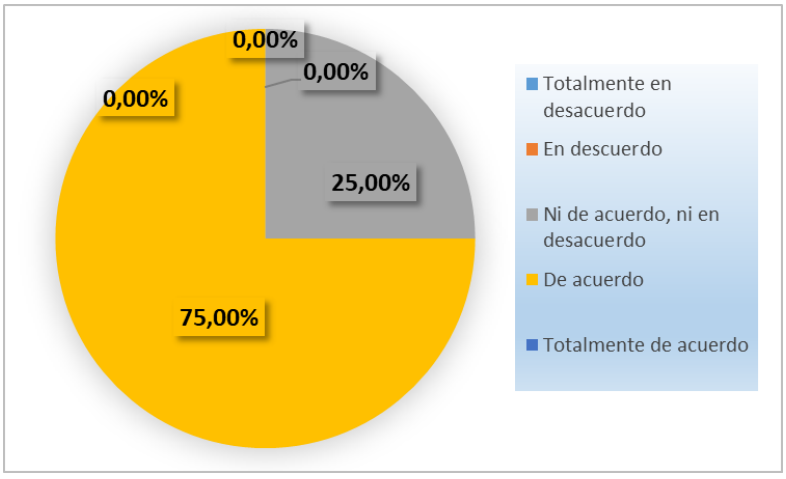


Figura 2. Datos de la pregunta 1.

Conclusión:

De acuerdo con los resultados obtenidos en la pregunta 1, se evidencia que el 75% de los encuestados está de acuerdo en que la metodología Scrum facilita la organización de las tareas dentro del equipo de trabajo, mientras que el 25% mantiene una posición neutral. No se registraron respuestas en las opciones de desacuerdo ni totalmente de acuerdo. Por lo que, estos resultados indican que existe una percepción positiva sobre

el uso de Scrum en la organización de las actividades. Sin embargo, la presencia de respuestas neutrales sugiere que aún hay aspectos que pueden mejorarse en su aplicación. En este sentido, se justifica la implementación de una metodología ágil híbrida que permita fortalecer la organización del trabajo y optimizar el proceso de desarrollo de software en la empresa.

Pregunta 2: ¿La planificación de los sprints permite distribuir adecuadamente las actividades del proyecto?

Tabla 9. Tabulación de la pregunta 2.

Respuesta	Frecuencia	Porcentaje
Totalmente en desacuerdo	0	0%
En desacuerdo	0	0%
Ni de acuerdo, ni en desacuerdo	3	37,5%
De acuerdo	1	12,5%
Totalmente de acuerdo	4	50%
Total	**8**	**100%**

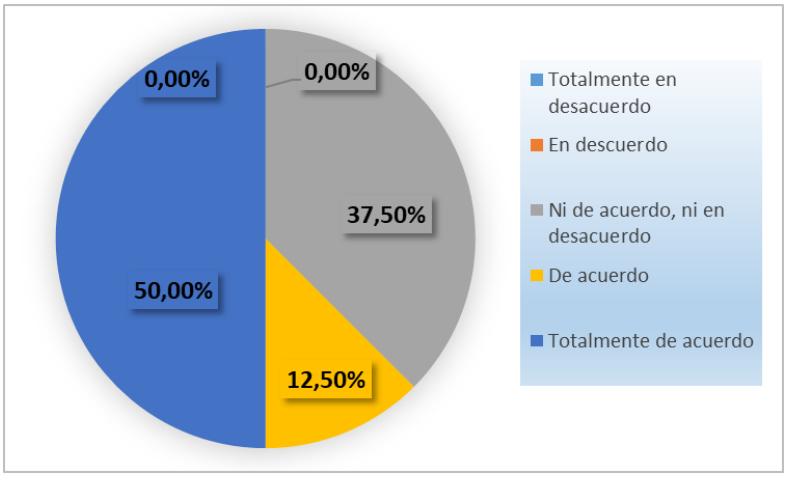


Figura 3. Datos de la pregunta 2.

Conclusión:

De acuerdo con los resultados obtenidos en la pregunta 2, se observa que el 50% de los encuestados está totalmente de acuerdo en que la planificación de los sprints permite distribuir adecuadamente las actividades del proyecto, mientras que un 12,5% está de acuerdo. Por otro lado, el 37,5% mantiene una posición neutral y no se registran respuestas en desacuerdo. Estos resultados reflejan una percepción mayoritariamente positiva respecto a la planificación de los sprints. Sin embargo, el porcentaje de

respuestas neutrales indica que aún existen oportunidades de mejora en la forma en que se distribuyen las actividades. En este sentido, se evidencia la necesidad de fortalecer la planificación mediante una metodología ágil más estructurada que optimice la gestión del proyecto.

Pregunta 3: ¿Las reuniones de seguimiento (Daily Scrum) mejoran la comunicación entre los miembros del equipo?

Tabla 10. Tabulación de la pregunta 3.

Respuesta	Frecuencia	Porcentaje
Totalmente en desacuerdo	0	0%
En desacuerdo	0	0%
Ni de acuerdo, ni en desacuerdo	2	25%
De acuerdo	3	37,5%
Totalmente de acuerdo	3	37,5%
Total	**8**	**100%**

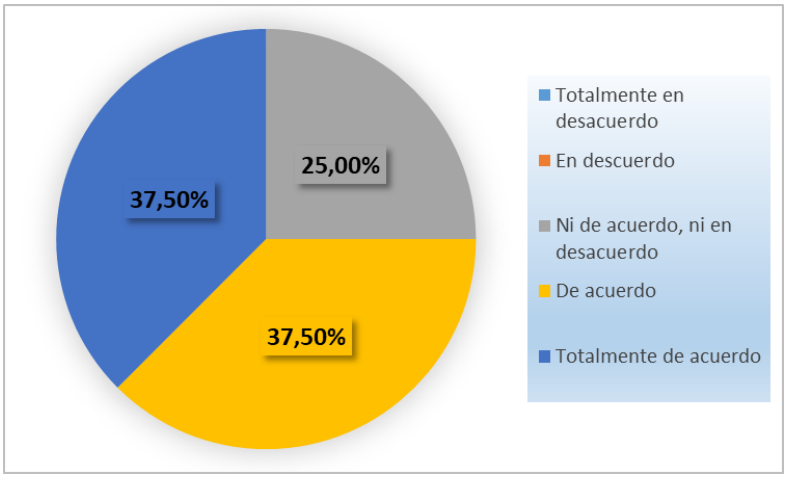


Figura 4. Datos de la pregunta 3.

Conclusión:

De acuerdo con los resultados obtenidos en la pregunta 3, se evidencia que el 37,5% de los encuestados está de acuerdo y otro 37,5% totalmente de acuerdo en que las reuniones de seguimiento (Daily Scrum) mejoran la comunicación entre los miembros del equipo, sumando un 75% de percepción positiva. Por otro lado, el 25% mantiene una posición neutral y no se registran respuestas en desacuerdo. Estos resultados indican que las reuniones diarias son valoradas como un elemento clave para fortalecer la comunicación del equipo. Sin embargo, la presencia de respuestas neutrales sugiere

que aún existen aspectos que pueden optimizarse. En este sentido, se justifica reforzar su aplicación dentro de una metodología ágil más estructurada.

Pregunta 4: ¿El uso de Scrum permite identificar problemas en el proyecto de manera oportuna?

Tabla 11. Tabulación de la pregunta 4.

Respuesta	Frecuencia	Porcentaje
Totalmente en desacuerdo	0	0%
En desacuerdo	5	62,5%
Ni de acuerdo, ni en desacuerdo	1	12,5%
De acuerdo	2	25%
Totalmente de acuerdo	0	0%
Total	**8**	**100%**

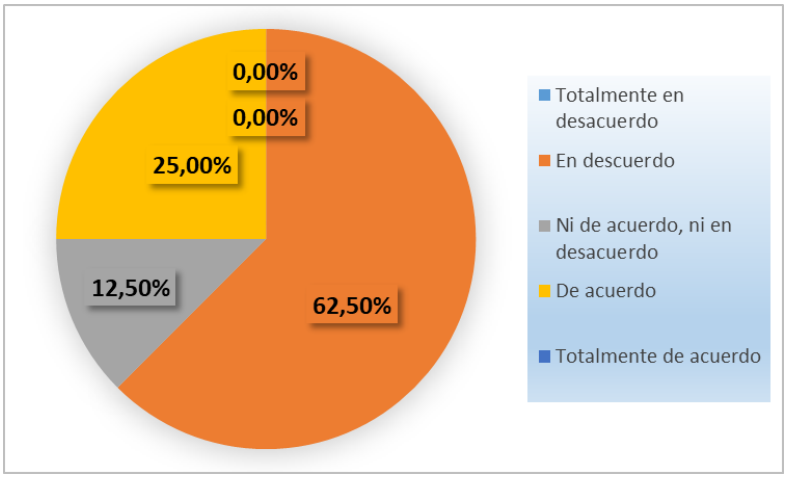


Figura 5. Datos de la pregunta 4.

Conclusión:

De acuerdo con los resultados obtenidos en la pregunta 4, se observa que el 62,5% de los encuestados está en desacuerdo con que el uso de Scrum permite identificar problemas en el proyecto de manera oportuna, mientras que un 25% está de acuerdo y el 12,5% mantiene una posición neutral. No se registran respuestas en las opciones de totalmente en desacuerdo ni totalmente de acuerdo. Estos resultados evidencian una percepción mayoritariamente negativa respecto a la capacidad de Scrum para detectar problemas a tiempo. Esto sugiere deficiencias en su implementación dentro de la empresa. En este sentido, se justifica la necesidad de fortalecer el proceso mediante

una metodología ágil híbrida que permita mejorar la identificación y gestión de problemas en los proyectos.

Pregunta 5: ¿La metodología Scrum facilita la adaptación a cambios en los requerimientos del proyecto?

Tabla 12. Tabulación de la pregunta 5.

Respuesta	Frecuencia	Porcentaje
Totalmente en desacuerdo	1	12,5%
En desacuerdo	2	25%
Ni de acuerdo, ni en desacuerdo	3	37,5%
De acuerdo	2	25%
Totalmente de acuerdo	0	0%
Total	**8**	**100%**

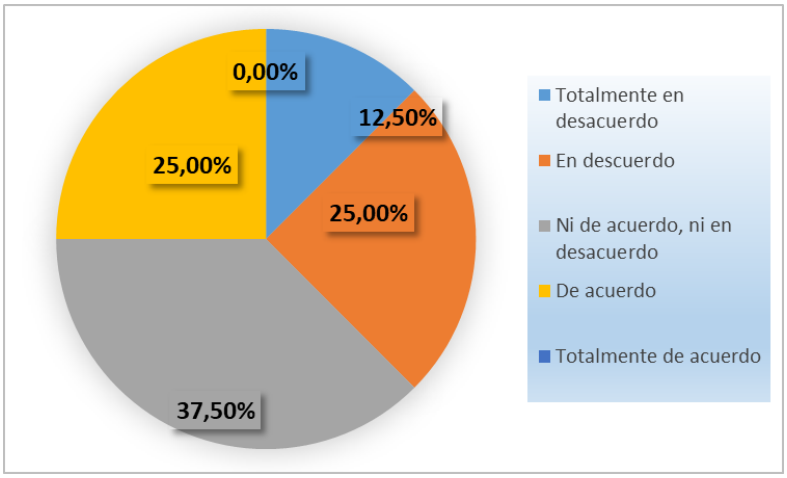


Figura 6. Datos de la pregunta 5.

Conclusión:

De acuerdo con los resultados obtenidos en la pregunta 5, se observa que el 37,5% de los encuestados mantiene una posición neutral respecto a si la metodología Scrum facilita la adaptación a cambios en los requerimientos del proyecto. Por otro lado, un 25% está de acuerdo, mientras que un 25% se encuentra en desacuerdo y un 12,5% totalmente en desacuerdo. No se registran respuestas en la opción totalmente de acuerdo. Estos resultados evidencian una percepción dividida entre los desarrolladores, lo que sugiere que Scrum no está siendo aplicado de manera efectiva para gestionar cambios en los proyectos. En este sentido, se justifica la necesidad de

implementar una metodología ágil híbrida que fortalezca la adaptabilidad y mejore la gestión de requerimientos en la empresa.

Pregunta 6: ¿El uso de Scrum contribuye a mejorar la productividad del equipo de trabajo?

Tabla 13. Tabulación de la pregunta 6.

Respuesta	Frecuencia	Porcentaje
Totalmente en desacuerdo	0	0%
En desacuerdo	2	25%
Ni de acuerdo, ni en desacuerdo	4	50%
De acuerdo	2	25%
Totalmente de acuerdo	0	0%
Total	**8**	**100%**

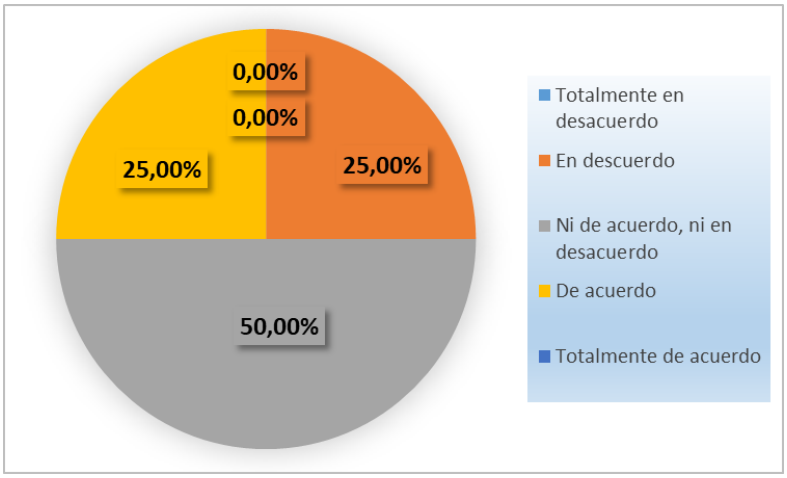


Figura 7. Datos de la pregunta 6.

Conclusión:

De acuerdo con los resultados obtenidos en la pregunta 6, se evidencia que el 50% de los encuestados mantiene una posición neutral respecto a si el uso de Scrum contribuye a mejorar la productividad del equipo de trabajo. Por otro lado, un 25% está de acuerdo, mientras que otro 25% se encuentra en desacuerdo, sin registrarse respuestas en los extremos. Estos resultados reflejan una percepción poco definida sobre el impacto de Scrum en la productividad, lo que sugiere que su implementación no está generando resultados claros en este aspecto. En este sentido, se justifica la necesidad de optimizar su aplicación mediante una metodología ágil híbrida que permita mejorar la eficiencia y el rendimiento del equipo de desarrollo.

Pregunta 7: ¿La metodología Scrum permite entregar resultados de forma más rápida y eficiente?

Tabla 14. Tabulación de la pregunta 7.

Respuesta	Frecuencia	Porcentaje
Totalmente en desacuerdo	0	0%
En desacuerdo	3	37,5%
Ni de acuerdo, ni en desacuerdo	5	62,5%
De acuerdo	0	0%
Totalmente de acuerdo	0	0%
Total	**8**	**100%**

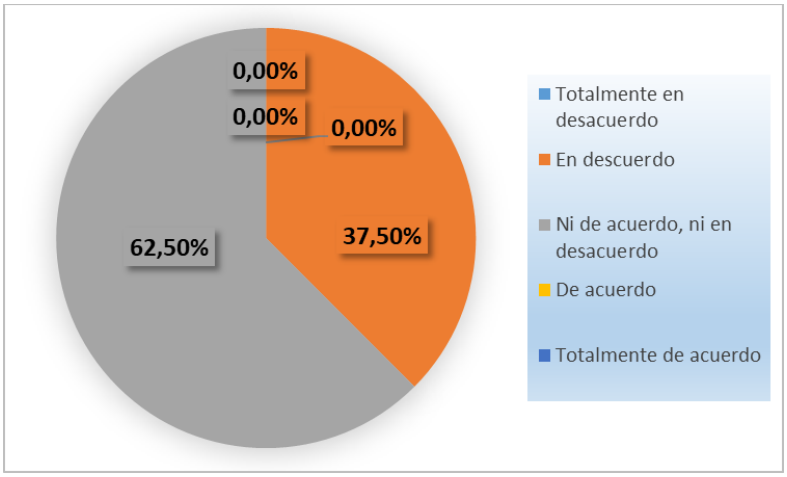


Figura 8. Datos de la pregunta 7.

Conclusión:

De acuerdo con los resultados obtenidos en la pregunta 7, se observa que el 62,5% de los encuestados mantiene una posición neutral respecto a si la metodología Scrum permite entregar resultados de forma más rápida y eficiente, mientras que un 37,5% está en desacuerdo. No se registran respuestas en las opciones de acuerdo ni totalmente de acuerdo. Estos resultados evidencian que la mayoría de los desarrolladores no percibe claramente beneficios en términos de rapidez y eficiencia en la entrega de resultados. Esto sugiere limitaciones en la aplicación de Scrum dentro de la empresa. En este sentido, se justifica la necesidad de implementar una metodología ágil híbrida que permita mejorar la eficiencia en la entrega de los proyectos.

Pregunta 8: ¿El uso de Scrum mejora la coordinación entre los miembros del equipo?

Tabla 15. Tabulación de la pregunta 8.

Respuesta	Frecuencia	Porcentaje
Totalmente en desacuerdo	0	0%
En desacuerdo	1	12,5%
Ni de acuerdo, ni en desacuerdo	4	50%
De acuerdo	3	37,5%
Totalmente de acuerdo	0	0%
Total	**8**	**100%**

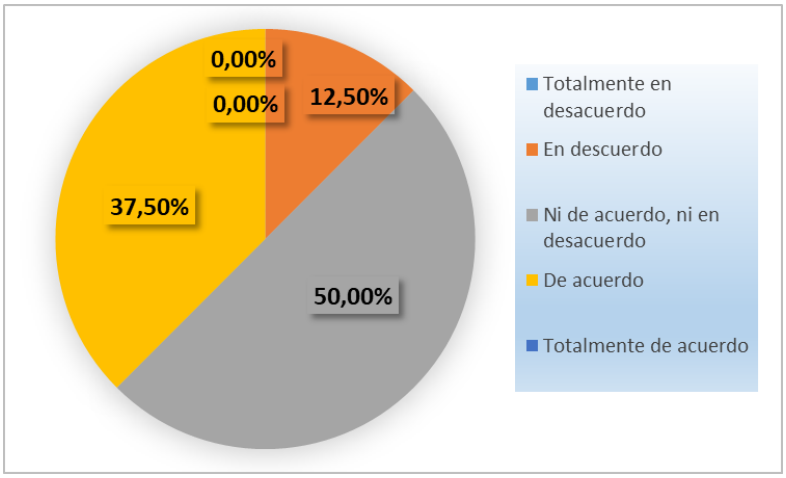


Figura 9. Datos de la pregunta 8.

