ATMIRE

Title	Modelo de gestión de incidencias para una mesa de ayuda y gestión de requisitos para una fábrica de software en una empresa del sector de soluciones de beneficios y engagement para empleados
Item Type	http://purl.org/coar/resource_type/c_bdcc
Authors	Valencia Barreto, Crysthian Fabio;Gonzalez Escobar, Elio;Tito Alegre, Gladys Mercedes
Publisher	Universidad Peruana de Ciencias Aplicadas (UPC)
Rights	http://purl.org/coar/access_right/c_abf2
Download date	2026-11-27 20:56:27
Item License	https://creativecommons.org/licenses/by/4.0/
Link to Item	https://hdl.handle.net/10757/691664

UNIVERSIDAD PERUANA DE CIENCIAS APLICADAS

ESCUELA DE POSTGRADO

PROGRAMA DE MAESTRÍA EN DIRECCIÓN DE SISTEMAS Y TECNOLOGÍAS DE LA INFORMACIÓN

Modelo de gestión de incidencias para una mesa de ayuda y gestión de requisitos para una fábrica de software en una empresa del sector de soluciones de beneficios y engagement para empleados

TRABAJO DE INVESTIGACIÓN

Presentado como parte de los requisitos para optar el grado académico de Maestro en Dirección de Sistemas y Tecnologías de la Información

AUTOR(ES)

Valencia Barreto, Crysthian Fabio	0009-0005-4065-2331
Gonzalez Escobar, Elio	0009-0009-9613-9843
Tito Alegre, Gladys Mercedes	0009-0001-7675-7187

ASESOR(ES)

BARRIENTOS PADILLA, Alfredo	0000-0002-0029-4913

Lima, 09 de julio de 2026

Dedicatoria

A mi familia y, en especial, a mis hijos, Matías y Natalia, que son mi mayor inspiración. Gracias por su amor, paciencia y comprensión y por recordarme cada día la importancia de luchar con perseverancia y corazón A Dios, por darme la fortaleza, serenidad y la claridad para llegar hasta este momento tan importante en mi vida.

Gladys T.

A Dios, Todopoderoso quien con su inmensa voluntad ha permitido darme las fuerzas y voluntad de emprender con esfuerzo el viaje de lograr ser magister

Elio G.

A Dios, por estar siempre presente. A mi hijo Eduardo, por ser mi mayor fuente de amor, fortaleza y motivación. A su mamá Elena, por su valioso apoyo durante la maestría. A mi Madre, por enseñarme el valor del esfuerzo, perseverancia y como ser un buen padre.

Crysthian V.

Agradecimientos

En primer lugar, agradecer a Dios por brindarnos su fuerza y guía en cada momento de nuestro camino.

Gracias a nuestras familias por su amor, aliento, comprensión y apoyo constante hacia nosotros durante este arduo proceso.

Gracias a nuestros profesores cuya orientación y apoyo fueron el pilar académico fundamental para llevar a cabo nuestra investigación

Gracias a nuestro asesor por su orientación experta y consejos prácticos fueron esenciales para la culminación de este trabajo.

Resumen

Los servicios de TI en una empresa del sector de soluciones de beneficios y engagement requieren niveles de servicio óptimos para garantizar eficiencia que son esenciales para la operación y el soporte a sus áreas. La creciente digitalización de servicios en dicho sector exige altos estándares de rendimiento, disponibilidad y escalabilidad. Para garantizar la calidad del servicio, es fundamental implementar medidas que permitan evaluar los niveles de servicio y detectar oportunidades de mejora.

Nuestro proyecto de tesis propone optimizar la gestión de su mesa de ayuda mejorando la gestión de incidencias, así como también la gestión de requisitos de software optimizando el proceso de desarrollo de su fábrica de software integrando ambos mundos bajo un mismo modelo adaptado al entorno de la empresa.

Nuestro modelo se apoya en el catálogo de servicios y organiza la atención en dos frentes:

(i) Mesa de Ayuda para incidencias y solicitudes de servicios aplicando prácticas de ITIL 4 de Gestión de Incidentes, Gestión de Solicitudes de Servicio, Gestión del Conocimiento y Gestión de Cambios asegurando tiempos de respuesta / resolución conforme a SLA.

(ii) Fábrica de Software responsable de requisitos (nuevas funcionalidades y mejoras) gestionados bajo Scrum.

De este modo, incidencias y solicitudes se atienden en la oferta de Mesa de Ayuda y los requisitos en la oferta de Fábrica de Software, ambas registradas en el catálogo de servicios, coordinadas por el mismo modelo y logrando optimizar la gestión de TI al reducir tiempos de respuesta, garantizar servicios eficientes y alinearlos a objetivos estratégicos.

Palabras clave: ITIL 4; Scrum; Gestión de Incidentes; Gestión de Requisitos de Software.

Objetivo de Desarrollo Sostenible (ODS) relacionado:

Objetivo de Desarrollo Sostenible (ODS)	Marque
ODS 1: Fin de la pobreza||GDS 1: No poverty	
ODS 2: Hambre cero||GDS 2: Zero hunger	
ODS 3: Salud y bienestar||GDS 3: Good health and well-being	
ODS 4: Educación de calidad||GDS 4: Quality education	
ODS 5: Igualdad de género||GDS 5: Gender equality	
ODS 6: Agua limpia y saneamiento||GDS 6: Clean water and sanitation	
ODS 7: Energía asequible y no contaminante||GDS 7: Affordable and clean energy	
ODS 8: Trabajo decente y crecimiento económico||GDS 8: Decent work and economic growth	X
ODS 9: Industria, innovación e infraestructuras||GDS 9: Industry innovation and infrastructure	X
ODS 10: Reducción de las desigualdades||GDS 10: Reduced inequalities	
ODS 11: Ciudades y comunidades sostenibles||GDS 11: Sustainable cities and communities	
ODS 12: Producción y consumo responsables||GDS 12: Responsible consumption and production	X
ODS 13: Acción por el clima||GDS 13: Climate action	
ODS 14: Vida submarina||GDS 14: Life below water	
ODS 15: Vida de ecosistemas terrestres||GDS 15: Life and land	
ODS 16: Paz, justicia e instituciones sólidas||GDS 16: Peace, justice and strong institutions	
ODS 17: Alianzas para lograr los objetivos||GDS 17: Partnerships for the goals	

Integrated Model for Service Desk Incident Management and Software Factory Requirements Management in a Company in the Employee Benefits and Engagement Solutions Sector

Abstract

IT services in a company within the benefits and employee-engagement solutions sector require optimal service levels to ensure operational efficiency, as they are essential to business operations and support for internal areas. The sector's increasing digitalization demands high standards of performance, availability, and scalability. To guarantee service quality, it is critical to implement rigorous measures that assess service levels and identify opportunities for improvement.

This thesis proposes optimizing the Service Desk by strengthening incident management and improving software requirements management, streamlining the development process of the Software Factory and integrating both domains under a single model adapted to the company context.

Our model is based on the service catalog and organizes work along two fronts:

(i) Service Desk, for incidents and service requests, applying ITIL 4 practices of Incident Management, Service Request Management, Knowledge Management, and Change Enablement, ensuring response/resolution times in accordance with SLAs.

(ii) Software Factory, responsible for requirements (new features and enhancements) managed under Scrum.

Accordingly, incidents and requests are handled within the Service Desk offering, and requirements within the Software Factory offering, both registered in the service catalog and coordinated by the same model, thereby optimizing IT management by reducing response times, ensuring efficient services, and aligning operations and software evolution with strategic objectives.

Keywords: ITIL 4; Scrum; Incident Management; Software Requirements Management.

e202310839_ Valencia Barreto, Crysthian Fabio_ Modelo de gestión de incidencias para una mesa de ayuda y gestión de requisitos para una fábrica de software en una empresa del sector de soluciones de b

INFORME DE ORIGINALIDAD

7%	7%	4%	3%
INDICE DE SIMILITUD	FUENTES DE INTERNET	PUBLICACIONES	TRABAJOS DEL ESTUDIANTE

FUENTES PRIMARIAS

1	repositorioacademico.upc.edu.pe Fuente de Internet	1%
2	hdl.handle.net Fuente de Internet	1%
3	upc.aws.openrepository.com Fuente de Internet	1%
4	Submitted to Universidad TecMilenio Trabajo del estudiante	<1%
5	www.campustudionline.com Fuente de Internet	<1%
6	repository.ucc.edu.co Fuente de Internet	<1%
7	Submitted to Universidad Tecnológica Centroamericana UNITEC Trabajo del estudiante	<1%
8	Lanuza Bustamante, Narda Naomi. "Gestión de incidencias basadas en ITIL 4.0 para reducir tiempos en la dirección de tecnologías de información de una Universidad Privada", Universidad Nacional del Altiplano de Puno (Peru) Publicación	<1%
9	Submitted to Universitat Oberta de Catalunya Trabajo del estudiante	<1%
10	pdfcoffee.com Fuente de Internet	<1%
11	www.eceasst.org Fuente de Internet	<1%
12	Herrera Yucra, Benazir Francis. "Modelo de gestión de proyectos para desarrollo de software en la Universidad Peruana Unión - Filial Juliaca", Universidad Nacional del Altiplano de Puno (Peru) Publicación	<1%

www.coursehero.com

Tabla de contenido

1. Definición del Problema 1

1.1 Antecedentes 1

1.1.1 Caso de Negocio 2

1.2 Justificación del Estudio 4

1.2.1 Motivación 4

1.2.2 Indicadores de Negocio 4

1.2.3 Definición del Problema 7

1.2.4 Pregunta del Problema 8

2. Marco Teórico y Estado del Arte 9

2.1 Marco Teórico 9

2.1.1 Evolución y fundamentos de ITIL v4 10

2.1.2 Estructura y clasificación de las prácticas de ITIL v4 12

2.1.3 La cadena de valor del servicio (Service Value Chain) 21

2.1.4 Flujos de valor (Value streams) y creación del valor 22

2.1.5 Métricas e indicadores de desempeño 24

2.1.6 Mejora continua y cultura organizacional del servicio 26

2.1.7 Mesa de Ayuda: Función, Estructura y Gestión del Soporte Técnico 29

2.1.8 Integración ITIL v4 y Scrum: Sinergia para la Agilidad y Estabilidad del Servicio 31

2.1.9 Conclusiones del Marco Teórico 33

2.2 Estado del Arte 35

2.3 Bases Legales 48

2.3.1 Ley N° 28051 - Ley de Prestaciones Alimentarias en Beneficio de los Trabajadores Sujetos al Régimen Laboral de la Actividad Privada 48

2.3.2 Ley N° 29733 - Ley de Protección de Datos Personales 49

2.3.3 Ley N° 30096 - Ley de Delitos Informáticos 49

2.4 Marco Metodológico 51

2.4.1 ITIL – Gestión de Servicios de TI 51

2.4.2 Practicas Relevantes de ITIL v4 en la Gestión de Servicios 53

2.4.3 Ceremonias y Prácticas Operativas en ITIL v4 para la Mesa de Ayuda 55
2.4.4 Scrum (marco ágil) 56
2.4.5 ISO/IEC/IEEE 15288 (Procesos del ciclo de vida de sistemas) 61
2.4.6 ISO/IEC 12207 (Procesos del ciclo de vida del software). 62
2.4.7 ISO/IEC 29110 (VSE – Very Small Entities). 63
2.4.8 ISO/IEC 25010 (Calidad del producto software). 64
2.4.9 Gestión de Proyectos – PMBOK (PMI). 64
2.4.10 Organizational Project Management Maturity Model - OPM3 (PMI). 65