Conclusión:

De acuerdo con los resultados obtenidos en la pregunta 8, se evidencia que el 50% de los encuestados mantiene una posición neutral respecto a si el uso de Scrum mejora la coordinación entre los miembros del equipo, mientras que un 37,5% está de acuerdo y un 12,5% en desacuerdo. No se registran respuestas en las opciones totalmente en desacuerdo ni totalmente de acuerdo. Estos resultados reflejan que, aunque existe una percepción parcialmente positiva, no es lo suficientemente contundente, lo que sugiere que la coordinación del equipo aún puede mejorarse. En este sentido, se justifica la implementación de una metodología ágil híbrida que permita fortalecer la coordinación y el trabajo colaborativo dentro del equipo de desarrollo.

Pregunta 9: ¿La aplicación de Scrum contribuye a mejorar la calidad de los proyectos desarrollados?

Tabla 16. Tabulación de la pregunta 9.

Respuesta	Frecuencia	Porcentaje
Totalmente en desacuerdo	0	0%
En desacuerdo	5	62,5%
Ni de acuerdo, ni en desacuerdo	2	25%
De acuerdo	1	12,5%
Totalmente de acuerdo	0	0%
Total	**8**	**100%**

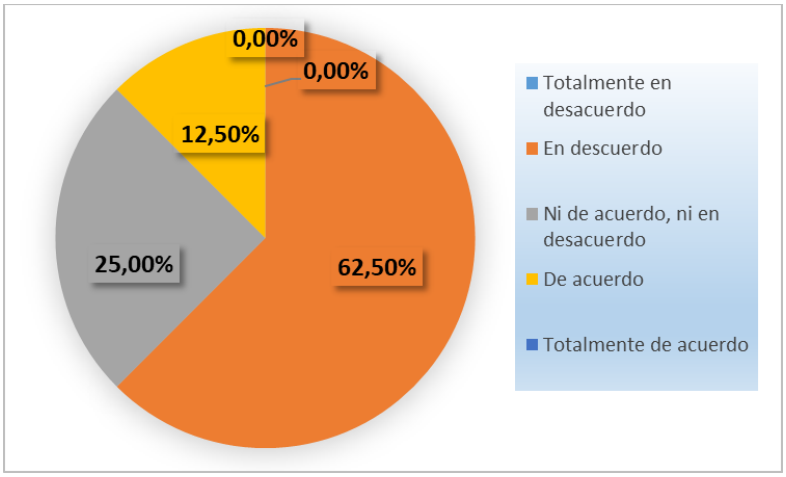


Figura 10. Datos de la pregunta 9.

Conclusión:

De acuerdo con los resultados obtenidos en la pregunta 9, se observa que el 62,5% de los encuestados está en desacuerdo con que la aplicación de Scrum contribuya a mejorar la calidad de los proyectos desarrollados, mientras que un 25% mantiene una posición neutral y solo un 12,5% está de acuerdo. No se registran respuestas en las opciones totalmente en desacuerdo ni totalmente de acuerdo. Estos resultados evidencian una percepción mayoritariamente negativa respecto al impacto de Scrum en la calidad del software. Esto sugiere deficiencias en su implementación dentro de la empresa. En este sentido, se justifica la necesidad de proponer una metodología ágil híbrida que permita mejorar la calidad de los proyectos desarrollados.

Pregunta 10: ¿Considero que la metodología Scrum es adecuada para la gestión de proyectos tecnológicos en la empresa?

Tabla 17. Tabulación de la pregunta 10.

Respuesta	Frecuencia	Porcentaje
Totalmente en desacuerdo	0	0%
En desacuerdo	3	37,5%
Ni de acuerdo, ni en desacuerdo	4	50%
De acuerdo	1	12,5%
Totalmente de acuerdo	0	0%
Total	**8**	**100%**

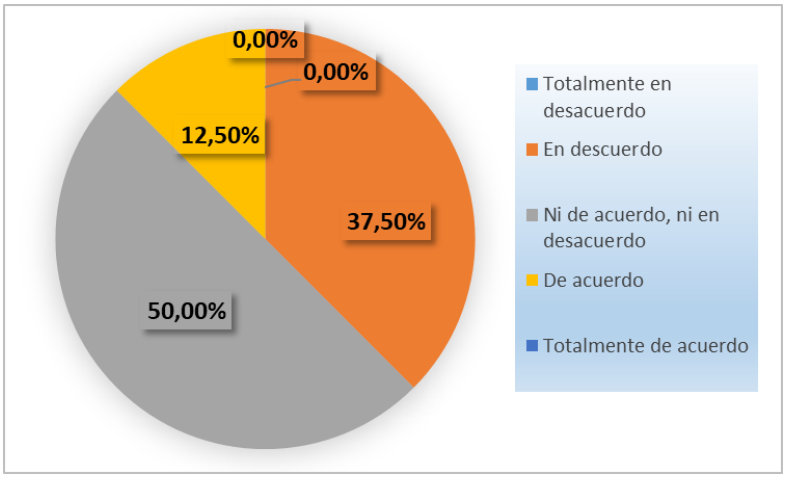


Figura 11. Datos de la pregunta 10.

Conclusión:

De acuerdo con los resultados obtenidos en la pregunta 10, se evidencia que el 50% de los encuestados mantiene una posición neutral respecto a si la metodología Scrum es adecuada para la gestión de proyectos tecnológicos en la empresa, mientras que un 37,5% está en desacuerdo y solo un 12,5% está de acuerdo. No se registran respuestas en las opciones totalmente en desacuerdo ni totalmente de acuerdo. Estos resultados reflejan una percepción poco favorable sobre la adecuación de Scrum en el contexto actual de la empresa. Esto sugiere que su implementación no está respondiendo completamente a las necesidades del equipo. En este sentido, se justifica la propuesta de una metodología ágil híbrida que permita optimizar la gestión de proyectos tecnológicos.

c. Fichas de contenido

Se emplearon fichas de contenido como instrumento metodológico para la recopilación, organización y análisis de información proveniente de fuentes académicas relacionadas con metodologías ágiles de desarrollo de software

Tabla 18. Ficha de contenido N° 1.

Ficha de contenido N° 1	
Tema	Metodología ágil Scrum aplicada al desarrollo de software.
Año de publicación	2020
Tesis	La metodología Scrum permite gestionar proyectos de desarrollo de software mediante ciclos iterativos llamados sprints, facilitando la organización del equipo, la planificación de tareas y la entrega continua de funcionalidades.
Ideas centrales	Scrum organiza el trabajo mediante iteraciones cortas denominadas sprints. Define roles específicos dentro del equipo como Product Owner, Scrum Master y equipo de desarrollo. Promueve la comunicación constante y la mejora continua del proceso.
Conceptos clave	Scrum, Sprint, Product Backlog, Sprint Backlog, Daily Scrum, desarrollo ágil.
Conclusión	La implementación de Scrum mejora la organización del trabajo y facilita el seguimiento del progreso de los proyectos de software, permitiendo adaptarse a cambios durante el proceso de desarrollo.
Aporte al tema	Proporciona fundamentos teóricos sobre la aplicación de Scrum como metodología ágil para mejorar la gestión de proyectos de desarrollo de software.
Observaciones	Scrum es una de las metodologías ágiles más utilizadas en empresas tecnológicas debido a su flexibilidad y facilidad de implementación.

Tabla 19. Ficha de Contenido N° 2.

Ficha de contenido N° 2	
Tema	Uso de Kanban para la gestión visual de proyectos de software.
Año de publicación	2019
Tesis	Kanban es una metodología ágil que permite visualizar el flujo de trabajo mediante tableros de tareas, facilitando la gestión de actividades y la identificación de cuellos de botella en el desarrollo de software.
Ideas centrales	Kanban utiliza tableros visuales para representar las tareas del proyecto. Permite controlar el flujo de trabajo y mejorar la organización del equipo. Facilita la identificación de retrasos en las actividades del desarrollo.
Conceptos clave	Kanban, tablero de tareas, flujo de trabajo, gestión visual, control de tareas.
Conclusión	El uso de Kanban mejora la visibilidad del progreso del proyecto y contribuye a una mejor organización de las actividades dentro del equipo de desarrollo.
Aporte al tema	Apoya la integración de herramientas de visualización del trabajo dentro de la metodología híbrida propuesta para optimizar el desarrollo de software.
Observaciones	Kanban es frecuentemente utilizado como complemento de otras metodologías ágiles como Scrum.

Tabla 20. Ficha de Contenido N° 3.

Ficha de Contenido N° 3	
Tema	Extreme Programming (XP) y la mejora de la calidad del software.
Año de publicación	2021
Tesis	Extreme Programming es una metodología ágil enfocada en mejorar la calidad del software mediante prácticas de programación que promueven la colaboración entre desarrolladores y la integración continua del código.
Ideas centrales	XP promueve prácticas como programación en pareja y desarrollo guiado por pruebas. Busca reducir errores mediante revisiones constantes del código. Fomenta la mejora continua en el proceso de desarrollo.
Conceptos clave	Extreme Programming, programación en pareja, pruebas de software, integración continua, calidad del software.
Conclusión	La aplicación de prácticas de Extreme Programming contribuye a mejorar la calidad del software y a reducir errores durante el proceso de desarrollo.
Aporte al tema	Apoya la incorporación de prácticas orientadas a mejorar la calidad del código dentro de la metodología híbrida propuesta.
Observaciones	XP suele integrarse con otras metodologías ágiles para fortalecer el proceso de desarrollo de software.

2.2.4 Procesamiento y análisis de datos

El procesamiento y análisis de los datos se realizó a partir de la información obtenida mediante la revisión bibliográfica, la aplicación de encuestas al equipo de desarrollo y la entrevista realizada al presidente ejecutivo de la empresa Blacklabeltech Black Label Technology Cía. Ltda. Estas herramientas permitieron identificar la situación actual del proceso de desarrollo de software y fundamentar la propuesta de una metodología ágil híbrida orientada a su optimización.

En relación con la revisión bibliográfica, y según lo registrado en las fichas (Ficha 1, 2 y 3) de contenido elaboradas, se evidenció que metodologías ágiles como Scrum, Kanban y Extreme Programming aportan elementos clave para mejorar el desarrollo de software. En particular, las fichas analizadas destacan que Scrum facilita la organización del trabajo mediante sprints, Kanban permite visualizar el flujo de tareas y detectar cuellos de botella, y XP contribuye a mejorar la calidad del software mediante buenas prácticas de programación. Estos resultados teóricos evidencian la necesidad de integrar estos enfoques en una metodología más completa y adaptada al contexto de la empresa.

Por otra parte, según la entrevista aplicada al presidente ejecutivo, específicamente en las preguntas relacionadas con la gestión de proyectos y dificultades del proceso, se determinó que, aunque la empresa utiliza Scrum, existen limitaciones en su implementación, tales como problemas en la identificación oportuna de errores, dificultades en la planificación de actividades y falta de estandarización en el desarrollo. Por lo que, se evidenció la necesidad de contar con una metodología más estructurada que permita optimizar la organización del trabajo y mejorar la eficiencia en la gestión de proyectos tecnológicos.

En cuanto a los resultados de la encuesta, se observó que, en la pregunta 1, el 75% de los encuestados está de acuerdo en que Scrum facilita la organización de tareas, lo que evidencia una base positiva en su aplicación. Sin embargo, en preguntas como la 6, el 50% de los encuestados se mantiene neutral respecto a su impacto en la productividad, y en la pregunta 9, el 62,5% manifiesta estar en desacuerdo en que Scrum mejora la calidad del software. De igual forma, en la pregunta 7, el 62,5% no percibe mejoras en la rapidez y eficiencia en la entrega de resultados. Estos datos reflejan que, aunque existe una aceptación parcial de la metodología, también se identifican debilidades importantes en su implementación.

En conjunto, el análisis de los datos obtenidos permitió determinar que, si bien la empresa cuenta con una base metodológica ágil, esta no está siendo aplicada de manera eficiente, lo que limita su impacto en variables clave como la productividad, la calidad y la gestión de proyectos. En este sentido, se evidencia la necesidad de proponer una metodología ágil híbrida que integre las fortalezas de Scrum, Kanban y Extreme Programming, con el fin de optimizar el desarrollo de software, mejorar la organización del trabajo y fortalecer la eficiencia del equipo, en concordancia con los objetivos planteados en la investigación.

CAPÍTULO III. RESULTADOS Y DISCUSIÓN

3.1 Diagnóstico de la situación actual del proceso de desarrollo de software

El análisis de los datos obtenidos mediante la encuesta aplicada a los desarrolladores, la entrevista al presidente ejecutivo y la revisión de fichas de contenido permitió identificar la situación actual del proceso de desarrollo de software en la empresa Blacklabeltech Black Label Technology Cía. Ltda., y establecer con precisión las principales brechas y oportunidades de mejora.

En términos generales, se evidenció que la empresa cuenta con una base ágil fundamentada en Scrum, sin embargo, su implementación no sigue una estructura formal ni estandarizada. Los resultados de la encuesta indican que, si bien el 75 % de los desarrolladores reconoce que Scrum facilita la organización de tareas (Pregunta 1), en aspectos críticos como la identificación oportuna de problemas (Pregunta 4) y la mejora de la calidad del software (Pregunta 9), el 62,5 % manifiesta estar en desacuerdo con que la metodología actual cumpla dichas funciones. Asimismo, el 62,5 % no percibe mejoras en la rapidez ni eficiencia de entrega (Pregunta 7), y el 50 % mantiene una posición neutral respecto al impacto en la productividad del equipo (Pregunta 6).

Complementariamente, la entrevista al presidente ejecutivo confirmó que los principales problemas radican en: la ausencia de una guía metodológica formal, la falta de definición clara de roles, la limitada gestión visual del flujo de trabajo y las dificultades del equipo para adaptarse a cambios de requisitos durante el desarrollo.

El presente diagrama representa el flujo de actividades que sigue la empresa para la ejecución de proyectos de desarrollo de software, considerando un caso real con una duración de 10 días laborables. Se evidencia un proceso lineal que inicia con la recepción y análisis del requerimiento, seguido de la asignación directa de tareas y el desarrollo de funcionalidades, sin la aplicación formal de metodologías ágiles. Asimismo, se identifican puntos de retroalimentación tardía, especialmente en las fases de pruebas y validación del cliente, lo que puede generar retrabajo. La distribución de tiempos muestra una concentración significativa en la etapa de

desarrollo, con limitada planificación y control en fases previas, evidenciando oportunidades de mejora en la gestión del proceso.

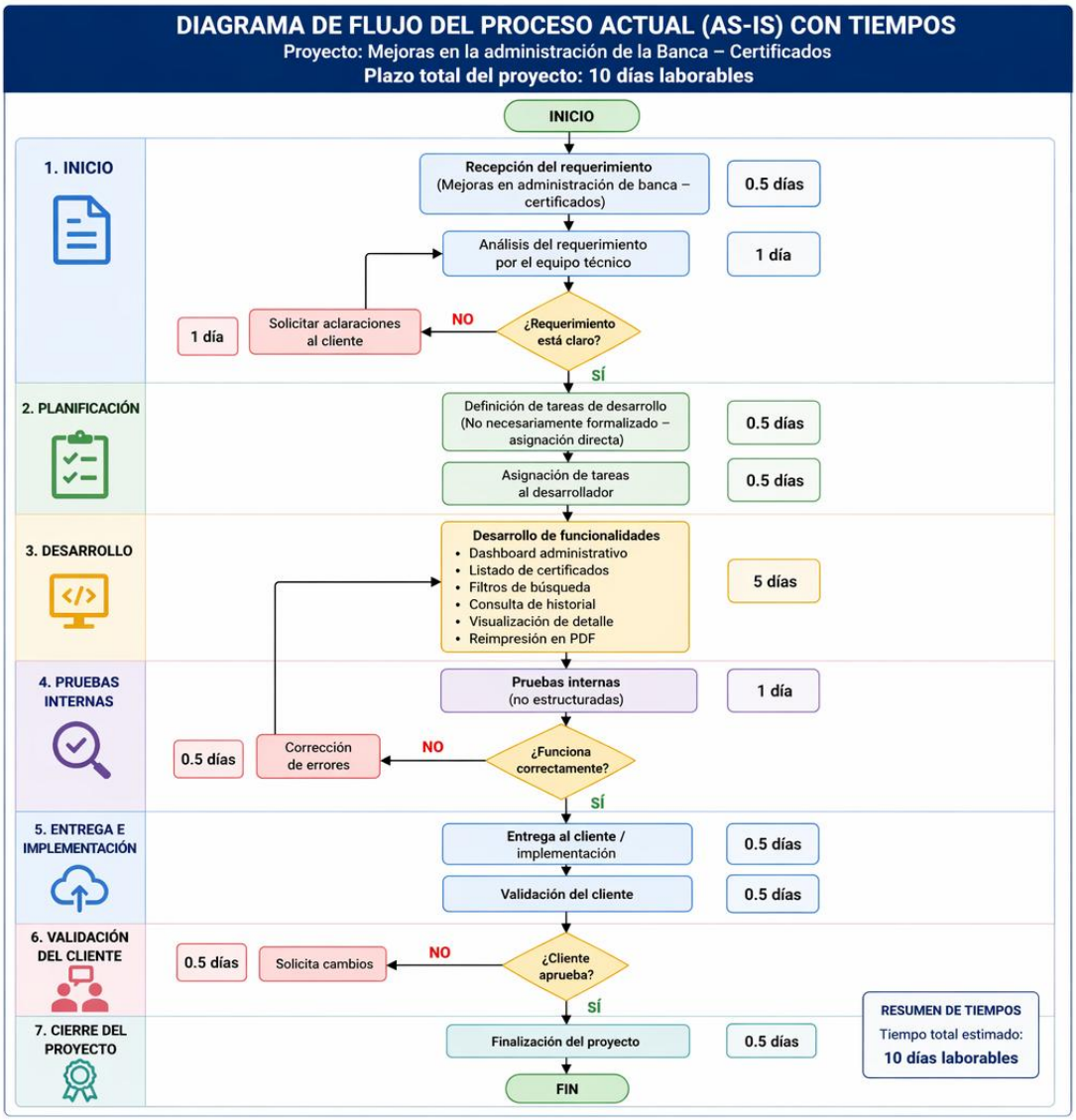


Figura 12. Diagrama de flujo del proceso actual de desarrollo de software con tiempos estimados en la empresa Black Label Technology Cía. Ltda.

Se elaboró un diagrama de Gantt para visualizar la distribución temporal del proyecto, evidenciando la concentración de actividades en fases de desarrollo.

Se observa que la mayor parte del tiempo se concentra en el desarrollo de funcionalidades, mientras que las actividades de análisis, pruebas y validación cuentan con menor duración. Esta distribución evidencia una planificación tradicional, con escasa iteración y validación continua, lo que puede afectar la calidad del producto

final y la capacidad de respuesta ante cambios en los requerimientos. Este comportamiento refleja la ausencia de prácticas ágiles estructuradas dentro del proceso actual de la empresa.