3. Situación Actual 66

3.1 Misión 66
3.2 Visión 66
3.3 Objetivos Estratégicos 67
3.4 Organigrama 69
3.5 Cadena de Valor 72
3.5.1 Actividades Primarias (Generación Directa de Valor) 72
3.5.2 Actividades de Soporte (Infraestructura y Crecimiento) 73
3.6 Mapa de Procesos 75
3.6.1 Macroproceso General (Gestión del Pedido de Tarjetas de Beneficiarios) . 75
3.6.2 Macroproceso General vs Objetivos Estratégicos 79
3.6.3 Mapa de Procesos Específicos: IT - Plataformas de Pago e Infraestructura y Ciberseguridad 81
3.6.4 Procesos Específicos vs Objetivos 85
3.7 AS-IS del Proceso de Gestión de Incidencias (Mesa de Ayuda) y Gestión de Requisitos (Fábrica de Software) 86
3.7.1 Descripción del Proceso de Atención de Incidencias (Mesa de Ayuda) 87
3.7.2 Descripción del Proceso de Atención de Requisitos (Fábrica de Software) 89
3.8 Puntos de Dolor del Proceso Atención de Incidencias (Mesa de Ayuda) y Atención de Requisitos (Fábrica de Software) 91
3.8.1 Métricas e Indicadores Detectados al Proceso Gestión de Incidencias (Mesa de Ayuda) y Gestión de Requisitos (Fábrica de Software) 94
3.9 Objetivos 109
3.9.1 Objetivo General 109

3.9.2 Objetivos Secundarios 109
3.9.3 Indicadores de logro de cada objetivo 110

4. Desarrollo de la Solución 111

4.1 Gestión del Proyecto 111
4.1.1 Plan de Gestión del Alcance 111
4.1.2 Alcance del Proyecto 112
4.1.3 Plan de Gestión de los Riesgos 112
4.1.4 Riesgos 113
4.1.5 Gestión del Cronograma 118
4.1.6 Cronograma 119
4.1.7 Hitos 119
4.1.8 Roles y Responsabilidades 120
4.1.9 Limitaciones 120
4.2 Desarrollo de la Solución 121
4.2.1 Modelo de gestión de la mesa de ayuda (Incidentes N1 - N2) 121
4.2.2 Modelo de gestión de requisitos de software (Mejora Continua y Bugs) . 143
4.2.3 Modelo de gestión de requisitos de software (Nuevos Productos) 152
4.3 Gestión del Cambio 163
4.3.1 Ámbito de Aplicación del Cambio 163
4.3.2 Tipo de Cambio Implicado 164
4.3.3 Factores Críticos de Éxito 165
4.3.4 Riesgos del Cambio 165

5. Evaluación Costo Beneficio 166

5.1 Estructura de Costos 166
5.2 Beneficios financieros 169
5.3 Rentabilidad y factibilidad del proyecto 171

Conclusiones 174

Recomendaciones 175

Referencias 176

Lista de Tablas

Tabla 1 CSAT% (Anual) Mesa de Ayuda 4
Tabla 2 CSAT% (Anual) Fábrica de Software 6
Tabla 3 ISO/IEC 25010 64
Tabla 4 Objetivos Estratégicos Kred C&S 67
Tabla 5 Macroproceso General vs Objetivos Estratégicos 79
Tabla 6 Procesos Específicos vs Objetivos Estratégicos 85
Tabla 7 Descripción del Proceso de Atención de Incidencias (Mesa de Ayuda) 87
Tabla 8 Descripción del Proceso de Atención de Requisitos (Fábrica de Software) 89
Tabla 9 Puntos de dolor 91
Tabla 10 Puntos de dolor vs Roles 93
Tabla 11 Prioridad – SLA (Mesa de Ayuda) 94
Tabla 12 Cumplimiento de Tickets por SLA (Mesa de ayuda) 95
Tabla 13 % Cumplimiento de SLA Mesa de Ayuda (Mensual y Prioridad) 95
Tabla 14 Total de Tickets Reabiertos por Prioridad 96
Tabla 15 % de Reapertura (Calidad del Cierre) 96
Tabla 16 Resultado Encuesta Recomendación - Mesa de Ayuda 98
Tabla 17 Resultado NPS - Mesa de Ayuda 98
Tabla 18 Resultados Encuesta Satisfacción - Mesa de Ayuda 99
Tabla 19 CSAT (Mensual) - Mesa de Ayuda 99
Tabla 20 Requisitos atendidos por Sprint 101
Tabla 21 % de Fallos Post-Implementación 103
Tabla 22 Resultado Encuesta Recomendación - Fábrica de Software 105
Tabla 23 Resultado NPS - Fábrica de Software 105
Tabla 24 Resultado Encuesta de Satisfacción - Fábrica de Software 106
Tabla 25 CSAT (Mensual) - Fábrica de Software 107
Tabla 26 Calificación del Riesgo 113
Tabla 27 Riesgos 114
Tabla 28 Hitos 119
Tabla 29 Roles y Responsabilidades 120
Tabla 30 Descripción del modelo de gestión de la mesa de ayuda N1 – N2 125
Tabla 31 Ceremonias Mesa de Ayuda N1 – N2 127
Tabla 32 Oferta de Servicio 129

Tabla 33 Catálogo de Servicios 129
Tabla 34 Ticket de Atención - Campos 131
Tabla 35 Roles - Mesa de Ayuda 132
Tabla 36 Roles - Fábrica de Software 132
Tabla 37 Mesa de Ayuda - Matriz RACI 133
Tabla 38 Personal por Área 134
Tabla 39 Matriz Impacto por Urgencia -> Prioridad (N1) 139
Tabla 40 Niveles de Servicio 140
Tabla 41 Mesa de Ayuda – Métricas e Indicadores Propuestos 141
Tabla 42 Descripción del modelo de gestión de requisitos de software (Mejora Continua y Bugs) 146
Tabla 43 Ceremonias Desarrollo de Software Ágil (Mejora Continua) 148
Tabla 44 Mejora Continua - Roles y Responsabilidades 149
Tabla 45 Mejora Continua – Matriz RACI 150
Tabla 46 Mejora Continua – Métricas e Indicadores Propuestos 151
Tabla 47 Descripción del modelo de gestión de requisitos de software (Nuevos Productos) 156
Tabla 48 Ceremonias Desarrollo de Software Ágil (Nuevo Producto) 159
Tabla 49 Modelo Nuevo Producto – Roles 160
Tabla 50 Modelo Nuevo Producto – Matriz RACI 160
Tabla 51 Modelo Nuevo Producto – Métricas e Indicadores Propuestos 162
Tabla 52 Tipos de Cambio 164
Tabla 53 Riesgos del Cambio 165
Tabla 54 Estructura de Costos 167
Tabla 55 Costos Licencias 168
Tabla 56 Costos Capacitación 168
Tabla 57 Costos operativos del proyecto 169
Tabla 58 Costo / Beneficio Mesa de ayuda 170
Tabla 59 Costo / Beneficio– Fábrica de Software 170
Tabla 60 Retorno de Inversión 171
Tabla 61 Flujo de Caja CAPEX / OPEX 173

Lista de Figuras

Figura 1 CSAT% (Anual) Mesa de Ayuda – Gráfico 5
Figura 2 CSAT% (Anual) Fábrica de Software – Gráfico 6
Figura 3 Línea de tiempo de la evolución de ITIL (1980–2019) 12
Figura 4 Diagrama de flujo del proceso de gestión de servicios del Help Desk de TI 31
Figura 5 Integración del Sistema de Valor del Servicio ITIL 4 con metodología Scrum en la actividad de Diseño y Transición 33
Figura 6 Las 34 prácticas de ITIL 4 en relación con las actividades de la Cadena de Valor de Servicios 51
Figura 7 Marco de trabajo Scrum: roles, eventos y artefactos del ciclo de Sprint 56
Figura 8 Métrica Aplicadas en ITIL y Desarrollo Ágil 60
Figura 9 Clasificación de los procesos del ciclo de vida del sistema según ISO/IEC/IEEE 15288: Procesos Empresariales 61
Figura 10 Procesos específicos de software según ISO/IEC 12207: implementación, soporte y reutilización de software 62
Figura 11 Arquitectura documental de la serie ISO/IEC 29110: perfiles, evaluación y certificación, y guías para Pequeñas Organizaciones (VSE) 63
Figura 12 Organigrama Kred C&S 69
Figura 13 Cadena de Valor 72
Figura 14 Mapa de Procesos – Macroproceso General 75
Figura 15 Mapa de Procesos – Procesos Específicos de TI 81
Figura 16 BPMN (As-Is) Fábrica de Software 86
Figura 17 Velocidad de Entrega de RQs vs Objetivo por Sprint 101
Figura 18 Total de Fallos Post-Implantación por Sprint 103
Figura 19 Cronograma 119
Figura 20 Modelo Adaptativo Mesa de Ayuda N1 – N2 123
Figura 21 Modelo de gestión de la mesa de ayuda (Incidentes N1 - N2) 124
Figura 22 Formula Capacidad del Servicio - Mesa de Ayuda 135
Figura 23 Resultado Capacidad del Servicio - Mesa de Ayuda 136
Figura 24 Formula Capacidad Efectiva del Equipo de Fabrica de Software 137
Figura 25 Formula Horas de QA Requeridas por Sprint 138
Figura 26 Formula Recursos QA Mínimos para Atención del Sprint 138
Figura 27 Modelo Adaptativo Desarrollo de Software Ágil (Mejora Continua) 144

Figura 28 Modelo de gestión de requisitos de software (Mejora Continua y Bugs)..... 145
Figura 29 Flujo de Gestión de Requisitos (Nuevo Producto) 152
Figura 30 Modelo Adaptativo Desarrollo de Software Ágil (Nuevo Producto) 154
Figura 31 Modelo de gestión de requisitos de software (Nuevos Productos) 155

1. Definición del Problema

1.1 Antecedentes

Kred Consulting & Services (Kred C&S) es una empresa que pertenece al sector de servicios especializada en beneficios y engagement para empleados, es líder en Perú en brindar soluciones de tarjetas para alimentación, incentivos y movilidad, colaborando estrechamente con las áreas de recursos humanos de empresas privadas y públicas para mejorar el bienestar y la calidad de vida de los colaboradores.

Su sede principal está ubicada en Francia con presencia en 31 países alrededor del mundo.

El sector de servicios de beneficios y engagement para empleados se enfoca en ofrecer soluciones de gestión de gastos, optimizando sus costos laborales y brindando beneficios para empresas y empleados. Las compañías que brindan estos servicios se especializan en soluciones como tarjetas de alimentación, movilidad, regalos, combustible y de otros tipos que permiten mejorar la experiencia laboral tanto para empleados como para empleadores.

En función a lo anterior, Kred C&S ofrece soluciones sostenibles y personalizadas buscando innovar en la experiencia de pago en el mercado local. Los servicios que ofrecen actualmente:

- Tarjeta de Alimentación: Una tarjeta de alimentación que permite a las empresas motivar a sus colaboradores y ahorrar en sobrecostos de planilla.
- Tarjeta de Navidad: Tarjetas de regalo aceptadas en más de 100,000 comercios físicos y en línea, ideales para reconocer y celebrar con los empleados durante las festividades de fin de año.
- Tarjeta de Incentivo: Soluciones personalizadas para mejorar la productividad del equipo mediante planes de incentivos adaptados a las necesidades específicas de cada empresa.
- Tarjeta de Regalo: Tarjetas de regalo que se ajustan a las preferencias y necesidades de los colaboradores, ofreciendo flexibilidad y reconocimiento.
- Tarjeta de Gasolina: Una solución para cubrir los gastos de combustible de la empresa y sus colaboradores, facilitando la gestión y control de estos gastos.
- Tarjeta de Escolaridad: Beneficios diseñados pensando en la familia de los colaboradores, apoyando en gastos relacionados con la educación y promoviendo su bienestar.

En noviembre de 2024, Kred C&S se convirtió en la primera empresa de beneficios e incentivos en Perú en habi litar Google Pay asociados a sus tarjetas, permitiendo a sus usuarios realizar pagos digitales de manera más sencilla y segura desde un dispositivo móvil y en marzo se culminó con la habilitación en dispositivos Apple por medio de su servicio Apple Pay.

1.1.1 Caso de Negocio

La gerencia de Sistemas de Kred C&S está conformada por el área de "IT - Plataformas de Pago" y el área de "Infraestructura – Ciberseguridad". Estas áreas gestionan una fábrica de software y mesa de ayuda respectivamente.

Actualmente, estos servicios operan sin un marco metodológico definido y sus canales de comunicación no son los óptimos, lo cual afecta la eficiencia y calidad de estos que son prestados a las diversas áreas de la empresa.