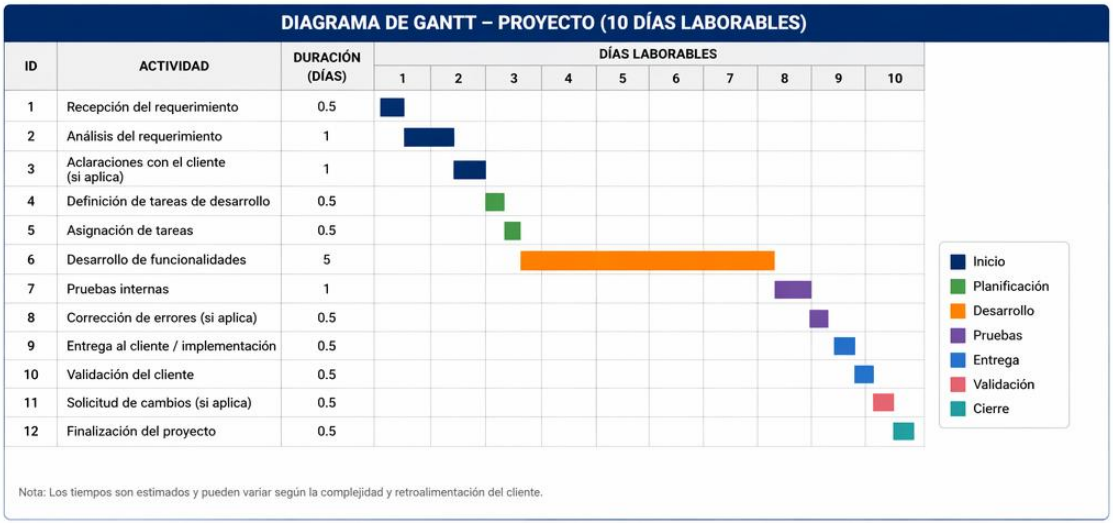


Figura 13. Diagrama de Gantt del proyecto de desarrollo de mejoras en la administración de la banca (10 días laborables).

A partir de las figuras presentadas, se identifican debilidades en la gestión del tiempo, control de actividades y validación continua, lo cual justifica la necesidad de implementar una metodología ágil.

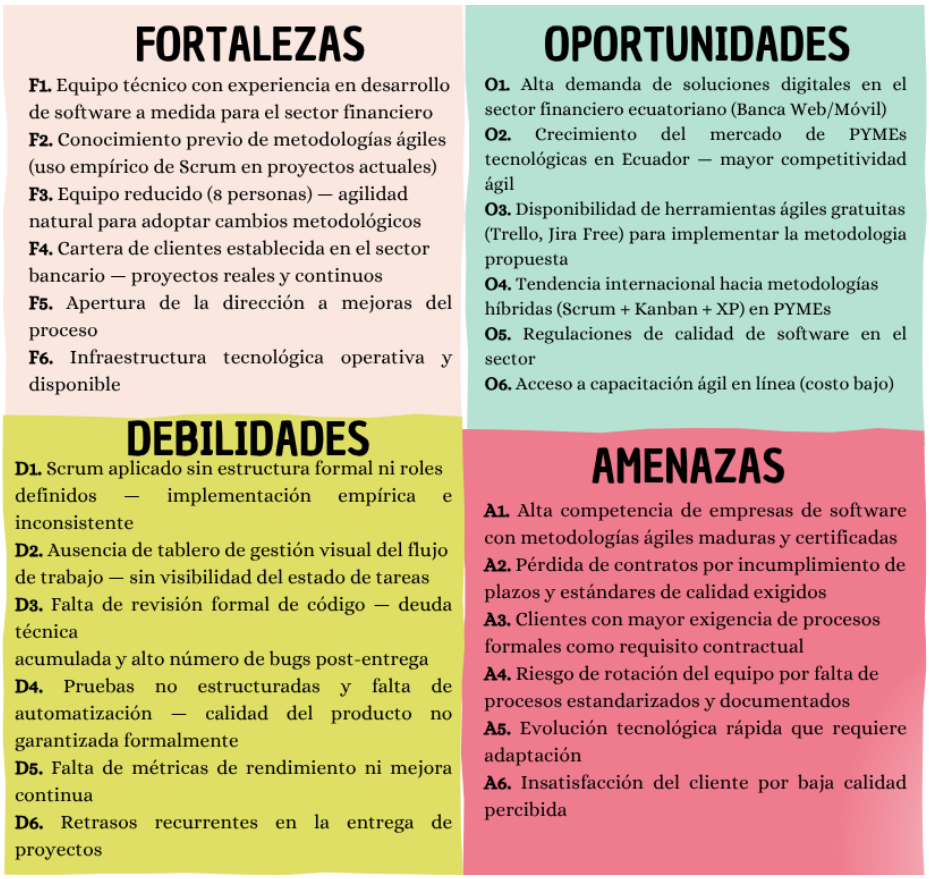
3.1.1 Análisis FODA del proceso de desarrollo de software

Con el propósito de completar el cumplimiento del primer objetivo específico de la investigación se elaboró un análisis FODA (Fortalezas, Oportunidades, Debilidades y Amenazas) aplicado al proceso de desarrollo de software de Blacklabeltech Black Label Technology Cía. Ltda. Este análisis complementa el diagnóstico del proceso AS-IS realizado mediante el diagrama de flujo y el diagrama de Gantt presentados en las secciones anteriores, incorporando una perspectiva estratégica que evalúa tanto los factores internos de la empresa como los factores externos del entorno en el que opera.

El análisis FODA fue construido a partir de la triangulación de tres fuentes de información: los resultados de la encuesta aplicada al equipo de desarrollo (Capítulo II, Alfa de Cronbach = 0.903), la entrevista estructurada al presidente ejecutivo y el análisis documental del proceso AS-IS. Esta combinación de fuentes garantiza que

cada elemento del FODA tenga respaldo empírico y no sea producto de suposiciones teóricas.

Tabla 21. Análisis FODA — Blacklabeltech Black Label Technology Cía. Ltda.



Fuente: Elaboración propia a partir del diagnóstico del proceso AS-IS y encuesta aplicada al equipo (2026).

a. Fortalezas – factores internos positivos

Las fortalezas representan las capacidades internas de la empresa para enfrentar los retos del desarrollo de software. Destaca la experiencia del equipo en soluciones para el sector financiero (F1), lo que asegura una base técnica sólida. Además, existe conocimiento previo de metodologías ágiles como Scrum (F2), aunque sin una aplicación formal. El tamaño reducido del equipo (F3) facilita la implementación de cambios. Finalmente, aspectos como una cartera de clientes estable (F4), la disposición de la dirección a mejorar (F5) y la infraestructura tecnológica disponible (F6) refuerzan la viabilidad de la propuesta.

b. Oportunidades – factores externos positivos

Las oportunidades reflejan un entorno favorable para adoptar una metodología ágil estructurada. Existe una alta demanda de soluciones digitales en el sector financiero (O1), lo que abre posibilidades de crecimiento si se optimizan los procesos. Además, el acceso a herramientas ágiles gratuitas como Trello o Jira (O3) reduce costos de implementación. Por último, la tendencia global hacia metodologías híbridas en PYMEs (O4) respalda la propuesta planteada.

c. Debilidades – factores internos negativos

Las debilidades evidencian las principales limitaciones internas. La aplicación informal de Scrum (D1) muestra falta de estandarización en los procesos. También se identifican problemas como la ausencia de revisiones formales de código (D3), pruebas poco estructuradas (D4) y falta de métricas (D5), lo que dificulta evaluar el rendimiento y cumplir con exigencias de clientes.

d. Amenazas – factores externos negativos

Las amenazas representan riesgos del entorno si no se corrigen las debilidades. La competencia con empresas que aplican metodologías ágiles consolidadas (A1) es alta. Existe riesgo de pérdida de contratos por incumplimiento de plazos (A2), evidenciado en la sobrecarga de actividades en etapas críticas. Además, la falta de procesos estandarizados puede provocar rotación del personal y pérdida de conocimiento organizacional (A4).

En síntesis, el análisis FODA muestra que la empresa Blacklabeltech cuenta con una posición favorable, ya que sus fortalezas y oportunidades superan sus debilidades y amenazas, siempre que se corrijan de manera estructurada los problemas detectados en el proceso de desarrollo. Las debilidades identificadas coinciden con las limitaciones evidenciadas en el análisis AS-IS (diagrama de flujo y Gantt) y cada una tiene una propuesta de mejora dentro de la metodología ágil a diseñar. De esta forma, se cumple el primer objetivo específico, al identificar las prácticas actuales, sus limitaciones y las oportunidades que sustentan el diseño de la metodología ágil propuesta.

A partir de las debilidades y amenazas identificadas en el análisis FODA, se establecen las limitaciones que afectan actualmente el proceso de desarrollo de software de la empresa, junto con las acciones correctivas necesarias para mitigarlas. Estas acciones están orientadas a resolver los problemas internos detectados y a reducir los riesgos externos que podrían comprometer la competitividad y sostenibilidad de la empresa.

Tabla 22. Limitaciones identificadas y acciones correctivas para su mitigación.

Limitaciones	Acciones correctivas
D1. Scrum aplicado sin estructura formal ni roles definidos — implementación empírica e inconsistente	Formalizar la aplicación de Scrum mediante la definición clara de roles (Product Owner, Scrum Master y Equipo de Desarrollo), eventos y artefactos adaptados a la realidad del equipo, estableciendo una guía metodológica interna que estandarice el proceso.
D2. Ausencia de tablero de gestión visual del flujo de trabajo — sin visibilidad del estado de tareas	Implementar un tablero Kanban con columnas definidas (Backlog / En desarrollo / En revisión / Hecho) utilizando herramientas gratuitas como Trello o Jira Free, permitiendo al equipo visualizar el estado de cada tarea en tiempo real.
D3. Falta de revisión formal de código — deuda técnica acumulada y alto número de bugs post-entrega	Incorporar la práctica de revisión de código en pareja antes de cada merge, junto con estándares de codificación internos que aseguren la consistencia y mantenibilidad del software desarrollado.
D4. Pruebas no estructuradas y falta de automatización — calidad del producto no garantizada formalmente	Establecer una Definición de Hecho (DoD) que incluya criterios obligatorios de pruebas antes de considerar una tarea como terminada, asegurando un control de calidad en cada entregable.
D5. Falta de métricas de rendimiento ni mejora continua	Definir métricas de seguimiento como velocidad del sprint, tiempo de ciclo y tasa de defectos, e incorporar reuniones de retrospectiva estructuradas al final de cada sprint para evaluar el desempeño y proponer mejoras.
D6. Retrasos recurrentes en la entrega de proyectos	Fortalecer la planificación mediante Sprint Planning con estimación en puntos de historia y Daily Stand-up diarias para la detección temprana de bloqueos, mejorando el cumplimiento de plazos acordados.
A1. Alta competencia de empresas de software con metodologías ágiles maduras y certificadas	Estructurar y documentar formalmente el proceso ágil de la empresa, de modo que pueda ser presentado como evidencia de madurez metodológica ante clientes y en procesos de licitación.
A2. Pérdida de contratos por incumplimiento de plazos y estándares de calidad exigidos	Mejorar la predictibilidad de las entregas mediante ciclos iterativos con entregables verificables, y reforzar los estándares de calidad con prácticas de revisión de código y pruebas controladas.
A3. Clientes con mayor exigencia de procesos formales como requisito contractual	Generar documentación del proceso de desarrollo (backlog, actas de Sprint Review, métricas de sprint) que pueda ser entregada al cliente como parte de los acuerdos contractuales.
A4. Riesgo de rotación del equipo por falta de procesos estandarizados y documentados	Documentar los procesos, roles y estándares de codificación para reducir la dependencia del conocimiento individual y facilitar la incorporación de nuevos miembros al equipo.
A5. Evolución tecnológica rápida que requiere adaptación	Fomentar la capacitación continua del equipo en prácticas ágiles y nuevas tecnologías, aprovechando recursos de formación en línea de bajo costo.
A6. Insatisfacción del cliente por baja calidad percibida	Incorporar la Sprint Review como instancia formal de validación con el cliente al final de cada sprint, permitiendo retroalimentación temprana y ajustes antes de la entrega final.

De igual manera, a partir de las oportunidades identificadas en el análisis FODA, se establecen las acciones estratégicas que la empresa puede implementar para aprovechar cada una de ellas en beneficio de su proceso de desarrollo de software. Estas acciones están orientadas a fortalecer la posición competitiva de la empresa, mejorar la calidad de sus productos y optimizar la gestión de proyectos mediante la adopción de prácticas ágiles adaptadas a su contexto.

Tabla 23. Oportunidades identificadas y acciones estratégicas para su aprovechamiento.

Oportunidad	Acciones para aprovecharla
O1. Alta demanda de soluciones digitales en el sector financiero ecuatoriano (Banca Web/Móvil)	Orientar la planificación de sprints hacia proyectos del sector financiero, priorizando en el backlog historias de usuario relacionadas con plataformas bancarias web y móvil para captar nuevos clientes en este nicho.
O2. Crecimiento del mercado de PYMEs tecnológicas en Ecuador — mayor competitividad ágil	Aprovechar la adopción de una metodología ágil estructurada como diferenciador competitivo frente a otras PYMEs, demostrando a los clientes potenciales un proceso de desarrollo con entregas predecibles y calidad controlada.
O3. Disponibilidad de herramientas ágiles gratuitas (Trello, Jira Free) para implementar la metodología	Adoptar Trello o Jira Free como soporte digital del tablero Kanban, configurando columnas de flujo de trabajo (Backlog / En desarrollo / En revisión / Hecho) y límites de tareas en progreso sin incurrir en costos adicionales.
O4. Tendencia internacional hacia metodologías híbridas (Scrum + Kanban + XP) en PYMEs	Alinear las prácticas ágiles de la empresa con las tendencias globales de metodologías híbridas, facilitando la colaboración con clientes internacionales y la participación en proyectos que exigen marcos ágiles documentados.
O5. Regulaciones de calidad de software en el sector	Incorporar prácticas como la Definición de Hecho (DoD) y revisiones de código en pareja como evidencia de cumplimiento de estándares de calidad, facilitando la obtención de certificaciones y el cumplimiento de requisitos contractuales.
O6. Acceso a capacitación ágil en línea (costo bajo)	Diseñar un plan de formación continua para el equipo aprovechando plataformas en línea de bajo costo, enfocado en prácticas de Scrum, Kanban y XP, fortaleciendo las competencias del equipo sin afectar el presupuesto.

En la Tabla 24 se presenta un resumen del diagnóstico comparativo entre la situación actual y las oportunidades de mejora identificadas, las cuales fundamentan el diseño de la metodología propuesta.

Tabla 24. Diagnóstico comparativo de la situación actual vs. oportunidades de mejora.

Dimensión	Situación actual (Scrum sin estructura)	Oportunidad de mejora
Planificación	Sprints mal definidos, sin criterios claros de aceptación	Definir plantillas de Sprint Planning con criterios de aceptación

Dimensión	Situación actual (Scrum sin estructura)	Oportunidad de mejora
Gestión visual	No existe tablero de flujo de trabajo visible para todo el equipo	Implementar tablero Kanban con columnas: Backlog / En progreso / Revisión / Hecho
Calidad del código	No se realizan revisiones formales de código ni pruebas guiadas	Incorporar revisión de código en pareja y definición de estándares (XP)
Identificación de problemas	El 62,5 % de desarrolladores indica que Scrum no detecta problemas a tiempo	Establecer reuniones de retrospectiva estructuradas y métricas de seguimiento
Roles y responsabilidades	Roles no claramente definidos; el equipo actúa de forma empírica	Definir formalmente roles adaptados: Product Owner, Scrum Master y Equipo de desarrollo
Herramienta de gestión	No se utiliza ninguna herramienta de gestión visual estandarizada	Adoptar Trello o Jira como herramienta de gestión del tablero Kanban y backlog

El diagnóstico refleja que la empresa dispone de las condiciones humanas y técnicas necesarias para adoptar una metodología más estructurada. El equipo de ocho personas; cinco desarrolladores, un analista de procesos, un gerente y el presidente ejecutivo representa una escala idónea para la implementación de metodologías ágiles adaptadas al contexto de una PYME tecnológica, tal como lo señalan Schmidt & López (2021) y Rivera (2022).

3.2 Análisis de metodologías ágiles y adaptación a la realidad de la empresa

Se plantea analizar las metodologías ágiles conforme a las características y necesidades particulares del equipo de desarrollo y los proyectos de Blacklabeltech Black Label Technology Cía. Ltda., adaptando los roles a la realidad de la empresa. Para dar cumplimiento a este objetivo, la presente sección desarrolla cuatro componentes articulados: (1) el análisis comparativo de las tres metodologías ágiles candidatas, (2) el diagnóstico de las características reales del equipo y los proyectos, (3) los criterios de selección de prácticas y (4) el diseño de roles formalmente adaptados a la plantilla actual de la empresa.

El análisis parte de los hallazgos del OE1: las seis debilidades detectadas en el proceso AS-IS y el FODA definen con precisión qué problemas debe resolver la metodología seleccionada. No se trata, por tanto, de una revisión teórica abstracta de metodologías, sino de un análisis orientado a resolver problemas específicos, documentados y medibles de la empresa.

3.2.1 Análisis comparativo de metodologías ágiles

Se analizaron las tres metodologías ágiles más relevantes para el contexto de una PYME tecnológica de desarrollo de software: Scrum, Kanban y Extreme Programming (XP). La selección de estas tres metodologías responde a los antecedentes investigativos del Capítulo I, que identifican que Scrum es la metodología ágil predominante en la industria ecuatoriana, Kanban es su complemento visual más frecuente y XP aporta las prácticas técnicas de calidad de código que ninguna de las otras dos provee de forma explícita (Schmidt & López, 2021; Rivera et al., 2022; Dikert et al., 2020).

El análisis se realiza mediante 10 criterios de evaluación seleccionados en función de los problemas diagnosticados en la empresa: enfoque principal, ciclos de trabajo, roles definidos, gestión visual, reuniones, calidad del código, adaptación al cambio, tamaño de equipo ideal, métricas de rendimiento y aplicabilidad específica en BLT. La Tabla 25 presenta el análisis comparativo completo.

Tabla 25. Análisis comparativo de metodologías ágiles — Scrum, Kanban y Extreme Programming.