- Mesa de ayuda: Operada por personal interno de la empresa, encargada de recibir, clasificar, gestionar y atender tickets de incidencias y derivar requisitos de software solicitados por usuarios internos.
- Fábrica de software: Operado por personal externo de la empresa bajo la modalidad Manpower, responsable del desarrollo y mantenimiento de las aplicaciones core de Kred C&S en base a los requisitos de software que le son derivados por mesa de ayuda.

Actualmente, la empresa enfrenta alta demanda en la gestión de incidencias y la gestión requisitos de software, originado principalmente por la falta de un marco metodológico de gestión para ambas áreas, deficiente base de conocimiento del negocio y técnico de la mesa de ayuda, cambios en tecnologías solicitado por la casa matriz a sus filiales y procesos de negocio con errores en su automatización lo cual impacta en la calidad del servicio que se brinda a las áreas.

En coherencia con lo anterior, la magnitud de la demanda analizada en la presente tesis corresponde a lo siguiente:

- Mesa de Ayuda: 2,378 incidentes como evidencia cuantitativa del periodo ago-2024 a ene-2025 obtenidos de Jira Service Management, 396 tickets en promedio por mes. El desglose y análisis figuran en el Capítulo 3, secciones 3.8 y 3.8.1.
- Fábrica de software: 103 requisitos como evidencia cuantitativa desde el sprint FY25-Q1-SP5 al FY25-Q2-SP5 (28/oct/24 al 7/feb/25) obtenidos de Azure DevOps, 12.9 historias por sprint y 16 release ejecutados a producción (2 por sprint). El desglose y análisis figuran en el Capítulo 3, secciones 3.8 y 3.8.1.

1.2 Justificación del Estudio

1.2.1 Motivación

La ausencia de un marco metodológico para la gestión de incidencias y la gestión de requisitos conlleva una serie de problemas que se ve reflejado en la percepción de los usuarios con respecto a cómo la mesa de ayuda y la fábrica de software los atienden.

Por tal razón, nuestra motivación principal es optimizar la gestión de estas áreas con el fin de garantizar un soporte eficiente, efectivo y alineado con los objetivos estratégicos de Kred C&S, a su vez que mejoramos la calidad del servicio.

1.2.2 Indicadores de Negocio

El análisis de indicadores de negocio evidencia una brecha significativa entre el desempeño actual y los niveles óptimos de eficiencia, calidad de las soluciones entregadas y la satisfacción de los usuarios internos.

Con respecto a la mesa de ayuda, se cuenta con el siguiente indicador de negocio:

- **CSAT% (Anual) - Servicio de Mesa de Ayuda**: En la empresa, se realiza esta evaluación una vez al año sobre una muestra significativa de colaboradores. El objetivo de este indicador es obtener el nivel de satisfacción (CSAT) con respecto a la atención recibida por el servicio de mesa de ayuda.

  Pregunta: "¿cómo calificarías tu satisfacción general con el servicio de mesa de ayuda recibida en el último año?"

  Escala Likert: 1 – 5

  CSAT% = ((Muy Satisfecho + Satisfecho) / Total) * 100

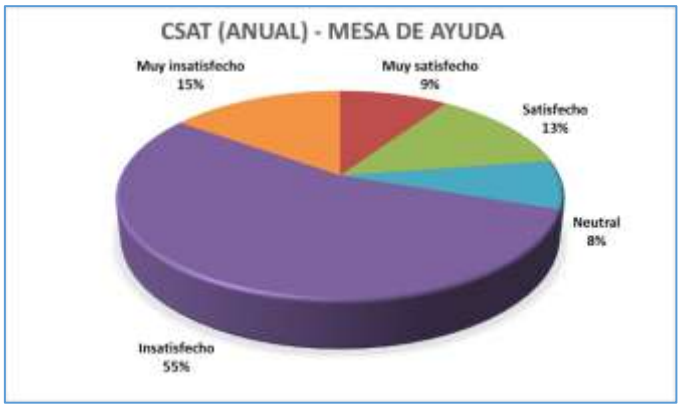
  Muestra: 53 trabajadores

  Periodo: Ago/2024 – Jul/2025

Tabla 1
CSAT% (Anual) Mesa de Ayuda

Grado de Satisfacción	Frecuencia	Porcentaje
Muy satisfecho	5	9%
Satisfecho	7	13%
Neutral	4	8%
Insatisfecho	29	55%
Muy insatisfecho	8	15%

Figura 1
CSAT% (Anual) Mesa de Ayuda – Gráfico



A partir de los resultados, se pueden extraer las siguientes conclusiones:

- Puntuación CSAT - Mesa de Ayuda:
  - Solo un 9% de los colaboradores se sienten muy satisfechos y un 13% satisfechos alcanzando un índice CSAT de 22%.
- Baja satisfacción general:
  - El 55% de los usuarios están insatisfechos y un 15% muy insatisfechos, sumando un total del 70% de usuarios con experiencias negativas, mientras un 8% opinan de manera neutral.

Conclusiones del indicador:

- La mesa de ayuda no está cumpliendo con las expectativas de los usuarios en términos de atención, tiempos de respuesta o resolución de problemas.
- El nivel de satisfacción es demasiado bajo para un servicio que debe garantizar eficiencia y soporte de manera oportuna y continua.

Con respecto a la fábrica de software, se cuenta con el siguiente indicador de negocio.

- **CSAT% (Anual) – Servicio de Fábrica de Software**: Al igual que en el indicador relacionado a mesa de ayuda, también se realiza esta evaluación a la par una vez al año sobre una muestra significativa de colaboradores. El objetivo de este indicador es obtener el nivel de satisfacción (CSAT) con respecto a la atención recibida por el servicio de fábrica de software.

  Pregunta: "¿cómo calificarías tu satisfacción general con el servicio de fábrica de software recibido en el último año?"