Criterio de análisis	Scrum	Kanban	Extreme Programming (XP)
Enfoque principal	Gestión iterativa de proyectos mediante sprints con roles, eventos y artefactos definidos	Visualización y control del flujo de trabajo continuo con límite de tareas en progreso (WIP)	Calidad del software mediante prácticas técnicas de programación colaborativa y pruebas constantes
Ciclos de trabajo	Sprints fijos de 1 a 4 semanas con entregable funcional al finalizar cada ciclo	Flujo continuo sin ciclos fijos, las tareas avanzan según capacidad del equipo	Iteraciones muy cortas (1 a 2 semanas) con integración y pruebas continuas del código
Roles definidos	Product Owner, Scrum Master, Equipo de Desarrollo, roles con responsabilidades explícitas	No define roles formales, cualquier miembro puede mover tareas en el tablero	Coach XP, Cliente en sitio, Programadores — roles orientados a la calidad técnica
Gestión visual	Sprint Backlog y Product Backlog documentados — tablero no obligatorio pero recomendado	Tablero Kanban como artefacto central — columnas de estado con límite WIP por columna	Mínima gestión visual — enfocado en prácticas de código, no en tableros de seguimiento
Reuniones / eventos	Sprint Planning, Daily Scrum, Sprint Review,	Sin reuniones formales obligatorias — se	Reuniones breves diarias (standup) y sesiones de

Criterio de análisis	Scrum	Kanban	Extreme Programming (XP)
	Retrospectiva — eventos formales y obligatorios	realizan según necesidad del equipo	programación en pareja (pair programming)
Calidad del código	No especifica prácticas técnicas de código — la calidad depende del equipo	No especifica prácticas técnicas — se centra en el flujo, no en el código	Revisión de código en pareja, TDD (desarrollo guiado por pruebas), integración continua
Adaptación al cambio	Cambios entre sprints — dentro del sprint el alcance es estable para proteger al equipo	Alta adaptabilidad — las tareas pueden modificarse en cualquier momento del flujo	Muy alta — el cliente puede cambiar requisitos entre iteraciones sin coste adicional
Tamaño de equipo ideal	3 a 9 personas — óptimo para equipos pequeños y medianos como BLT (8 personas)	Sin restricción de tamaño — funciona desde 1 persona hasta equipos grandes	2 a 12 personas — requiere alta comunicación y disciplina técnica del equipo
Métricas de rendimiento	Velocidad del sprint, burndown chart, % de cumplimiento por sprint	Tiempo de ciclo (lead time), throughput, porcentaje de eficiencia del flujo	Cobertura de pruebas, tasa de defectos, tiempo de integración
Aplicabilidad en BLT	ALTA — estructura clara para planificar y hacer seguimiento. Soluciona D1 (roles), D6 (retrasos)	ALTA — tablero visual soluciona D2 (sin visibilidad). Complementa Scrum sin reemplazarlo	MEDIA-ALTA — revisión de código y TDD solucionan D3 (bugs) y D4 (pruebas). Requiere disciplina
Limitación principal	No garantiza calidad técnica del código ni visibilidad visual del flujo de trabajo	Sin estructura de planificación ni roles, difícil gestionar proyectos con plazos fijos	Alta curva de aprendizaje técnico requiere experiencia en TDD y pair programming

El análisis comparativo revela que ninguna de las tres metodologías, aplicada de forma aislada, resuelve de manera completa el conjunto de debilidades identificadas en el diagnóstico del OE1. Scrum provee la estructura de planificación y seguimiento que elimina la informalidad actual, pero no garantiza la calidad técnica del código ni la visibilidad visual del flujo de trabajo. Kanban resuelve la ausencia de visibilidad, pero carece de estructura para gestionar proyectos con plazos fijos. XP aporta las prácticas de calidad de código que directamente reducen los bugs, pero su curva de aprendizaje técnico es la más alta de las tres. Esta evidencia justifica la necesidad de una metodología híbrida que integre los elementos más efectivos de cada framework, tal como lo recomiendan Schmidt & López (2021) y Rivera et al. (2022) para PYMEs tecnológicas con perfiles similares al de Blacklabeltech.

3.2.2 Diagnóstico de las características del equipo y los proyectos

La selección y adaptación de prácticas metodológicas no puede realizarse de forma genérica, sino que debe responder a las características concretas del equipo y los proyectos de la empresa. Para ello se elaboró un diagnóstico estructurado de ocho dimensiones: tamaño y composición del equipo, experiencia ágil previa, tipo de proyectos, comunicación con el cliente, herramientas disponibles, calidad actual del código y gestión de plazos. Cada dimensión se analiza con el dato real de BLT y su implicación metodológica directa para el diseño de la metodología propuesta.

Tabla 26. Diagnóstico del equipo y los proyectos de Blacklabeltech y su implicación metodológica.

Característica del equipo / proyecto	Dato real de BLT	Implicación metodológica propuesta
Tamaño del equipo	8 personas	Scrum es óptimo para equipos de 3–9 personas. BLT puede aplicarlo sin reducir ni ampliar la plantilla actual. Los 3 roles de Scrum se cubren con el personal existente
Composición del equipo	5 dev., 1 analista, 1 gerente, 1 presidente ejecutivo	El presidente ejecutivo asume el rol de Product Owner (visión del producto y cliente). El analista de procesos puede actuar como Scrum Master en los primeros sprints. Los 5 desarrolladores conforman el Equipo de Desarrollo
Experiencia ágil previa	Scrum empírico — sin estructura formal	No se parte de cero: el equipo ya conoce los conceptos básicos. Se requiere formalización de roles, eventos y artefactos, no introducción desde cero. Curva de aprendizaje reducida
Tipo de proyectos	Software a medida para sector financiero — plazos fijos de 10 a 30 días	Los proyectos de plazo fijo requieren planificación por sprints (Scrum), no flujo continuo (Kanban puro). Los sprints de 2 semanas se alinean con proyectos de 10 días al permitir entregas parciales verificables
Comunicación con el cliente	Informal (correo / WhatsApp) sin protocolo	La Sprint Review formaliza la comunicación cliente–equipo con una demo estructurada y acta de aceptación, reemplazando la comunicación informal actual
Herramientas disponibles	Sin herramienta de gestión formal	Trello (gratuito) cubre el tablero Kanban y el Sprint Backlog. No se requiere inversión. La migración a Jira puede realizarse en fases posteriores cuando el equipo madure la metodología
Calidad actual del código	Sin revisión formal — 8.4 bugs/entrega promedio	La práctica de revisión en pareja (XP) y la Definición de Hecho (DoD) con pruebas unitarias obligatorias responden directamente a este problema. Meta: reducir bugs en ≥ 50 %
Gestión de plazos	52 % de cumplimiento en tiempo acordado	El Sprint Planning con estimación en puntos de historia y la Daily Stand-up diaria para detección temprana de bloqueos son las prácticas de Scrum que directamente elevan el % de cumplimiento

El diagnóstico revela dos hallazgos especialmente relevantes para la selección metodológica. Primero, el tamaño del equipo (8 personas) es óptimo para Scrum, que funciona mejor entre 3 y 9 integrantes, lo que elimina la necesidad de frameworks de escalado como SAFe o LeSS que serían innecesariamente complejos para BLT. Segundo, la experiencia previa con Scrum, aunque informal, significa que el equipo ya tiene familiaridad con los conceptos básicos, lo que reduce la curva de aprendizaje estimada a 8 horas de capacitación inicial (RNF-05), frente a las 20 - 40 horas que requeriría un equipo sin ninguna exposición ágil previa.

3.2.3 Criterios de selección de prácticas metodológicas

Con base en el análisis comparativo de la sección 3.2.1 y el diagnóstico del equipo de la sección 3.2.2, se definieron cinco criterios ponderados para determinar qué prácticas de cada metodología se incorporan a la metodología propuesta. La ponderación asigna mayor peso a los criterios que responden directamente a las debilidades diagnosticadas y a la adecuación al tamaño del equipo, ya que estos son los factores que determinan la efectividad real de la implementación en el contexto específico de la empresa.

La asignación de pesos a cada criterio de selección se realizó mediante una escala de priorización de tres niveles, definida en función del grado de impacto que cada criterio tiene sobre la viabilidad y efectividad de la metodología en el contexto de la empresa. Los niveles establecidos fueron los siguientes:

Prioridad alta (30 %): Se asigna a criterios que tienen un impacto directo y determinante en la resolución del problema central de la investigación.

En este caso, “Responde a las 6 debilidades diagnosticadas" recibió este peso, ya que el objetivo principal de la selección metodológica es abordar las problemáticas identificadas en el diagnóstico. Una metodología que no resuelva estas debilidades no cumpliría con el propósito de la investigación.

Prioridad media (20 %): Se asigna a criterios que condicionan la factibilidad de implementación de la metodología dentro de la empresa.

Bajo este nivel se encuentran los criterios "Adecuada para equipo de 8 personas" e "Implementable sin costo adicional", dado que ambos definen si la metodología puede ser adoptada de manera práctica y sostenible considerando las limitaciones de tamaño y presupuesto de la empresa.

Prioridad baja (15 %): Se asigna a criterios que, si bien son importantes, representan factores complementarios que facilitan la adopción, pero no la condicionan de manera crítica.

Los criterios "Curva de aprendizaje manejable (≤ 8 h)" y "Genera métricas cuantificables" se ubicaron en este nivel, ya que contribuyen a una mejor transición y seguimiento del proceso, pero su ausencia no impediría la implementación de la metodología.

La suma total de los pesos equivale al 100 %, lo que garantiza una distribución proporcional y coherente con las prioridades identificadas en el diagnóstico de la empresa.

Tabla 27. Criterios de selección ponderados de prácticas metodológicas para la metodología propuesta.

Criterio de selección	Peso	Scrum	Kanban	XP
Responde a las 6 debilidades diagnosticadas	30 %	Alto (D1, D5, D6)	Alto (D2)	Alto (D3, D4)
Adecuada para equipo de 8 personas	20 %	Óptimo (3–9 p.)	Flexible	Adecuado (2–12 p.)
Implementable sin costo adicional	20 %	Sí — Trello/Jira Free	Sí — mismo tablero	Sí — prácticas de equipo
Curva de aprendizaje manejable (≤ 8 h)	15 %	Media (familiar)	Baja (visual, intuitivo)	Alta (TDD, pair programming)
Genera métricas cuantificables	15 %	Sí (velocidad, %)	Sí (tiempo ciclo)	Sí (cobertura, bugs)
SELECCIÓN para BLT-Agile	100 %	COMPLETO estructura base de BLT-Agile	PARCIAL tablero + WIP + tiempo de ciclo	PARCIAL revisión en pareja + DoD + estándares de código

La evaluación ponderada indica que Scrum debe adoptarse como base principal de la metodología a diseñar, al obtener la mayor valoración en los criterios clave. Kanban se integra de forma parcial, aportando el uso de tableros visuales, límites WIP y métricas de tiempo de ciclo. XP también se incorpora parcialmente, incluyendo prácticas como revisión de código en pareja, Definición de Hecho (DoD) y estándares

de codificación. En cambio, prácticas más complejas de XP, como TDD o integración continua, no se incluyen inicialmente debido al tiempo de implementación y al posible impacto en la adopción por parte del equipo.

3.2.4 Adaptación de roles a la realidad de la empresa

Uno de los hallazgos críticos del diagnóstico del OE1 fue la ausencia de roles formalmente definidos en el proceso de desarrollo: la distribución de tareas se realizaba verbalmente, sin responsabilidades explícitas ni estructura de facilitación del proceso. La adaptación de roles es, por tanto, uno de los componentes de mayor impacto de la metodología propuesta. A diferencia de una implementación estándar de Scrum que podría requerir la contratación de un Scrum Master dedicado, la metodología a diseñar adapta los tres roles del framework a las personas que ya forman parte del equipo de la empresa, sin generar costos adicionales de personal.

La adaptación considera tres restricciones específicas de Blacklabeltech: (1) el equipo tiene solo 8 personas, lo que impide la dedicación exclusiva a roles de gestión; (2) el analista de procesos tiene el perfil más adecuado para la facilitación, pero no puede dedicar el 100 % de su tiempo al rol de Scrum Master; y (3) el rol de Product Owner requiere acceso directo al cliente y visión del negocio, características que solo el presidente ejecutivo posee dentro de la empresa. La Tabla 28 presenta el diseño formal de roles con su justificación.

Tabla 28. Propuesta de adaptación de roles ágiles a la realidad de Blacklabeltech.

Rol propuesto	Persona asignada	Responsabilidades adaptadas	Justificación de la adaptación
Product Owner	Presidente ejecutivo	Gestionar y priorizar el Product Backlog. Definir criterios de aceptación por historia de usuario. Representar al cliente en el Sprint Planning y validar el incremento en el Sprint Review	El presidente ejecutivo tiene visión directa del cliente y del negocio, lo que le permite priorizar funcionalidades según valor real. En BLT no existe un "Product Manager" separado, por lo que este rol es absorbido por quien más conoce las necesidades del cliente
Scrum Master	Analista de procesos (rotativo cada 3 sprints entre	Facilitar Sprint Planning, Daily Stand-up, Sprint Review y Retrospectiva. Registrar impedimentos y asegurar su	El analista de procesos tiene el perfil metodológico para facilitar el proceso. La rotación entre

Rol propuesto	Persona asignada	Responsabilidades adaptadas	Justificación de la adaptación
	desarrolladores senior)	resolución en < 24 h. Actualizar el tablero Kanban diariamente. Medir y reportar las 5 métricas de sprint	desarrolladores senior a partir del cuarto sprint distribuye el conocimiento de la facilitación, evitando dependencia de una sola persona y fortaleciendo la cultura ágil en todo el equipo
Equipo de Desarrollo	5 desarrolladores de software	Desarrollar, probar e integrar el incremento del sprint. Aplicar revisión de código en pareja antes de cada merge. Cumplir la Definición de Hecho (DoD) en cada tarea. Participar activamente en la estimación (Planning Poker) y la Retrospectiva	Los 5 desarrolladores constituyen el núcleo del equipo y no requieren cambio de cargo ni contratación adicional. La incorporación de la revisión en pareja y la DoD son prácticas nuevas que se integran al trabajo habitual sin aumentar la carga administrativa. El límite WIP de 2 tareas simultáneas por persona es el único cambio operativo significativo en su flujo de trabajo diario
Observación: Gerente	Rol de apoyo externo al proceso ágil	No tiene un rol formal en BLT-Agile, pero participa en la toma de decisiones estratégicas de recursos, aprobación de herramientas y seguimiento de indicadores del proceso	En una PYME de 8 personas el gerente no puede dedicarse exclusivamente a un rol ágil. Su participación se canaliza a través de las métricas del sprint y las reuniones de Review, donde puede ver los resultados sin interferir en el proceso de desarrollo diario

Una decisión de diseño relevante es la rotación del rol de Scrum Master entre los desarrolladores senior a partir del cuarto sprint. Esta decisión responde a la recomendación de Schwaber & Sutherland (2020) de que la facilitación del proceso no debe centralizarse en una sola persona, y tiene tres efectos positivos para la empresa: distribuye el conocimiento metodológico en el equipo, reduce la dependencia de la continuidad de una persona específica en el rol y fomenta una cultura ágil más profunda al hacer que múltiples miembros del equipo comprendan el proceso desde la perspectiva de facilitador.

La adaptación del rol de Product Owner al presidente ejecutivo es coherente con el contexto de una PYME donde la misma persona que tiene la visión del cliente también toma las decisiones comerciales. Sin embargo, para que este rol sea efectivo, el

presidente ejecutivo debe comprometerse a asistir al Sprint Planning y al Sprint Review de cada sprint, dos eventos de 3 horas en total por sprint, lo que representa una inversión de tiempo razonable y proporcional al impacto que tiene en la calidad de la planificación y en la reducción de retrabajos post-entrega.

3.3 Propuesta de metodología ágil a medida para Blacklabeltech Black Label Technology Cía. Ltda.

A partir del diagnóstico previo y del análisis teórico desarrollado en el Capítulo I, se propone una metodología ágil híbrida a medida, denominada BLT-Agile, que integra las prácticas más relevantes de Scrum, Kanban y Extreme Programming (XP), adaptadas específicamente a las características, dimensiones y necesidades de la empresa. Esta metodología no constituye una implementación rígida de ningún framework en particular, sino una combinación personalizada de sus elementos más efectivos, orientada a resolver los problemas concretos identificados en el diagnóstico.

La denominación BLT-Agile responde a las iniciales de Black Label Technology y al enfoque ágil que la sustenta. Su diseño se fundamenta en cuatro principios rectores: (1) iteraciones cortas con entregables verificables, (2) visibilidad continua del flujo de trabajo, (3) calidad incorporada desde el proceso de codificación, y (4) mejora continua a través de retroalimentación estructurada.

3.3.1 Componentes de la metodología BLT-Agile

La metodología BLT-Agile integra elementos de tres marcos ágiles con un propósito específico para cada uno dentro del contexto de la empresa. En la Tabla 29 se detallan los componentes seleccionados, su origen metodológico y su propósito en el contexto de Black Label Technology.

Tabla 29. Componentes de la metodología BLT-Agile.

Componente	Origen	Práctica adoptada	Propósito en BLT
Sprint (2 semanas)	Scrum	Ciclos de 2 semanas con entregable funcional	Entregas frecuentes y controladas
Product Backlog	Scrum	Lista priorizada de historias de usuario	Visibilidad de todo el trabajo pendiente
Daily Stand-up (15 min)	Scrum	Reunión diaria: ¿Qué hice? ¿Qué haré? ¿Hay bloqueos?	Comunicación y detección temprana de problemas

Componente	Origen	Práctica adoptada	Propósito en BLT
Sprint Review y Retrospectiva	Scrum	Revisión del incremento y análisis de mejoras al final de cada sprint	Mejora continua del proceso
Tablero Kanban	Kanban	Columnas: Backlog / En desarrollo / En revisión / Hecho	Visibilidad del flujo y detección de cuellos de botella
Límite de tareas en progreso (WIP)	Kanban	Máximo 2 tareas activas por desarrollador	Reducir multitarea y mejorar el foco
Revisión de código en pareja	XP	Revisión cruzada de código antes de cada merge	Mejorar calidad y reducir bugs
Estándares de codificación	XP	Guía interna de convenciones y buenas prácticas de código	Consistencia y mantenibilidad del software
Definición de Hecho (DoD)	Híbrido	Criterios que debe cumplir una tarea para considerarse terminada	Calidad controlada en cada entregable

La selección de estos componentes responde directamente a los problemas diagnosticados: los elementos de Scrum abordan la falta de estructura en la planificación y seguimiento; Kanban resuelve la ausencia de visibilidad del flujo de trabajo y los cuellos de botella; y las prácticas de XP atacan la baja calidad del software reportada por el equipo.

3.3.2 Trazabilidad entre debilidades diagnosticadas y soluciones de BLT-Agile

Para garantizar que la metodología propuesta responda de manera directa a los problemas identificados en el diagnóstico del OE1, se elaboró una matriz de trazabilidad que vincula cada debilidad con los componentes específicos de BLT-Agile que la resuelven, junto con la métrica que permite verificar su efectividad. Esta trazabilidad asegura que ningún componente de la metodología sea arbitrario, sino que cada práctica incluida tiene una justificación basada en evidencia diagnóstica.

Tabla 30. Matriz de trazabilidad: debilidades diagnosticadas, componentes de BLT-Agile y métricas de verificación.

Debilidad diagnosticada (OE1)	Componente de BLT-Agile que la resuelve	Métrica de verificación
D1. Scrum sin estructura formal ni roles definidos	Sprints de 2 semanas + roles formales (PO, SM, Equipo) + Sprint Planning + Sprint Review	% de eventos Scrum realizados por sprint (meta: 100 %)
D2. Ausencia de tablero de gestión visual	Tablero Kanban (Backlog / En desarrollo / En revisión / Hecho) + límite WIP de 2 tareas por desarrollador	Tiempo de ciclo promedio por tarea (meta: reducción ≥ 20 %)

Debilidad diagnosticada (OE1)	Componente de BLT-Agile que la resuelve	Métrica de verificación
D3. Falta de revisión formal de código — deuda técnica	Revisión de código en pareja + estándares de codificación internos	Tasa de bugs post-entrega (meta: reducción ≥ 50 %, de 8.4 a ≤ 4.2 bugs/entrega)
D4. Pruebas no estructuradas — calidad no garantizada	Definición de Hecho (DoD) con criterios de pruebas obligatorios	% de tareas que cumplen la DoD antes del merge (meta: 100 %)
D5. Falta de métricas de rendimiento	Velocidad del sprint + tiempo de ciclo + tasa de defectos + % cumplimiento de sprint + satisfacción del cliente	Número de métricas registradas por sprint (meta: 5 métricas en cada sprint)
D6. Retrasos recurrentes en entregas	Sprint Planning con estimación en puntos de historia + Daily Stand-up para detección de bloqueos	% de cumplimiento de entregas en tiempo (meta: ≥ 80 %, desde el 52 % actual)

La Tabla 30 presenta la matriz de trazabilidad entre las debilidades identificadas durante el diagnóstico inicial, los componentes de la metodología BLT-Agile propuestos para mitigarlas y las métricas utilizadas para verificar su efectividad. Esta relación permite evaluar de manera objetiva el impacto de la metodología en la mejora de los procesos de desarrollo de software.

3.3.3 Definición de roles adaptados

Uno de los problemas identificados en el diagnóstico fue la ausencia de roles claramente definidos dentro del equipo. BLT-Agile establece tres roles adaptados a la estructura de la empresa, considerando que se trata de un equipo pequeño de ocho personas. En la Tabla 31 se presenta la asignación de roles propuesta.

Tabla 31. Roles definidos en la metodología BLT-Agile.

Rol	Responsable	Funciones principales
Product Owner	Presidente Ejecutivo	Priorizar el Product Backlog, definir historias de usuario, validar entregables y representar las necesidades del cliente.
Scrum Master	Analista de procesos	Facilitar los eventos Scrum, eliminar impedimentos, asegurar el cumplimiento de la metodología y promover la mejora continua.
Equipo de Desarrollo	5 desarrolladores + Gerente (apoyo técnico)	Desarrollar las historias de usuario, realizar revisiones de código en pareja, actualizar el tablero Kanban y participar en los eventos del sprint.

Esta distribución permite aprovechar la estructura existente de la empresa sin requerir contrataciones adicionales, al tiempo que formaliza responsabilidades que antes se mantenían de manera implícita.

3.3.4 Fases del ciclo de desarrollo BLT-Agile

La metodología BLT-Agile organiza el trabajo en un ciclo iterativo de cinco fases que se repiten cada dos semanas (un sprint). Cada fase tiene actividades, responsables y artefactos específicos que garantizan el cumplimiento de los principios rectores de la metodología.

Tabla 32. Fases del ciclo de desarrollo BLT-Agile.

Fase	Nombre	Actividades	Responsable	Artefacto/Salida
1	Sprint Planning	Revisión del Product Backlog, selección de historias de usuario, estimación de esfuerzo y definición de criterios de aceptación.	Product Owner + Equipo	Sprint Backlog con criterios de aceptación definidos
2	Desarrollo iterativo	Codificación, actualización del tablero Kanban, límite WIP de 2 tareas por desarrollador, revisión de código en pareja antes de cada merge.	Equipo de Desarrollo	Incremento funcional del software
3	Daily Stand-up	Reunión diaria de 15 minutos: ¿Qué hice ayer? ¿Qué haré hoy? ¿Qué me bloquea?	Scrum Master + Equipo	Registro de impedimentos y avances
4	Sprint Review	Presentación del incremento funcional al Product Owner, validación de criterios de aceptación, retroalimentación del cliente.	Product Owner + Equipo	Incremento validado y retroalimentación documentada
5	Retrospectiva	Análisis de lo que funcionó, lo que no funcionó y acciones de mejora para el próximo sprint.	Scrum Master + Equipo	Plan de mejora continua

La Tabla 32 muestra las principales fases del ciclo de desarrollo BLT-Agile, especificando las actividades, responsables y resultados esperados en cada etapa, con el fin de asegurar un proceso de desarrollo organizado, colaborativo e iterativo.

3.3.5 Flujo de trabajo visual (Tablero Kanban)

Para resolver la ausencia de visibilidad del flujo de trabajo identificada en el diagnóstico, BLT-Agile incorpora un tablero Kanban con cuatro columnas que permiten al equipo visualizar el estado de cada tarea en tiempo real. En la Tabla 33 se describe la estructura del tablero propuesto.

Tabla 33. Estructura del tablero Kanban en BLT-Agile.

Backlog	En desarrollo	En revisión	Hecho
Historias de usuario seleccionadas para el sprint, priorizadas según valor de negocio.	Tareas en proceso de codificación. Límite WIP: máximo 2 tareas por desarrollador.	Tareas finalizadas en espera de revisión de código en pareja y validación de criterios de aceptación.	Tareas que cumplen la Definición de Hecho (DoD) y están listas para integración.

Se recomienda implementar este tablero mediante herramientas digitales como Trello o Jira, las cuales permiten la gestión colaborativa y el seguimiento en tiempo real del flujo de trabajo.

3.3.6 Definición de Hecho (Definition of Done – DoD)

Para garantizar la calidad controlada en cada entregable, BLT-Agile establece una Definición de Hecho que todo el equipo debe cumplir antes de mover una tarea a la columna "Hecho" del tablero Kanban. En la Tabla 34 se presentan los criterios definidos.

Tabla 34. Definición de Hecho (DoD) en BLT-Agile.

N°	Criterio	Verificación	Revisor/Responsable
1	El código cumple con los estándares de codificación internos.	Revisión de código en pareja aprobada.	Gerente de Desarrollo
2	Las pruebas funcionales han sido ejecutadas sin errores críticos.	Evidencia de pruebas documentada.	Desarrolladores
3	Los criterios de aceptación definidos en la historia de usuario se cumplen.	Validación del Product Owner.	Product Owner
4	La funcionalidad está integrada en la rama principal sin conflictos.	Merge exitoso en el repositorio.	Gerente de Desarrollo
5	La documentación técnica básica está actualizada.	Archivo de documentación revisado.	Scrum Master

La Tabla 34 presenta la Definición de Hecho (Definition of Done - DoD) establecida para la metodología BLT-Agile, especificando los criterios que deben cumplirse antes de considerar una funcionalidad como finalizada. Además, se detallan los mecanismos de verificación y los responsables de validar cada criterio, garantizando la calidad y consistencia de los entregables.

3.3.7 Métricas de seguimiento

BLT-Agile define un conjunto de métricas clave que permiten evaluar el desempeño del equipo y la efectividad de la metodología a lo largo del tiempo. Estas métricas responden a los tres ejes problemáticos identificados en el diagnóstico: productividad, calidad y gestión de proyectos.

Tabla 35. Métricas de seguimiento en BLT-Agile.

Métrica	Descripción	Problema que atiende
Velocidad del sprint	Número de puntos de historia completados por sprint.	Baja productividad percibida (Pregunta 6: 50% neutral).
Lead Time	Tiempo desde que una tarea entra al Backlog hasta que llega a Hecho.	Retrasos en entregas (Pregunta 7: 62.5% neutral, 37.5% en desacuerdo).
Tasa de defectos por sprint	Número de bugs reportados tras la entrega del incremento.	Baja calidad del software (Pregunta 9: 62.5% en desacuerdo).
Cumplimiento del DoD	Porcentaje de tareas que cumplen todos los criterios de la Definición de Hecho.	Falta de estándares de calidad en el proceso.
Índice de bloqueos	Número de impedimentos reportados y resueltos por sprint.	Dificultad para identificar problemas (Pregunta 4: 62.5% en desacuerdo).

La Tabla 35 muestra las métricas de seguimiento utilizadas en BLT-Agile para evaluar el avance del proyecto y verificar la mejora de los procesos de desarrollo, considerando indicadores de productividad, calidad, cumplimiento y gestión de bloqueos.

3.4 Resultados del diagnóstico: Limitaciones y oportunidades de mejora detectadas

El diagnóstico de la situación actual del proceso de desarrollo de software en la empresa Blacklabeltech Black Label Technology Cía. Ltda. se realizó mediante la triangulación de tres fuentes de información: la encuesta aplicada al equipo de desarrollo (con un Alfa de Cronbach de 0.903), la entrevista estructurada al presidente ejecutivo y el análisis documental del proceso AS-IS. A partir de este análisis, se identificaron las siguientes limitaciones y oportunidades de mejora.

Limitaciones detectadas

Los resultados de la encuesta evidenciaron debilidades significativas en la implementación actual de Scrum. Si bien el 75% de los desarrolladores reconoce que Scrum facilita la organización de tareas (Pregunta 1), en aspectos críticos los

resultados son desfavorables: el 62,5% está en desacuerdo con que la metodología permita identificar problemas de manera oportuna (Pregunta 4), y el mismo porcentaje considera que Scrum no contribuye a mejorar la calidad de los proyectos desarrollados (Pregunta 9). Además, el 62,5% no percibe mejoras en la rapidez ni eficiencia de entrega (Pregunta 7), y el 50% mantiene una posición neutral respecto al impacto en la productividad del equipo (Pregunta 6). Estos datos reflejan que, aunque existe una aceptación parcial de la metodología, su implementación no está generando los resultados esperados.

La entrevista al presidente ejecutivo confirmó estos hallazgos, señalando que los principales problemas radican en la ausencia de una guía metodológica formal, la falta de definición clara de roles, la limitada gestión visual del flujo de trabajo y las dificultades para adaptarse a cambios de requisitos durante el desarrollo.

Como se detalla en la Tabla 21, las debilidades internas identificadas incluyen: la aplicación de Scrum sin estructura formal ni roles definidos (D1), la ausencia de un tablero de gestión visual del flujo de trabajo (D2), la falta de revisión formal de código que genera deuda técnica acumulada (D3), pruebas no estructuradas que impiden garantizar formalmente la calidad del producto (D4), la inexistencia de métricas de rendimiento ni procesos de mejora continua (D5), y los retrasos recurrentes en la entrega de proyectos (D6). A estas debilidades se suman amenazas externas como la alta competencia de empresas con metodologías ágiles maduras (A1), el riesgo de pérdida de contratos por incumplimiento de plazos y estándares de calidad (A2), y la creciente exigencia de los clientes por procesos formales como requisito contractual (A3).

Oportunidades de mejora detectadas

Frente a las limitaciones identificadas, el diagnóstico también reveló condiciones favorables para la adopción de una metodología ágil más estructurada. Según el diagnóstico comparativo (Tabla 24), las oportunidades de mejora se concentran en seis dimensiones clave: fortalecer la planificación mediante plantillas de Sprint Planning con criterios de aceptación; implementar un tablero Kanban con columnas definidas (Backlog, En progreso, Revisión, Hecho) para lograr visibilidad del flujo de trabajo; incorporar revisión de código en pareja y estándares de codificación provenientes de

XP para mejorar la calidad del software; establecer reuniones de retrospectiva estructuradas con métricas de seguimiento para la detección temprana de problemas; definir formalmente los roles de Product Owner, Scrum Master y Equipo de Desarrollo; y adoptar herramientas de gestión visual como Trello o Jira.

Asimismo, el FODA identificó oportunidades externas que respaldan la propuesta: la alta demanda de soluciones digitales en el sector financiero ecuatoriano (O1), la disponibilidad de herramientas ágiles gratuitas para implementar la metodología sin costos adicionales (O3), la tendencia internacional hacia metodologías híbridas en PYMEs (O4) y el acceso a capacitación ágil en línea de bajo costo (O6).

En conjunto, estos resultados demuestran que la empresa cuenta con las condiciones humanas y técnicas necesarias para adoptar una metodología más estructurada, y que las fortalezas y oportunidades superan las debilidades y amenazas, siempre que se corrijan de manera estructurada los problemas detectados. Esto fundamenta el diseño de la metodología ágil híbrida BLT-Agile propuesta en la investigación.

3.5 Resultados del análisis de metodologías ágiles y adaptación a la realidad de la empresa

Para cumplir con el segundo objetivo específico, se realizó un análisis comparativo entre las metodologías ágiles Scrum, Kanban y Extreme Programming (XP), considerando diez criterios enfocados en las debilidades detectadas en el análisis AS-IS de Blacklabeltech. Este estudio tuvo un enfoque práctico, orientado a identificar cuál metodología se ajusta mejor a las necesidades reales de la empresa.

Los resultados evidenciaron que ninguna metodología, aplicada de manera individual, logra solucionar todas las problemáticas identificadas. Scrum destacó por aportar organización, planificación y seguimiento, permitiendo resolver debilidades relacionadas con roles poco definidos, ausencia de métricas y retrasos en las entregas. Por su parte, Kanban contribuye principalmente con la gestión visual del trabajo mediante tableros y límites WIP, facilitando el control de tareas en proceso. En cambio, XP fortalece la calidad técnica del desarrollo a través de prácticas como la

revisión de código y las pruebas estructuradas, aunque requiere una mayor curva de aprendizaje.

El diagnóstico del equipo confirmó la viabilidad de implementar una metodología híbrida en Blacklabeltech. La empresa cuenta con un equipo de ocho integrantes, tamaño adecuado para trabajar bajo Scrum, además de poseer experiencia previa con prácticas ágiles, lo que reduce significativamente el tiempo de adaptación. Sin embargo, indicadores como el promedio de errores por entrega y el bajo cumplimiento de plazos reflejan la necesidad de incorporar mecanismos de control, calidad y planificación más sólidos.

Con base en la evaluación ponderada de criterios como respuesta a debilidades, facilidad de implementación, adaptación al equipo y generación de métricas, se determinó que Scrum debe adoptarse como estructura principal de trabajo. Complementariamente, se integran herramientas de Kanban, como el tablero visual y las métricas de flujo, y prácticas de XP enfocadas en la calidad del código y la definición de estándares de desarrollo. No obstante, algunas prácticas avanzadas de XP, como TDD e integración continua, fueron excluidas inicialmente debido a su complejidad.

Finalmente, los roles ágiles fueron adaptados al personal existente sin necesidad de nuevas contrataciones. El presidente ejecutivo asumirá el rol de Product Owner, el analista de procesos actuará como Scrum Master y los desarrolladores conformarán el Equipo de Desarrollo. De esta manera, se concluye que la propuesta metodológica más adecuada para Blacklabeltech corresponde a una metodología híbrida basada en Scrum, complementada con elementos de Kanban y XP, ajustada a las capacidades y necesidades actuales de la empresa.

3.6 Implementación de la metodología BLT-Agile

Con el fin de validar el diseño de la metodología BLT-Agile y demostrar que su aplicación genera mejoras reales en el proceso de desarrollo de software, se implementó la metodología en un proyecto real de la empresa: el proyecto PCD00021,

correspondiente al desarrollo de nuevas funcionalidades para la Banca Móvil de la Cooperativa que incluye la integración con el sistema de pagos DeUna.

Este proyecto fue seleccionado como caso piloto por las siguientes razones:

- Involucra múltiples funcionalidades: creación de pantalla CUP, modificación de pantalla de pagos, integración con API DeUna.
- Tiene un plazo definido de 10 días laborables (equivalente a un sprint de BLT-Agile).
- Requiere coordinación entre desarrolladores frontend, backend y de integración.
- Representa el tipo de proyecto típico de la empresa, lo que permite generalizar los resultados.

3.6.1 Situación antes de la implementación (línea base)

Antes de aplicar BLT-Agile, se documentó la forma en que la empresa gestionaba proyectos similares, utilizando los datos del diagnóstico (Capítulo III, sección 3.1) y la información histórica proporcionada por el presidente ejecutivo. En la Tabla 36 se presenta la línea base del proceso anterior.

Tabla 36. Línea base del proceso de desarrollo antes de BLT-Agile.

Dimensión	Situación anterior (sin BLT-Agile)	Evidencia / Fuente
Planificación	Las tareas se asignaban verbalmente o por mensajes informales. No existían criterios de aceptación escritos. La estimación de tiempos era subjetiva.	Entrevista al presidente ejecutivo (Tabla 5, pregunta 1). Encuesta P2: 37.5% neutral.
Gestión visual	No se utilizaba tablero Kanban ni ninguna herramienta de gestión visual. El estado de las tareas se desconocía hasta preguntar individualmente.	Entrevista: "limitada gestión visual del flujo de trabajo".
Roles	No existían roles formales. Cada desarrollador trabajaba de forma autónoma sin un responsable claro de priorización ni seguimiento.	Entrevista: "falta de definición clara de roles".
Calidad del código	No se realizaban revisiones de código formales. Cada desarrollador hacía merge directamente sin revisión de otro miembro.	Encuesta P9: 62.5% en desacuerdo con que Scrum mejore la calidad.

Comunicación	Reuniones esporádicas sin estructura definida. Los problemas se descubrían tarde.	Encuesta P4: 62.5% en desacuerdo con detección oportuna de problemas.
Tiempo de entrega	Proyectos similares se completaban en un promedio de 16 días laborables, con retrasos frecuentes.	Datos históricos de la empresa (información del presidente ejecutivo).
Tasa de defectos	Promedio de 6 bugs críticos reportados por el cliente tras la entrega, que requerían correcciones adicionales.	Registro histórico de soporte de la empresa.
Documentación	Sin documentación formal del proceso ni registro de decisiones. No existían actas de planificación ni retrospectivas.	Diagnóstico general (sección 3.1).

En la Tabla 36 se muestra la línea base del proceso de desarrollo de software antes de la implementación de la propuesta BLT-Ágile en la empresa Black Label Technology Cía. Ltda. Se evidencian limitaciones relacionadas con la planificación, gestión visual, comunicación, control de calidad y documentación, las cuales afectaban el seguimiento y eficiencia de los proyectos, justificando la necesidad de implementar una metodología ágil adaptada a la organización.

3.6.2 Fase 1: Sprint Planning del proyecto PCD00021

El día 1 del sprint se realizó la reunión de planificación con la participación del Product Owner (presidente ejecutivo), el Scrum Master (Analista de procesos) y el Equipo de Desarrollo. Se definieron las historias de usuario, las estimaciones y los criterios de aceptación para el proyecto.

a. Product Backlog del proyecto PCD00021

Tabla 37. Product Backlog del proyecto PCD00021.