  Escala Likert: 1 – 5

  CSAT% = ((Muy Satisfecho + Satisfecho) / Total) * 100

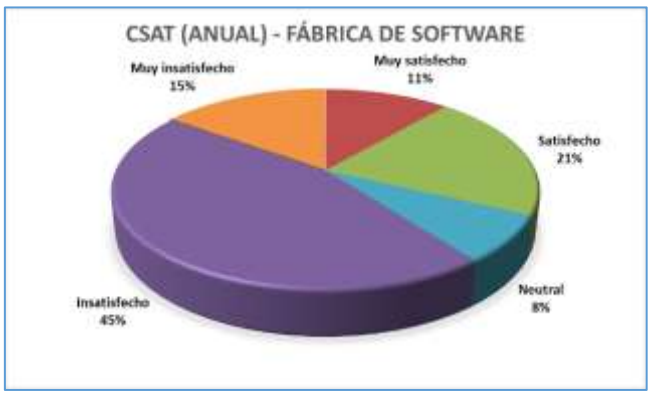
  Muestra: 53 trabajadores

  Periodo: Ago/2024 – Jul/2025

Tabla *2*
CSAT% (Anual) Fábrica de Software

Grado de Satisfacción	Frecuencia	Porcentaje
Muy satisfecho	6	11%
Satisfecho	11	21%
Neutral	4	8%
Insatisfecho	24	45%
Muy insatisfecho	8	15%

Figura 2
CSAT% (Anual) Fábrica de Software – Gráfico



A partir de los resultados, se pueden extraer las siguientes conclusiones:

- Puntuación CSAT – Fábrica de Software:
  - Solo un 11% de los colaboradores se sienten muy satisfechos y un 21% satisfechos alcanzando un índice CSAT de 32%.
- Baja satisfacción general:
  - El 45% de los colaboradores se sienten insatisfechos y el 15% muy insatisfechos, lo que suma un 60% de usuarios con experiencias negativas, mientras un 8% opinan de manera neutral.

Conclusiones del indicador:

- La fábrica de software no está cumpliendo con las expectativas en términos de calidad, tiempo de entrega o alineados con las necesidades del negocio.
- Si bien la satisfacción no es extremadamente baja, es necesario identificar oportunidades de mejora para aumentar la percepción positiva del servicio.

1.2.3 Definición del Problema

Con los resultados de las dos métricas de satisfacción obtenidas en el periodo analizado de seis meses (sección 3.8.1):

- CSAT% - Mesa de Ayuda = 22%
- CSAT% - Fábrica de Software = 32%

El resultado de CSAT se interpreta de forma integral para los dos servicios evaluados (Mesa de Ayuda y Fábrica de Software); por tanto, no se atribuye exclusivamente a la práctica de Gestión de Incidentes, sino al desempeño end-to-end de ambos servicios en el periodo analizado.

Por lo antes visto, esta carencia resulta entre otros factores en una gestión ineficaz de incidencias y la gestión de requisitos; tiempos de respuesta inadecuados, falta de estandarización en la prestación de servicios y una asignación desequilibrada de la carga laboral entre el personal de TI. Todo esto incrementa los tiempos de respuesta y dificulta la colaboración entre la mesa de ayuda y la fábrica de software. Como resultado, se generan interrupciones en los procesos de negocio y una mayor insatisfacción de los usuarios internos.

Importante: La medición mensual (CSAT y NPS) y los indicadores operativos se detallan en la sección 3.8.1 lo cual nos permite relacionar percepción con desempeño como parte de las conclusiones que ahí se indican.

1.2.4 Pregunta del Problema

¿Cómo podemos mejorar y optimizar los procesos de gestión de la mesa de ayuda y de la fábrica de software en Kred C&S?

2. Marco Teórico y Estado del Arte

2.1 Marco Teórico

En el contexto actual de las tecnologías de la información y alta demanda de eficiencia operativa, las organizaciones que proveen servicios de tecnología de la información enfrentan el reto de alinear sus procesos con las necesidades del negocio y de los usuarios finales. Esta necesidad se vuelve especialmente crítica en empresas que brindan soluciones de beneficios e incentivos para empleados, donde la calidad del servicio de TI y la agilidad en el desarrollo de software impactan directamente en la experiencia del cliente y en la propuesta de valor.

La presente investigación propone un modelo de mejora para la atención en la mesa de ayuda y en la fábrica de software de una empresa del rubro mencionado, fundamentado en la integración de dos marcos de trabajo reconocidos: ITIL v4 y Scrum. ITIL v4 es un marco de referencia para la gestión de servicios de TI que ofrece un enfoque basado en el valor, el servicio y la mejora continua (AXELOS, 2019). Por otro lado, Scrum es un marco ágil que permite abordar proyectos de desarrollo de software mediante ciclos iterativos, colaboración y autoorganización del equipo (Schwaber & Sutherland, 2020).

El objetivo de este marco teórico es establecer los fundamentos conceptuales que sustentan la propuesta del modelo, describiendo con rigor académico los principios y componentes de ITIL v4, las prácticas operativas más relevantes para el contexto de la mesa de ayuda, así como los elementos centrales de Scrum en entornos de desarrollo ágil. Asimismo, se explicará la manera en que ambos marcos pueden integrarse de forma sinérgica para optimizar los flujos de trabajo, reducir los tiempos de atención, mejorar la calidad del servicio y fomentar una cultura de mejora continua.

Este marco teórico servirá como base analítica y argumentativa para sustentar la viabilidad y pertinencia del modelo propuesto, considerando las mejores prácticas internacionales en la gestión de servicios de TI y desarrollo ágil de software. A lo largo de las siguientes secciones se desarrollarán con detalle estos conceptos, incluyendo gráficos explicativos y referencias académicas actualizadas que permitan contextualizar su aplicación práctica en el entorno organizacional objeto de estudio.

2.1.1 Evolución y fundamentos de ITIL v4

La Information Technology Infrastructure Library (ITIL) surgió en la década de 1980 como una iniciativa del gobierno del Reino Unido a través de la Central Computer and Telecommunications Agency (CCTA), con el propósito de mejorar la calidad y eficiencia en la prestación de servicios de tecnología de la información. Su desarrollo inicial se basó en recopilar las mejores prácticas operativas de organizaciones públicas y privadas, con el objetivo de establecer un marco estandarizado para la Gestión de Servicios de TI (IT Service Management, ITSM) (AXELOS, 2019).

Durante las décadas siguientes, ITIL evolucionó desde una guía operativa hacia un modelo integral de gestión del valor, adaptándose a las nuevas exigencias de los entornos digitales y de innovación tecnológica. ITIL v3 (publicado en 2007 y actualizado en 2011) estableció una estructura basada en el ciclo de vida del servicio, compuesto por cinco fases: Estrategia, Diseño, Transición, Operación y Mejora Continua. Sin embargo, la acelerada transformación digital, la adopción de metodologías ágiles y la automatización en los procesos de desarrollo de software hicieron necesaria una nueva versión orientada a la flexibilidad y a la colaboración continua (AXELOS, 2019).

En 2019, AXELOS presentó ITIL v4, que marcó un cambio sustancial en la filosofía de la gestión de servicios. Esta versión incorporó conceptos provenientes de Agile, DevOps y Lean IT, impulsando un enfoque más adaptable, dinámico y centrado en la co-creación de valor entre los proveedores de servicios y los consumidores. El propósito fundamental de ITIL v4 es permitir que las organizaciones adopten un modelo de gestión flexible que mantenga la estabilidad operativa sin sacrificar la agilidad y la innovación (AXELOS, 2020).

ITIL v4 se estructura sobre el Sistema de Valor del Servicio (Service Value System – SVS), que integra todos los componentes y actividades necesarios para facilitar la creación de valor. Este sistema promueve una visión integral, donde los servicios no son entidades aisladas, sino partes interconectadas que contribuyen colectivamente a los objetivos estratégicos de la organización.

Según AXELOS (2019), el SVS está compuesto por cinco elementos interrelacionados:

1. Principios guía (Guiding Principles): proporcionan recomendaciones universales aplicables a cualquier organización o contexto, tales como “enfocarse en el valor”,

"progresar iterativamente con retroalimentación" y "colaborar y promover la visibilidad".

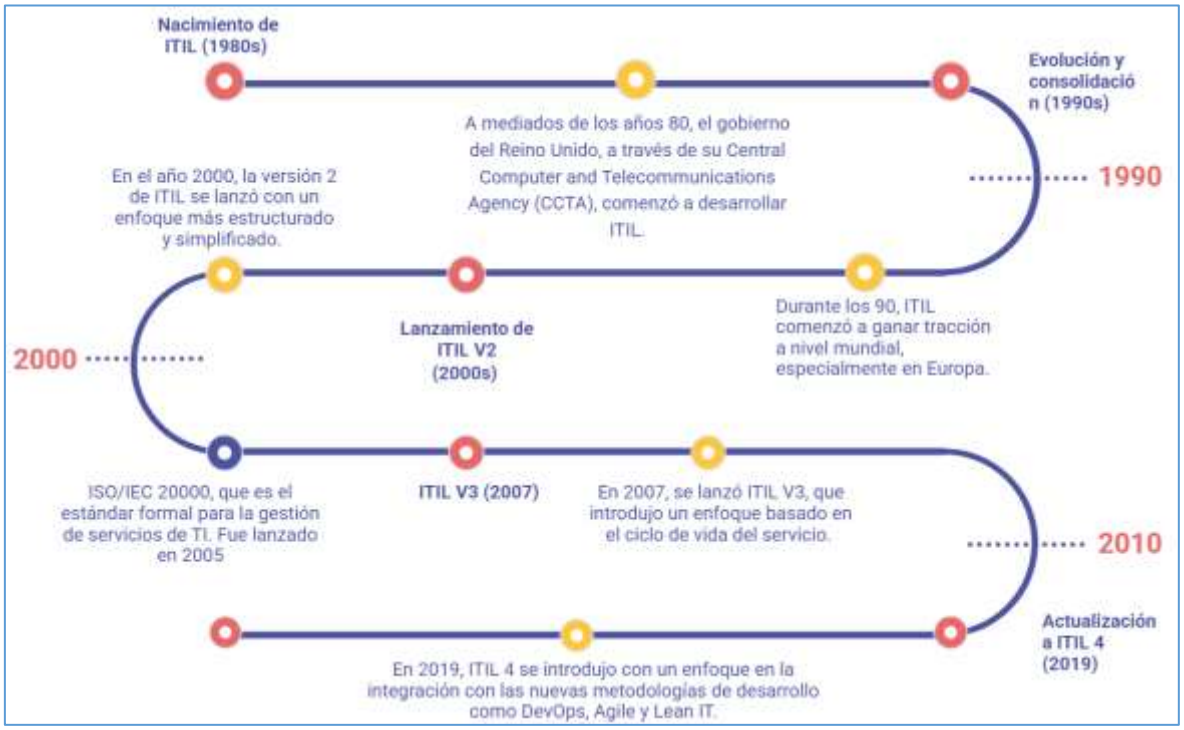
2. Gobernanza (Governance): define los mecanismos mediante los cuales la organización dirige y controla sus actividades, asegurando que las decisiones de TI estén alineadas con los objetivos corporativos.
3. Cadena de valor del servicio (Service Value Chain – SVC): describe las actividades centrales que transforman la demanda en valor, estructuradas en seis fases: *Plan, Improve, Engage*, *Design & Transition*, *Obtain/Build* y *Deliver & Support*.
4. Prácticas de gestión (Management Practices): conjunto de capacidades organizacionales que sustituyen el antiguo concepto de "procesos", promoviendo un enfoque más adaptable y transversal.
5. Mejora continua (Continual Improvement): asegura que los servicios evolucionen de manera constante y se ajusten a los cambios en las necesidades del negocio y del cliente.

El SVS, a diferencia del modelo lineal de versiones anteriores, promueve la integración de las diferentes prácticas de gestión bajo una estructura colaborativa y dinámica. Este marco favorece la alineación entre las áreas técnicas y las áreas de negocio, colocando al valor como el elemento central de toda la gestión de servicios.

Asimismo, ITIL v4 amplía su alcance hacia la gestión integral del valor organizacional, incorporando principios de sostenibilidad, automatización y experiencia del usuario. Tal enfoque se alinea con las tendencias contemporáneas de transformación digital, donde los servicios de TI no solo deben garantizar continuidad operativa, sino también generar ventaja competitiva a través de la innovación y la entrega continua de valor (AXELOS, 2021b).