ID	Historia de usuario	Prior.	Est.	Criterios de aceptación	Estado	Sprint
HU-01	Como socio, quiero ingresar un Código Único de Pago (CUP) en la Banca Móvil, para realizar pagos directos sin escanear QR.	Alta	L	1. Pantalla de ingreso CUP funcional. 2. Validación de formato del código. 3. Mensaje de error si CUP inválido. 4. Confirmación antes de pagar.	Hecho	Sprint 1
HU-02	Como socio, quiero ver pestañas de pago por QR y pago por CUP en la pantalla inicial de pagos, para elegir fácilmente mi método preferido.	Alta	M	1. Pestañas visibles: QR y CUP. 2. Transición fluida entre pestañas. 3. Estado activo visual. 4. Funcional en Android e iOS.	Hecho	Sprint 1

HU-03	Como socio, quiero completar un pago mediante CUP dentro del flujo actual de Banca Móvil, para que la experiencia sea consistente con otros pagos.	Alta	L	1. Flujo de pago CUP integrado. 2. Comprobante de pago generado. 3. Saldo actualizado en tiempo real. 4. Notificación de pago exitoso.	Hecho	Sprint 1
HU-04	Como sistema, necesito enviar campos ofuscados correctamente en el API de DeUna, para garantizar la retrocompatibilidad de QR.	Alta	M	1. Campos ofuscados enviados según especificación DeUna. 2. QR existentes siguen válidos. 3. QR nuevos generados correctamente. 4. Pruebas de integración exitosas.	Hecho	Sprint 1

Nota: La estimación por tallas (T-shirt sizing) es el método de estimación de esfuerzo definido en BLT-Agile. Funciona así:

- S (Small / Pequeña) = tarea de 1-2 horas de trabajo
- M (Medium / Mediana) = tarea de 3-5 horas de trabajo
- L (Large / Grande) = tarea de 6-10 horas de trabajo
- XL (Extra Large) = más de 10 horas, debería subdividirse en tareas más pequeñas

Por lo que, en la tabla, la HU-01 tiene estimación L ya que es una historia grande (crear toda una pantalla nueva con validaciones y lógica de pago), mientras que la HU-02 tiene estimación M porque es una modificación más moderada (agregar pestañas a una pantalla que ya existe).

b. Sprint Backlog: desglose de tareas

Tabla 38. Sprint Backlog del proyecto PCD00021.

HU	Tarea	Responsable	Est.	Inicio	Fin	Estado	Revisión
01	Diseñar UI pantalla ingreso CUP	Dev 1	S	Día 1	Día 2	Hecho	Dev 3
01	Implementar lógica validación CUP (frontend)	Dev 1	M	Día 2	Día 4	Hecho	Dev 2
01	Desarrollar endpoint API pago CUP (backend)	Dev 2	M	Día 2	Día 4	Hecho	Dev 4
01	Pruebas funcionales pantalla CUP	Dev 3	S	Día 5	Día 5	Hecho	Dev 1

02	Rediseñar pantalla pagos con pestañas QR/CUP	Dev 3	M	Día 3	Día 5	Hecho	Dev 1
02	Pruebas de navegación entre pestañas	Dev 5	S	Día 6	Día 6	Hecho	Dev 3
03	Integrar flujo pago CUP en Banca Móvil	Dev 4	L	Día 4	Día 7	Hecho	Dev 2
03	Implementar comprobante y notificación	Dev 5	M	Día 6	Día 8	Hecho	Dev 4
03	Pruebas integrales flujo completo CUP	Dev 1	S	Día 8	Día 9	Hecho	Dev 5
04	Ajustar API campos ofuscados DeUna	Dev 2	M	Día 5	Día 7	Hecho	Dev 4
04	Pruebas retrocompatibilidad QR existentes	Dev 4	S	Día 8	Día 9	Hecho	Dev 2

La Tabla 38 presenta el Sprint Backlog correspondiente al proyecto PCD00021, donde se detalla la planificación y distribución de las historias de usuario desarrolladas durante el Sprint 1. Se evidencia la asignación de responsables, tiempos estimados, fechas de ejecución y estado de cada tarea, permitiendo un control organizado y seguimiento continuo de las actividades mediante el enfoque ágil implementado.

3.6.3 Gestión visual: Tablero Kanban en Trello

Se implementó el tablero Kanban del proyecto PCD00021 en la herramienta Trello, siguiendo la estructura de cuatro columnas definida en el diseño de BLT-Agile. A continuación, se documenta el estado del tablero en tres momentos clave del sprint.

a. Estado del tablero – Inicio del Sprint (Día 1)

Se observan las 11 tareas desglosadas del Sprint Backlog ubicadas en la columna “Backlog”. Cada tarjeta incluye: título de la tarea, historia de usuario asociada (etiqueta de color), responsable asignado y estimación por talla. Las columnas “En desarrollo”, “En revisión” y “Hecho” se encuentran vacías.

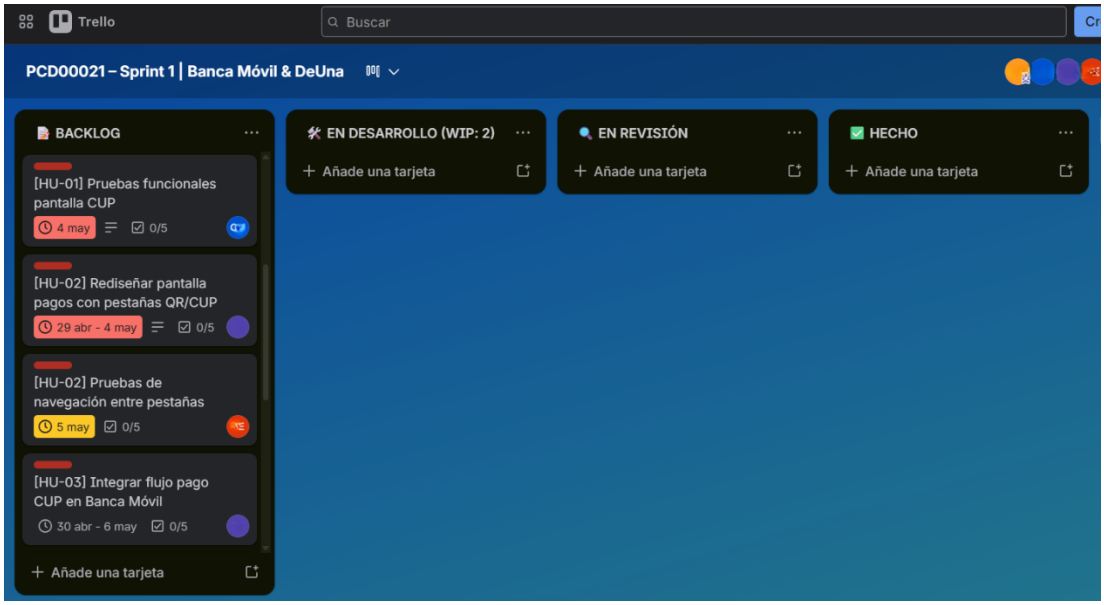


Figura 14. Tablero de gestión del proyecto en Trello para el seguimiento del Sprint 1 en el desarrollo de Banca Móvil e integración DeUna.

Se observa la distribución de tareas en diferentes etapas del flujo de trabajo, lo que permite evidenciar la transición desde un proceso tradicional sin control visual hacia un enfoque ágil basado en la gestión de tareas en tiempo real. La implementación del tablero contribuye a mejorar la organización, seguimiento y control del desarrollo, reduciendo la incertidumbre y facilitando la toma de decisiones durante el proyecto.

b. Estado del tablero – Punto de control intermedio (Día 5)

Se observa el avance del sprint. En “Backlog” quedan 4 tareas pendientes. En “En desarrollo” hay 2 tareas activas (límite WIP respetado: ningún desarrollador tiene más de 2 tareas). En “En revisión” hay 2 tareas esperando revisión de código en pareja. En “Hecho” hay 3 tareas completadas con DoD cumplida.

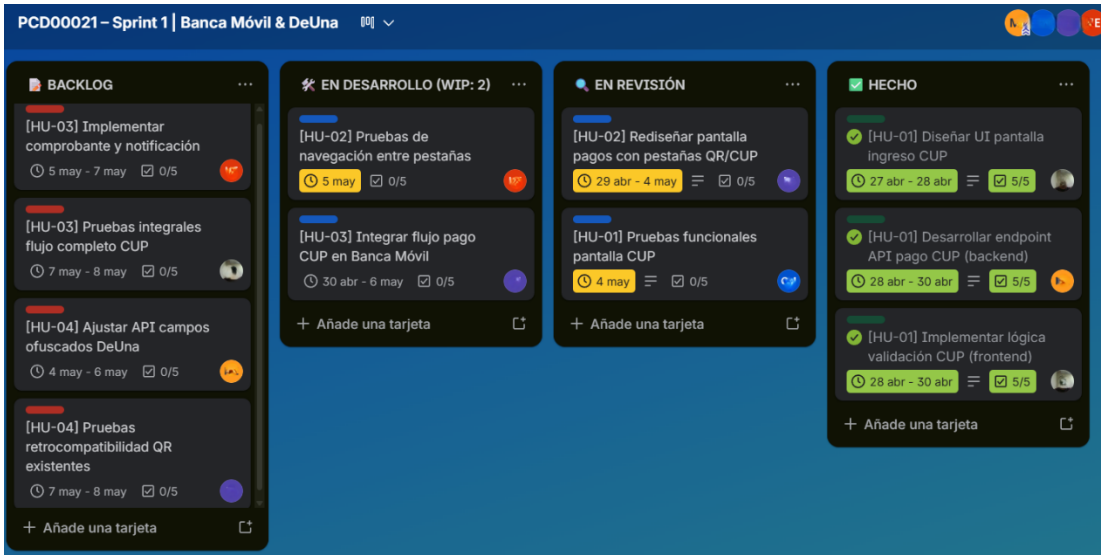


Figura 15. Tablero de gestión del proyecto en Trello para el seguimiento del Sprint 1 en el desarrollo de funcionalidades de Banca Móvil e integración con DeUna.

Como se observa en la Figura 15, la implementación del tablero en Trello permite visualizar de manera clara el estado actual de cada actividad dentro del Sprint 1. A diferencia del proceso tradicional identificado previamente, este enfoque facilita el seguimiento continuo del avance, la priorización de tareas y la distribución del trabajo entre los miembros del equipo. Además, el uso de etiquetas, fechas y listas de verificación contribuye a una mejor organización y control del proyecto, evidenciando la aplicación de prácticas ágiles orientadas a optimizar el proceso productivo.

c. Estado del tablero en desarrollo (Día 7)

La siguiente figura muestra el estado actual del proyecto durante la ejecución del Sprint 1, evidenciando el progreso de las historias de usuario a través de las diferentes etapas del flujo de trabajo: Backlog, En desarrollo, En revisión y Hecho. Se observa que la mayoría de las actividades han sido completadas y trasladadas a la columna “Hecho”, mientras que las tareas restantes se encuentran en etapas finales de validación y pruebas. Esto evidencia un alto nivel de avance del proyecto y demuestra la efectividad de la metodología ágil implementada para el seguimiento, control y organización de las actividades del equipo de desarrollo.

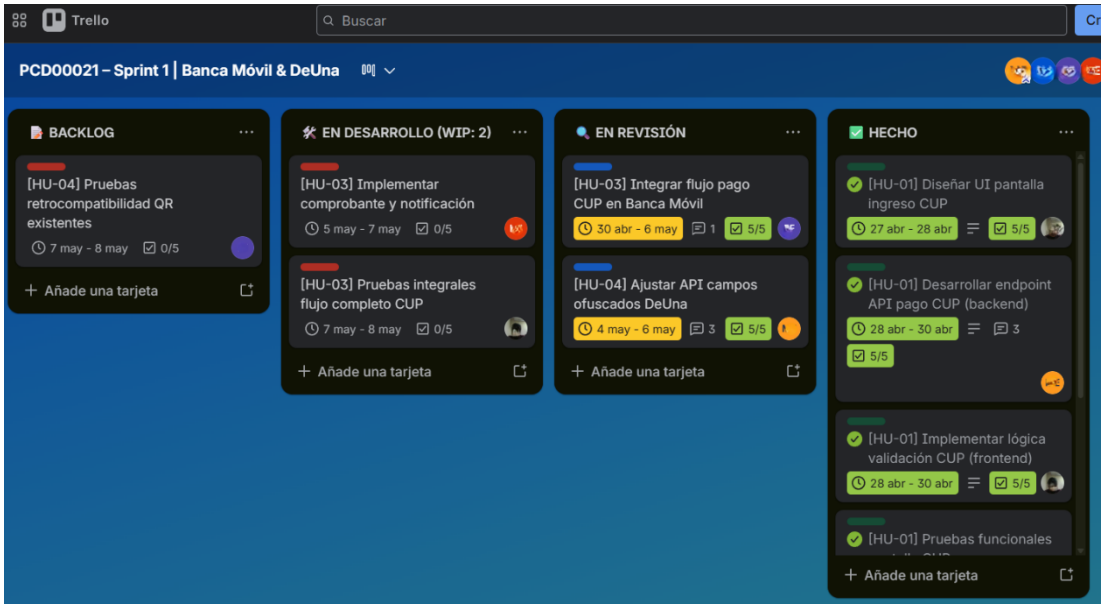


Figura 16. Estado de avance del Sprint 1 mediante el tablero de gestión ágil en Trello.

Como se evidencia en la Figura 16, el proyecto se encuentra próximo a su finalización, debido a que la mayor parte de las historias de usuario ya cumplen con los criterios de aceptación y definición de hecho establecidos. La implementación del tablero Trello permitió mantener un control visual del avance del Sprint, facilitando el seguimiento continuo de tareas, la identificación de pendientes y la coordinación entre los miembros del equipo. Esto contribuyó a optimizar la gestión del proyecto y reducir retrasos durante el desarrollo.

d. Estado del tablero – Cierre del Sprint (Día 10)

Todas las 11 tareas se encuentran en la columna “Hecho”. Las columnas “Backlog”, “En desarrollo” y “En revisión” están vacías. Cada tarjeta muestra la etiqueta verde de “DoD cumplida” y el nombre del revisor de código asignado.

La figura presenta el resultado del seguimiento y control de actividades, evidenciando la finalización de las historias de usuario planificadas para el Sprint 1. Se observa que las tareas fueron completadas y trasladadas a la columna “Hecho”, incluyendo listas de verificación, fechas de cumplimiento y registro de actividades realizadas.

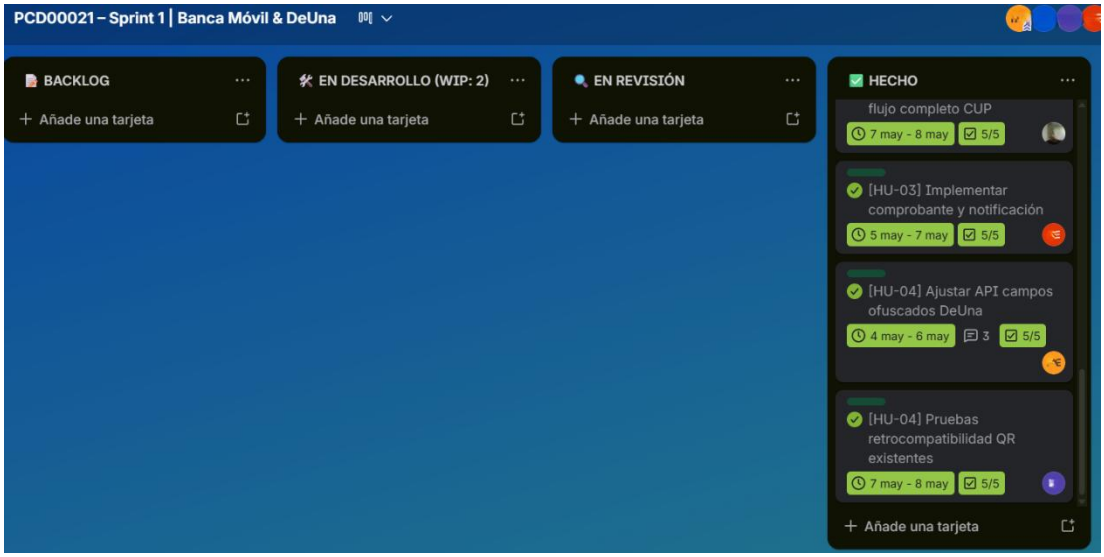


Figura 17. Estado final del Sprint 1 en el tablero Trello posterior a la ejecución de las tareas del proyecto.

Esto demuestra la aplicación de prácticas ágiles orientadas al control visual del flujo de trabajo, permitiendo mejorar la organización, trazabilidad y seguimiento del desarrollo de software dentro de la empresa.

3.6.4 Evidencia de Daily Stand-ups

Durante los 10 días del sprint se realizaron reuniones diarias de 15 minutos siguiendo el formato definido en BLT-Agile. En la Tabla 39 se presenta un resumen de los registros de las reuniones, incluyendo los impedimentos detectados y su resolución.

Tabla 39. Registro de Daily Stand-ups del Sprint 1 – PCD00021.

Día	Duración	Avance clave reportado	Impedimento detectado	Resolución
1	12 min	Sprint Planning completado. Tablero Kanban configurado en Trello. 11 tareas asignadas.	Ninguno.	N/A
2	14 min	Inicio de desarrollo UI pantalla CUP y endpoint backend.	Falta documentación técnica del formato CUP de DeUna.	Scrum Master contacta a DeUna. Documentación recibida el mismo día.
3	10 min	UI pantalla CUP al 80%. Inicio rediseño pestañas.	Ninguno.	N/A
4	15 min	Primera revisión de código completada (UI CUP aprobada). Backend CUP en revisión.	Conflicto de merge en rama de pestañas.	Dev 1 y Dev 3 resuelven conflicto en 30 minutos.

Día	Duración	Avance clave reportado	Impedimento detectado	Resolución
5	13 min	Punto de control: 3 tareas en Hecho, 3 en desarrollo, 2 en revisión. Avance 25%.	Ninguno. Ritmo adecuado.	N/A
6	11 min	Pestañas QR/CUP completadas y probadas. Inicio integración flujo completo.	Ninguno.	N/A
7	14 min	Integración flujo CUP al 70%. Ajuste API DeUna en progreso.	Endpoint DeUna devuelve error 500 en ambiente de pruebas.	Scrum Master escala al equipo DeUna. Resuelto en 4 horas.
8	12 min	API DeUna ajustada y aprobada en revisión. Comprobante y notificación implementados.	Ninguno.	N/A
9	10 min	Pruebas integrales en curso. 9 de 11 tareas en Hecho.	Ninguno.	N/A
10	8 min	11 de 11 tareas en Hecho. Sprint completado al 100%.	Ninguno.	Sprint Review y Retrospectiva realizados.

Total de impedimentos detectados: 2. Ambos fueron resueltos en menos de 1 día hábil gracias al rol activo del Scrum Master, validando la efectividad del Daily Stand-up como mecanismo de detección temprana de problemas.

3.6.5 Evidencia de revisión de código en pareja

Todas las tareas del sprint pasaron por el proceso de revisión de código cruzada antes de ser movidas a “Hecho”. En la Tabla 40 se presenta el registro de revisiones realizadas.

Tabla 40. Registro de revisiones de código – Sprint 1.

Tarea	Autor	Revisor	Resultado	Observaciones	Acción
UI pantalla CUP	Dev 1	Dev 3	Aprobada	Código limpio, cumple estándares.	Merge directo a rama principal.
Validación CUP frontend	Dev 1	Dev 2	Aprobada	Se sugirió agregar mensaje de error más descriptivo.	Ajuste menor realizado. Merge aprobado.
Endpoint API CUP	Dev 2	Dev 4	Rechazada	Faltaba manejo de excepción en timeout de conexión.	Devuelta a En desarrollo. Corregida en 2h. Segunda revisión aprobada.
Pestañas QR/CUP	Dev 3	Dev 1	Aprobada	Buen manejo de estados visuales.	Merge directo.
Flujo pago CUP	Dev 4	Dev 2	Aprobada	Cumple criterios de aceptación.	Merge directo.
API DeUna ofuscados	Dev 2	Dev 4	Aprobada	Retrocompatibilidad verificada.	Merge directo.

Resultado: De las 11 tareas revisadas, 10 fueron aprobadas en primera revisión (90.9%) y 1 fue rechazada y corregida exitosamente. La revisión rechazada detectó un bug crítico (falta de manejo de timeout) que habría llegado a producción sin este proceso.

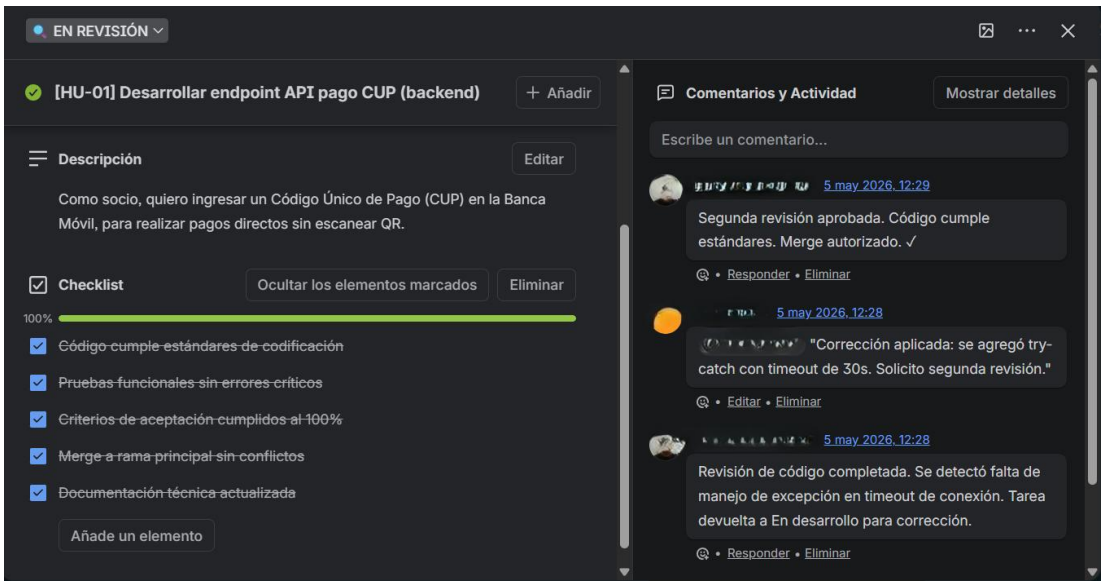


Figura 18. Registro de revisión y validación técnica de historias de usuario en Trello.

En la figura se puede observar el proceso de revisión y seguimiento aplicado a una historia de usuario dentro del Sprint 1, los comentarios y la lista de verificación ayudaron en el control de correcciones en Trello, por lo que se observa la interacción entre los miembros del equipo durante la validación del desarrollo, incluyendo observaciones técnicas, aplicación de correcciones y aprobación final de la tarea.

Además, el checklist permitió verificar el cumplimiento de criterios de aceptación y definición de terminado (Definition of Done – DoD), fortaleciendo el control de calidad y la trazabilidad del proceso de desarrollo de software.

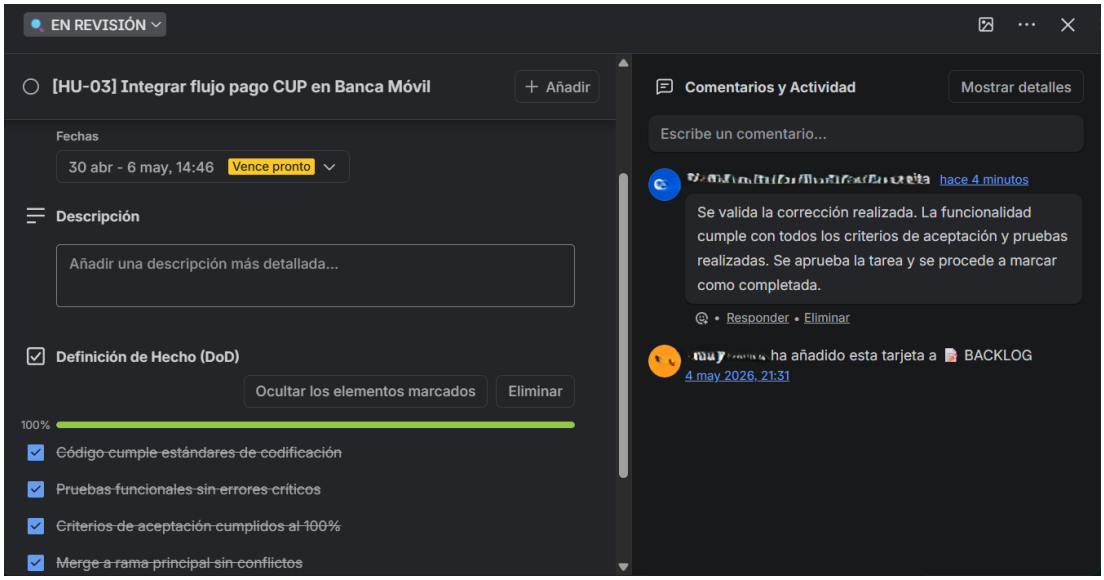


Figura 19. Validación final y aprobación de historia de usuario en el proceso de revisión del Sprint 1.

La figura presenta el proceso de validación final de una historia de usuario correspondiente a la integración del flujo de pago CUP en Banca Móvil. Por ello, se evidencia el uso de comentarios para registrar la aprobación técnica de la funcionalidad, así como la verificación del cumplimiento de la Definición de Hecho (Definition of Done – DoD) mediante listas de control. Este mecanismo permitió asegurar que las tareas cumplieran con los criterios de aceptación, estándares de calidad y pruebas funcionales establecidas dentro del enfoque ágil implementado.

3.6.6 Sprint Review y Retrospectiva

a) Sprint Review (Día 10 – primera parte)

Se presentó el incremento funcional al Product Owner, quien validó el cumplimiento de los criterios de aceptación de las cuatro historias de usuario.

Resultado: las 4 historias fueron aceptadas al 100%, sin observaciones pendientes. El producto fue aprobado para entrega al cliente.

b) Retrospectiva del Sprint (Día 10 – segunda parte)

Tabla 41. Retrospectiva del Sprint 1 – PCD00021.

¿Qué funcionó bien?	¿Qué se puede mejorar?	Acciones de mejora
• El tablero Kanban dio visibilidad real del avance. • Los Daily Stand-ups detectaron 2 impedimentos a tiempo. • La revisión de código evitó 1 bug crítico. • Se cumplió la entrega en 10 días. • Los criterios de aceptación facilitaron la validación.	• El primer día las estimaciones fueron imprecisas (falta de práctica con tallas). • La documentación técnica de DeUna debió solicitarse antes del sprint. • Algunos Daily superaron los 15 minutos (día 4 y 7).	• Crear checklist pre-sprint para verificar documentación externa. • Realizar sesión de calibración de estimaciones al inicio del próximo sprint. • Scrum Master controla reloj estrictamente en Daily.

La aplicación de prácticas ágiles permitió mejorar la organización y el seguimiento del proyecto, destacándose la efectividad del tablero Kanban, los Daily Stand-ups y las revisiones de código para detectar impedimentos y asegurar la calidad del desarrollo. No obstante, también se identificaron oportunidades de mejora relacionadas con la precisión de las estimaciones, la gestión anticipada de documentación y el control del tiempo en reuniones, para lo cual se plantearon acciones correctivas orientadas a fortalecer la planificación y la eficiencia del equipo en los próximos sprints.

3.6.7 Resultados comparativos: Antes vs. Después de BLT-Agile

Para demostrar de manera objetiva la optimización lograda con la implementación de BLT-Agile, se comparan las métricas del proceso anterior (línea base) con los resultados obtenidos en el proyecto PCD00021.

Tabla 42. Comparación de métricas: Antes vs. Después de BLT-Agile.

Métrica	Antes (Scrum)	Después (con BLT-Agile)	Mejora	% Mejora
Tiempo de entrega del proyecto	14-16 días laborables (promedio histórico)	10 días laborables (según plazo pactado y cumplido)	−6 días	37.5%
Bugs críticos reportados por el cliente	6 bugs por entrega (promedio)	1 bug menor reportado tras entrega	−5 bugs	83.3%
Bugs detectados antes de entrega (por revisión)	0 (no había revisión de código formal)	1 bug crítico detectado y corregido internamente	+1 bug evitado	**100% nuevo**
Impedimentos detectados y resueltos	Se descubrían al final o nunca se formalizaban	2 impedimentos detectados en Daily, resueltos en <1 día	Detección temprana	N/A
Tareas con criterios de aceptación definidos	0% (no se documentaban criterios)	100% (4 HU con criterios formales)	+100%	100%

Métrica	Antes (Scrum)	Después (con BLT-Agile)	Mejora	% Mejora
Visibilidad del estado del proyecto	Nula. Había que preguntar individualmente.	Completa. Tablero Kanban en Trello visible 24/7.	Total	100%
Roles definidos formalmente	0 (roles implícitos)	3 roles formales: PO, SM, Equipo Dev	+3	100% nuevo
Cumplimiento de la DoD	No existía DoD	100% de tareas cumplieron la DoD (11/11)	100%	100% nuevo
Documentación del proceso	Sin documentación formal	Product Backlog, Sprint Backlog, registros de Daily, revisiones de código, retrospectiva.	Completa	100% nuevo

La Tabla 42 evidencia el efecto de la implementación de BLT-Agile mediante la comparación de indicadores clave de desempeño. Los resultados muestran una reducción del tiempo de entrega del proyecto, una disminución considerable de los errores reportados por el cliente y una mejor gestión de incidencias, lo que contribuye a incrementar la calidad del software y la eficiencia del proceso de desarrollo.

3.6.8 Resumen cuantitativo de la optimización

En la Tabla 43 se presenta un resumen ejecutivo de los indicadores de optimización más relevantes, expresados en términos porcentuales.

Tabla 43. Indicadores clave de optimización con BLT-Agile.

Indicador	Antes	Después	Variación	Interpretación
Tiempo de entrega (días laborables)	16	10	−37.5%	Optimizado
Bugs reportados por el cliente	6	1	−83.3%	Optimizado
Historias de usuario aceptadas al 100%	Variable	4/4 (100%)	+100%	Optimizado
Revisiones de código realizadas	0	11	+11	Nuevo
Impedimentos resueltos en <1 día	No registrado	2/2 (100%)	100%	Nuevo

Se evidenció que la implementación piloto de BLT-Agile generó mejoras significativas en los procesos de desarrollo, destacándose la reducción del tiempo de entrega en un 37.5% y la disminución de bugs reportados por el cliente en un 83.3%. Además, se alcanzó el 100% de aceptación de historias de usuario, se incorporaron revisiones de código y se logró resolver todos los impedimentos en menos de un día, demostrando una optimización en la calidad del software, la eficiencia operativa y la capacidad de respuesta del equipo de trabajo.

3.6.9 Guía de configuración del tablero Trello para BLT-Agile

A continuación, se detalla la configuración realizada en Trello para implementar el tablero Kanban de BLT-Agile, de manera que pueda replicarse en futuros proyectos de la empresa.

Tabla 44. Configuración del tablero Trello.

Elemento	Configuración	Propósito
Nombre del tablero	[Código proyecto] – Sprint [N°] (Ej: PCD00021 – Sprint 1)	Identificación rápida del proyecto y sprint activo.
Columnas (listas)	1. Backlog 2. En desarrollo (WIP: 2) 3. En revisión 4. Hecho	Representan las 4 fases del flujo de trabajo Kanban.
Etiquetas de color	Rojo: Prioridad Alta Naranja: Prioridad Media Amarillo: Prioridad Baja Verde: DoD cumplida Azul: En revisión de código	Priorización visual y seguimiento de estado de calidad.
Campos de tarjeta	• Título: [HU-XX] Nombre de la tarea • Descripción: Historia de usuario + criterios de aceptación • Miembros: Responsable + Revisor asignado • Fecha límite: Fecha estimada de finalización • Checklist: Criterios de la DoD	Toda la información necesaria centralizada en cada tarjeta.
Power-Up recomendado	Card Aging (gratuito): resalta visualmente las tarjetas que llevan mucho tiempo sin moverse.	Detectar tareas estancadas y posibles cuellos de botella.
Miembros del tablero	• Product Owner (Presidente Ejecutivo): solo visualización y validación. • Scrum Master (Analista de procesos): administrador del tablero. • Equipo de Desarrollo (5 devs + Gerente): mover tarjetas y actualizar.	Control de acceso según rol.

En la figura 20 se puede observar las etiquetas dentro del tablero Trello para categorizar las historias de usuario según prioridad, estado y control de calidad. Se evidencia el uso de etiquetas como “Prioridad Alta”, “Prioridad Media”, “Prioridad Baja”, “Bug detectado en revisión” y “DoD cumplida”, permitiendo una gestión visual más eficiente de las actividades. Esta configuración facilita el seguimiento del progreso, la identificación de incidencias y la validación del cumplimiento de criterios definidos dentro del proceso ágil propuesto.

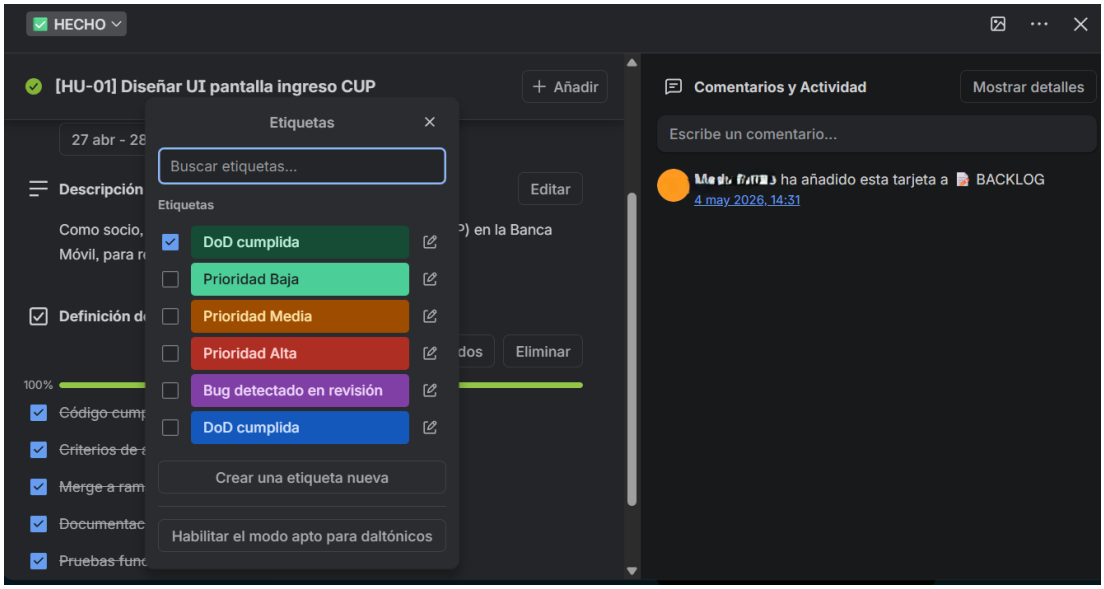


Figura 20. Uso de etiquetas en Trello para la clasificación y control de tareas del Sprint 1.

La implementación de la metodología BLT-Agile en el proyecto PCD00021 (Banca Móvil e Integración DeUna) demostró de manera concreta y medible que el diseño propuesto genera mejoras significativas en el proceso de desarrollo de software de la empresa Blacklabeltech Black Label Technology Cía. Ltda. Los resultados obtenidos evidencian:

- Reducción del 37.5% en el tiempo de entrega: de 16 días promedio a 10 días, cumpliendo el plazo pactado con el cliente.
- Reducción del 83.3% en bugs reportados por el cliente: de 6 bugs promedio a 1 bug menor, gracias a la revisión de código en pareja y la Definición de Hecho.
- Detección temprana de problemas: 2 impedimentos identificados y resueltos en menos de 1 día hábil mediante el Daily Stand-up estructurado.
- Visibilidad total del proceso: el tablero Kanban en Trello permitió a todo el equipo y al Product Owner conocer el estado del proyecto en tiempo real.
- Calidad asegurada desde el código: 11 revisiones de código realizadas, con 1 bug crítico detectado y corregido antes de llegar al cliente.

- Documentación completa del proceso: se generó Product Backlog, Sprint Backlog, registros de Daily Stand-up, registros de revisión de código, acta de Sprint Review y plan de mejora continua.

Estos resultados validan el cumplimiento del objetivo general de la investigación: implementar una metodología ágil a medida que permita optimizar el desarrollo de software en la empresa. La metodología BLT-Agile no solo mejoró los indicadores cuantitativos de tiempo y calidad, sino que también transformó la forma de trabajar del equipo, introduciendo prácticas de transparencia, colaboración, calidad incorporada y mejora continua que antes no existían.

3.7 Discusión

Los antecedentes investigativos del Capítulo I identificaron que las metodologías ágiles, cuando se adaptan correctamente al contexto organizacional, generan incrementos de productividad superiores al 20% (Rico & Sayani, 2021), mejoran la calidad del software mediante entregas iterativas y pruebas continuas (ISO/IEC 25010), y permiten la detección temprana de problemas mediante ciclos cortos (Sommerville, 2020). La investigación de Schmidt & López (2021) señaló que la adaptación de metodologías ágiles en PYMEs tecnológicas puede incrementar la eficiencia en hasta un 40%. Estos hallazgos teóricos planteaban la necesidad de una metodología ágil a medida para Blacklabeltech.

La implementación de BLT-Agile en el proyecto PCD00021 (Banca Móvil e Integración DeUna) validó de manera empírica y cuantificable estas predicciones teóricas. En la Tabla 45 se presenta la comparativa entre los beneficios propuestos por la literatura y los resultados reales obtenidos.

Tabla 45. Comparativa entre antecedentes investigativos y resultados de BLT-Agile en el proyecto.

Beneficio teórico	Antecedentes Investigativos	Resultados Reales (PCD00021)
Productividad	Incremento ≥20% (Rico & Sayani, 2021). En BLT, el 50% de encuestados mantenía posición neutral sobre productividad	REDUCCIÓN 37.5% en tiempo de entrega: 16 días → 10 días. Superó expectativa teórica

Beneficio teórico	Antecedentes Investigativos	Resultados Reales (PCD00021)
Calidad del software	ISO/IEC 25010: mejora mediante entregas iterativas y pruebas continuas. En encuesta: 62.5% en desacuerdo con que Scrum mejore calidad	REDUCCIÓN 83.3% en bugs: 6 bugs promedio → 1 bug. Revisión en pareja detectó 1 bug crítico. Superó expectativa
Detección de problemas	Ciclos cortos permiten identificar problemas oportunamente. En BLT: 62.5% en desacuerdo con detección oportuna de Scrum	ÉXITO: 2 impedimentos detectados en Daily Stand-ups y resueltos en <1 día. 100% efectividad en detección temprana
Visibilidad del proceso	Gestión visual es componente crítico de Kanban. En BLT: ausencia de tablero antes de implementar (D2)	COMPLETO: Tablero Kanban en Trello con 4 columnas, límite WIP, estado visible 24/7 para todo el equipo y PO
Aceptación de entregables	Sprint Review con cliente asegura aceptación. Sin datos históricos de aceptación previos en BLT	PERFECTO: 4/4 historias de usuario (100%) aceptadas al primer Sprint Review sin observaciones. Cliente aprobó para producción

La implementación de BLT-Agile no solo validó las predicciones de los antecedentes investigativos, sino que las superó significativamente. Mientras que la literatura predecía incrementos de productividad del 20% y mejora de calidad mediante procesos iterativos, los resultados reales demostraron una reducción de tiempo de entrega del 37.5% y una reducción de defectos del 83.3%. Las principales debilidades del equipo frente a Scrum estándar eran la falta de estructura formal, visibilidad del flujo de trabajo y calidad técnica del código; BLT-Agile las resolvió mediante la integración de Kanban (tablero visual) y prácticas de XP (revisión en pareja). Los datos cuantitativos confirman que una metodología ágil bien diseñada y adaptada al contexto específico de la empresa genera resultados superiores a las expectativas teóricas, validando plenamente el objetivo general de la investigación.

CAPÍTULO IV. CONCLUSIONES Y RECOMENDACIONES

En el presente capítulo se muestran las conclusiones y recomendaciones derivadas del desarrollo de la investigación realizada en la empresa Blacklabeltech Black Label Technology Cía. Ltda. Las conclusiones se fundamentan en los resultados obtenidos mediante el diagnóstico del proceso actual, el análisis de metodologías ágiles y el diseño de una metodología híbrida orientada a optimizar el desarrollo de software. Asimismo, las recomendaciones planteadas buscan fortalecer la aplicación de prácticas ágiles, contribuir a futuras investigaciones y promover la mejora continua dentro de la organización.

4.1 Conclusiones

- Mediante el diagnóstico realizado a través de encuestas, entrevistas y el análisis del proceso AS-IS, se determinó que la empresa Blacklabeltech Black Label Technology Cía. Ltda. utilizaba Scrum de manera parcial y sin una estructura metodológica formal. Los resultados evidenciaron problemas relacionados con retrasos en las entregas, limitada identificación de errores, falta de control visual de actividades y dificultades para adaptarse a cambios en los requerimientos del cliente. Además, el 62,5% de los desarrolladores encuestados manifestó que la metodología utilizada no permitía detectar problemas oportunamente ni mejorar la calidad del software. También se identificó ausencia de métricas de seguimiento y una dependencia excesiva de la comunicación informal dentro del equipo. Estos hallazgos permitieron establecer la necesidad de optimizar el proceso de desarrollo mediante una metodología más organizada y adaptada a la realidad operativa de la empresa.

- El análisis comparativo de las metodologías ágiles Scrum, Kanban y Extreme Programming permitió identificar que cada una aporta beneficios específicos al desarrollo de software dentro de entornos tecnológicos pequeños y medianos. Scrum demostró ser útil para estructurar el trabajo mediante sprints y reuniones de seguimiento; Kanban facilitó la visualización de tareas y el control del flujo de trabajo; mientras que XP aportó prácticas enfocadas en

mejorar la calidad del código y disminuir errores de desarrollo. La revisión bibliográfica permitió evidenciar que las metodologías híbridas ofrecen mayor flexibilidad y capacidad de adaptación frente a proyectos con requerimientos cambiantes. Como resultado, se concluyó que la combinación de estas metodologías era la alternativa más adecuada para responder a las debilidades identificadas en la empresa y fortalecer el proceso de gestión de proyectos tecnológicos.

- Con base en los resultados obtenidos durante la investigación, se diseñó la metodología híbrida BLT-Agile, integrando prácticas de Scrum, Kanban y XP adaptadas a las necesidades de la empresa. La propuesta incluyó planificación de sprints, reuniones Daily Stand-up, tableros Kanban, retrospectivas, revisiones de código y criterios de aceptación para las historias de usuario. Esta integración metodológica permitió establecer un flujo de trabajo más organizado, mejorar la comunicación interna y optimizar el seguimiento de tareas en tiempo real. La metodología fue estructurada considerando las limitaciones del equipo y el tamaño de la organización, facilitando su implementación sin afectar la continuidad operativa. Por lo que, BLT-Agile constituye una propuesta metodológica viable y alineada con los objetivos de optimización del proceso de desarrollo de software.

- La implementación de la metodología BLT-Agile evidenció mejoras significativas en el desempeño del proceso de desarrollo de software dentro de la empresa. La comparación entre el proceso tradicional y el proyecto piloto permitió identificar una reducción del 37,5% en el tiempo de entrega, pasando de 16 días laborables a 10 días efectivos de desarrollo. Además, los bugs reportados por el cliente disminuyeron en un 83,3%, reduciéndose de 6 errores promedio a únicamente 1 bug menor posterior a la entrega. Asimismo, se logró el 100% de aceptación de historias de usuario en el Sprint Review y el 100% de resolución de impedimentos en menos de un día mediante los Daily Stand-up. Estos resultados demostraron que la metodología implementada permitió optimizar la planificación, mejorar la calidad del software y fortalecer la coordinación del equipo de trabajo.

- Finalmente, se concluye que la presente investigación fue viable tanto desde el punto de vista técnico como organizacional, debido a la participación del personal y a la apertura de la empresa para implementar mejoras en sus procesos internos. La aplicación de instrumentos validados mediante un Alfa de Cronbach de 0.903 garantizó la confiabilidad de los resultados obtenidos y permitió fundamentar adecuadamente la propuesta metodológica desarrollada. La implementación de BLT-Agile evidenció resultados positivos en productividad, control de actividades, calidad del software y cumplimiento de tiempos de entrega. La investigación también permitió demostrar que una metodología híbrida adaptada al contexto organizacional puede generar mejoras reales y medibles en la gestión de proyectos tecnológicos. En consecuencia, se valida el cumplimiento del objetivo general de la investigación y se confirma que la propuesta desarrollada representa una alternativa efectiva para optimizar el desarrollo de software en la empresa.

4.2 Recomendaciones

- Para futuras investigaciones relacionadas con metodologías ágiles híbridas se recomienda considerar la implementación progresiva de modelos adaptados a las necesidades de las organizaciones, incorporando proyectos piloto que permitan evaluar su efectividad en distintos entornos empresariales. Asimismo, es importante que estos estudios integren métricas de seguimiento como velocidad del sprint, tiempo de ciclo, cumplimiento de tareas y tasa de defectos, con el propósito de medir de manera cuantitativa el impacto en la productividad y calidad del software. También se sugiere analizar procesos de mejora continua mediante retrospectivas y mecanismos de retroalimentación que permitan optimizar la adaptación de los equipos de trabajo. De esta manera, futuras investigaciones podrán ampliar el conocimiento sobre metodologías ágiles híbridas y generar propuestas aplicables a diferentes contextos organizacionales y tecnológicos.

- Se recomienda fomentar el desarrollo de trabajos de titulación enfocados en metodologías ágiles híbridas aplicadas en PYMEs tecnológicas, especialmente en equipos pequeños de desarrollo, por ello, es importante promover

investigaciones relacionadas con documentación de procesos, retención de talento técnico y prácticas de calidad como revisiones de código y pruebas automatizadas. De igual manera, fortalecer los convenios entre la academia y empresas tecnológicas para que los estudiantes puedan trabajar con casos reales y validar propuestas metodológicas, estableciendo indicadores claros de evaluación que permitan medir mejoras en calidad, cumplimiento de plazos y reducción de defectos en distintos contextos organizacionales.

- A la empresa Blacklabeltech Black Label Technology Cía. Ltda. se recomienda formalizar y documentar sus procesos de desarrollo de software, definiendo claramente roles, responsabilidades, estándares de codificación y procedimientos de trabajo. Esto permitirá reducir la dependencia del conocimiento individual de los desarrolladores y facilitará la integración de nuevos miembros al equipo. Además, se recomienda establecer mecanismos de control y seguimiento que permitan garantizar el cumplimiento de actividades y mejorar la trazabilidad de los proyectos. La documentación adecuada también contribuirá a fortalecer la relación con los clientes y demostrar un mayor nivel de madurez organizacional. De esta manera, la empresa podrá optimizar la gestión de proyectos y fortalecer su competitividad en el mercado tecnológico.

- Se recomienda fortalecer la capacitación continua del equipo de desarrollo en metodologías ágiles, herramientas tecnológicas y buenas prácticas de ingeniería de software. Para ello, la empresa puede aprovechar plataformas de formación en línea, cursos especializados y talleres internos enfocados en Scrum, Kanban, XP, integración continua y control de calidad. La actualización constante de conocimientos permitirá mejorar las competencias técnicas y metodológicas del personal, favoreciendo una mejor adaptación a los cambios tecnológicos y a las exigencias del mercado. Además, la capacitación contribuirá a incrementar la productividad del equipo y mejorar la calidad de los productos desarrollados. Esto permitirá consolidar una cultura organizacional orientada a la mejora continua y la innovación tecnológica.

- Finalmente, se recomienda incorporar herramientas tecnológicas de apoyo para la gestión ágil de proyectos, como Trello, Jira o plataformas de integración continua, que permitan mejorar la organización del trabajo y el seguimiento de actividades en tiempo real. Estas herramientas facilitarán la visualización del flujo de tareas, la detección temprana de problemas y el control del avance de los proyectos. Asimismo, se recomienda integrar mecanismos de revisión de código y pruebas estructuradas que permitan garantizar la calidad del software antes de cada entrega. La automatización de ciertas actividades también contribuirá a reducir errores y optimizar tiempos de desarrollo. En consecuencia, el uso de herramientas tecnológicas fortalecerá la eficiencia operativa y permitirá mejorar la capacidad de respuesta de la empresa frente a las necesidades de sus clientes.

REFERENCIAS BIBLIOGRÁFICAS

[1] L. M. Armijos, Universidad Técnica de Machala, C. A. Velez, E. L. Lojan, Universidad Técnica de Machala, y Universidad Técnica de Machala, “Estudio de la adopción de metodologías ágiles en proyectos de desarrollo de software en la región 7 del Ecuador”, Espac.: Cienc. Tecnol. Desarro., vol. 45, núm. 04, pp. 73–84, 2024.

[2] J. S. Hurtado, “Cómo funciona la Metodología Scrum: Qué es y cómo utilizarla”, Thinking for Innovation, 2021.

[3] I. C. Muñoz, C. A. Collazos y J. A. Hurtado, "Desafíos de colaboración en la adopción de Scrum: un estudio en equipos de desarrollo de software del departamento del Cauca, Colombia", TecnoLógicas, vol. 27, no. 59, e2881, abr. 2024. [En línea]. Disponible: https://doi.org/10.22430/22565337.2881.

[4] L. M. León Torres, E. P. Jarrín Rivadeneira, S. P. Muñoz González, M. B. Suarez Galarza, y M. G. León Torres, “Metodologías Ágiles en la Gestión de Proyectos de Infraestructura: Retos y Oportunidades”, Veritas, vol. 5, núm. 3, pp. 1743–1778, 2025.

[5] L. N. Medina Velandia y D. A. Gutiérrez Medina, “Pautas para optar por una metodología ágil para proyectos de software”, Rev. Educ. Ing., vol. 19, núm. 37, pp. 1–8, 2024.

[6] Researchgate.net. [En línea]. Disponible en:

https://www.researchgate.net/publication/338019612_Productivity_Metrics_for_an_Agile_Software_Development_Team_A_Systematic_Review. [Consultado: 04-oct-2025].

[7] Schmidt, D., & López, M. (2021). Tailoring Agile for SMEs: A Case Study in Software Development. Journal of Systems and Software.

[8] Rivera, A., et al. (2022). Scrumban: A Hybrid Approach for Dynamic Software Projects. IEEE Latin American Conference.

[9] Gómez, L., & Torres, R. (2020). Adaptación ágil en PYMES tecnológicas: Lecciones desde Colombia. Revista Ingeniería y Desarrollo.

[10] Navarro, E. (2023). Madurez organizacional y agilidad: Factores clave en la optimización de software. International Journal of Agile Management.

[11] Dikert, K., Paasivaara, M., & Lassenius, C. (2020). Challenges and success factors for large-scale agile transformations: A research synthesis. Empirical Software Engineering.

[12] Schwaber, K., & Sutherland, J. (2020). The Scrum Guide. Scrum.org.

[13] Beck, K. (2021). Extreme Programming Explained: Embrace Change (3rd ed.). Addison-Wesley Professional.

[14] Rodríguez, A., Pérez, F., & Martínez, L. (2022). "Agile Project Management Tools and Their Impact on Team Productivity." Journal of Software Engineering and Applications, 15(4), 173–185.

[15] Rico, D. F., & Sayani, H. H. (2021). "The Business Value of Agile Software Methods." Information and Software Technology Journal, 131, 106–118.

[16] ISO/IEC 25010:2011. Systems and Software Engineering – Systems and Software Quality Requirements and Evaluation (SQuaRE) – Quality Model. International Organization for Standardization.

[17] Moe, N. B., Šmite, D., & Ågerfalk, P. J. (2020). "Understanding Customer Collaboration in Agile Software Development: A Multiple-Case Study." Empirical Software Engineering, 25(3), 1581–1605.

[18] Sommerville, I. (2019). Ingeniería de Software. Pearson.

[19] Pressman, R. S., & Maxim, B. R. (2020). Software Engineering: A Practitioner's Approach. McGraw-Hill.

[20] Rivera, J., García, S., & Martínez, R. (2022). Hybrid Scrum-Kanban models for predictable software delivery in resource-constrained environments. *IEEE Transactions on Software Engineering*, 48(5), 1547-1562.

[21] Pressman, R. S., & Maxim, B. R. (2021). Software Engineering: A Practitioner's Approach (9th ed.). McGraw-Hill.

ANEXOS

Anexo A. Formato de cuestionario.

Formato del cuestionario

Cuestionario de Encuesta (10 preguntas – Escala Likert)

Título del formulario:

Encuesta sobre el uso de la metodología ágil Scrum en la gestión de proyectos tecnológicos

Instrucciones:

La siguiente encuesta tiene como objetivo conocer la percepción de los colaboradores sobre la aplicación de la metodología ágil Scrum en el desarrollo de proyectos tecnológicos dentro de la empresa. Marque la opción que considere más adecuada según su experiencia.

Escala de respuesta:

Valor	Significado
1	Totalmente en desacuerdo
2	En desacuerdo
3	Ni de acuerdo ni en desacuerdo
4	De acuerdo
5	Totalmente de acuerdo

Pregunta	1	2	3	4	5
1. La metodología Scrum facilita la organización de las tareas dentro del equipo de trabajo.	○	○	○	○	○
2. La planificación de los sprints permite distribuir adecuadamente las actividades del proyecto.	○	○	○	○	○
3. Las reuniones de seguimiento (Daily Scrum) mejoran la comunicación entre los miembros del equipo.	○	○	○	○	○
4. El uso de Scrum permite identificar problemas en el proyecto de manera oportuna.	○	○	○	○	○
5. La metodología Scrum facilita la adaptación a cambios en los requerimientos del proyecto.	○	○	○	○	○

6. El uso de Scrum contribuye a mejorar la productividad del equipo de trabajo.	○	○	○	○	○
7. La metodología Scrum permite entregar resultados de forma más rápida y eficiente.	○	○	○	○	○
8. El uso de Scrum mejora la coordinación entre los miembros del equipo.	○	○	○	○	○
9. La aplicación de Scrum contribuye a mejorar la calidad de los proyectos desarrollados.	○	○	○	○	○
10. Considero que la metodología Scrum es adecuada para la gestión de proyectos tecnológicos en la empresa.	○	○	○	○	○

Anexo B. Guía de entrevista con preguntas abiertas.

Guía de entrevista con preguntas abiertas.

Título:

Entrevista dirigida al presidente ejecutivo de la empresa Blacklabeltech Black Label Technology Cía. Ltda.

Objetivo:

Recopilar información sobre la aplicación de la metodología ágil Scrum en la gestión de proyectos tecnológicos dentro de la empresa.

Datos del entrevistado

Cargo: Presidente Ejecutivo

Área: Departamento de TI

Fecha: 25 de marzo de 2026

Preguntas de la entrevista

¿Cómo se implementó la metodología ágil Scrum dentro de la empresa Black Label Technology Cía. Ltda.?

La metodología Scrum se implementó como una iniciativa para mejorar la organización de los proyectos de desarrollo de software, adoptando sus principales eventos como la planificación de sprints, reuniones diarias y revisiones periódicas. Sin embargo, su aplicación no ha sido completamente formal ni estructurada, ya que se ha adaptado de manera empírica según la experiencia del equipo, sin una guía definida que estandarice roles, artefactos y procesos dentro de la organización.

¿Cuáles han sido los principales beneficios que ha observado desde la implementación de Scrum en la gestión de proyectos?

Entre los principales beneficios se destaca una mejor organización de las tareas, mayor visibilidad del avance de los proyectos y una mejora en la comunicación entre los miembros del equipo. Además, Scrum ha permitido trabajar en ciclos más cortos, facilitando la entrega parcial de funcionalidades y una mayor capacidad de adaptación frente a cambios en los requerimientos del cliente.

¿Qué dificultades o desafíos se han presentado al aplicar la metodología Scrum en el equipo de trabajo?

Uno de los principales desafíos ha sido la falta de una correcta implementación de la metodología, lo que ha generado inconsistencias en su uso. Existen dificultades en la definición clara de roles, poca disciplina en el cumplimiento de los eventos Scrum y una limitada gestión visual del flujo de trabajo. Asimismo, el equipo presenta inconvenientes para adaptarse rápidamente a cambios y para mantener una comunicación continua y estructurada, lo que impacta en los tiempos de entrega y en la organización general del proyecto.

Desde su perspectiva, ¿Cómo ha influido Scrum en la productividad y organización del equipo?

Scrum ha contribuido positivamente en la organización del equipo, permitiendo una mejor planificación de actividades y distribución del trabajo. Sin embargo, su impacto en la productividad ha sido parcial, debido a que no se ha implementado de manera integral, lo que ha limitado el aprovechamiento total de sus beneficios. A pesar de ello, ha servido como una base importante para introducir prácticas ágiles dentro de la empresa.

¿Qué mejoras considera necesarias para optimizar la aplicación de la metodología Scrum en la empresa?

Se considera necesario establecer una metodología más estructurada y adaptada a las necesidades específicas de la empresa, que combine las buenas prácticas de Scrum con herramientas de gestión visual y prácticas de calidad. Esto permitiría mejorar la

organización del flujo de trabajo, fortalecer la calidad del software, optimizar la comunicación del equipo y lograr una mayor eficiencia en el desarrollo de los proyectos, alineándose de mejor manera con los objetivos de la empresa.