Figura 3

Línea de tiempo de la evolución de ITIL (1980–2019)



Nota. Adaptado de Evolución de ITIL: Una profundización en la historia de ITIL, por ManageEngine ServiceDesk Plus, s.f.-b (https://www.manageengine.com/latam/service-desk/itsm/evolucion-de-itil.html).

2.1.2 Estructura y clasificación de las prácticas de ITIL v4

Una de las principales innovaciones de ITIL v4 es la sustitución del concepto de "procesos" utilizado en versiones anteriores por el de prácticas de gestión (Management Practices). Este cambio no es meramente terminológico: representa una evolución conceptual hacia una gestión más flexible, integrada y adaptable a los entornos empresariales actuales, donde la transformación digital y las metodologías ágiles exigen estructuras menos rígidas y más colaborativas (AXELOS, 2020).

ITIL v4 define un total de 34 prácticas de gestión, agrupadas en tres categorías principales, que abarcan tanto aspectos organizacionales como técnicos:

Prácticas de gestión general (General Management Practices): Se centran en las funciones transversales de dirección, planificación, control y soporte organizacional. Entre las más relevantes se encuentran la Gestión Financiera, Gestión de Estrategia, Gestión del Conocimiento, Gestión de Riesgos, Gestión de Seguridad de la Información, Gestión de la Medición y Reporte, Gestión de la Arquitectura y Gestión de la Fuerza Laboral y Talento:

Estas prácticas proporcionan el marco de gobernanza y alineamiento estratégico necesario para garantizar que los servicios de TI contribuyan a los objetivos empresariales y sean sostenibles a largo plazo.

Prácticas de gestión de servicios (Service Management Practices): Son las prácticas más directamente relacionadas con la entrega y soporte de servicios de TI. Comprenden disciplinas como Gestión de Incidentes, Gestión de Cambios, Gestión de Problemas, Gestión de Niveles de Servicio, Gestión de Requerimientos, Service Desk y Gestión de Continuidad del Servicio. Estas prácticas son las más aplicadas en el contexto de una mesa de ayuda (Service Desk) y una fábrica de software, tal como se desarrolla en esta tesis, ya que su implementación garantiza la estabilidad, la trazabilidad y la mejora continua del servicio.

Prácticas de gestión técnica (Technical Management Practices): Orientadas a la infraestructura y al desarrollo de soluciones tecnológicas, estas prácticas incluyen la Gestión de Despliegues, la Gestión de Infraestructura y Plataformas y la Gestión de Desarrollo de Software.
Su objetivo es asegurar que los componentes tecnológicos y las capacidades técnicas soporten de manera efectiva los requerimientos funcionales definidos en los procesos de gestión de servicios.

Según AXELOS (2020), esta división en tres grupos proporciona un marco coherente que permite alinear las operaciones técnicas con la estrategia del negocio, promoviendo una cultura de colaboración transversal entre las áreas de desarrollo, operaciones y gestión.

Asimismo, ITIL v4 fomenta la integración con otros marcos ágiles y colaborativos, destacando la compatibilidad con Scrum y DevOps. Esta compatibilidad es esencial para organizaciones que, como el caso estudiado en esta tesis, buscan un equilibrio entre control y velocidad en la entrega de valor.

En este contexto, Pressman y Maxim (2020) destacan que el valor de ITIL v4 reside en su capacidad de combinar la estabilidad operativa con la innovación continua, asegurando que los servicios de TI sean ágiles, confiables y medibles. Por ejemplo, en la mesa de ayuda, la Gestión de Incidentes se integra con los tableros Scrum para priorizar tickets y medir el cumplimiento de tiempos de respuesta, mientras que en la fábrica de software, la Gestión de Cambios se articula con los flujos de despliegue de DevOps para reducir riesgos en las entregas.

En suma, la clasificación de las prácticas de ITIL v4 ofrece un marco integral que une la planeación estratégica, la gestión operativa y el soporte técnico, permitiendo que las organizaciones gestionen de manera equilibrada la estabilidad y la innovación tecnológica.

Categoría ITIL 4	Práctica	Aplicación en la tesis (uso / integración)	Grado de relevancia
General Management	Gestión Financiera	Soporte a la evaluación de CAPEX y OPEX dentro del flujo de caja del modelo.	Alta
General Management	Gestión de Estrategia	Alineación del modelo ITIL–DevOps con los objetivos estratégicos de la empresa.	Alta
General Management	Gestión del Conocimiento	Base para la documentación de procedimientos y lecciones aprendidas.	Alta
General Management	Gestión de Riesgos	Identificación y mitigación de riesgos operativos y técnicos del servicio.	Media
General Management	Mejora Continua	Refuerzo de la cultura de aprendizaje y retroalimentación en la operación TI.	Alta
General Management	Gestión del Cambio Organizacional	Facilita la adopción del modelo ITIL–Scrum dentro del entorno corporativo.	Media
General Management	Medición y Reporte	Define los KPI y métricas de control del desempeño del servicio.	Alta

Categoría ITIL 4	Práctica	Aplicación en la tesis (uso / integración)	Grado de relevancia
General Management	Gestión de Seguridad de la Información	Controla la integridad, disponibilidad y confidencialidad de los datos.	▨ Media
General Management	Gestión de Proyectos	Planificación y control de iniciativas de implementación ITIL–DevOps.	▩ Alta
General Management	Gestión de Relaciones	Gestión de la comunicación entre la Mesa de Ayuda y las áreas de desarrollo.	▨ Media
General Management	Gestión del Proveedor	Supervisión de servicios externos y cumplimiento de SLA.	▨ Media
General Management	Gestión de Arquitectura	Define la estructura tecnológica base para la integración ITIL–DevOps.	▩ Alta
General Management	Workforce and Talent Management	Promueve la capacitación y asignación eficiente del personal TI.	☐ Baja
General Management	Portfolio Management	Garantiza la alineación de los proyectos TI con los objetivos estratégicos.	▩ Alta

Categoría ITIL v4	Práctica	Aplicación en la tesis (uso / integración)	Grado de relevancia
Service Management	Gestión de Incidentes	Práctica central del modelo para la Mesa de Ayuda.	▨ Alta
Service Management	Gestión de Problemas	Análisis de causas raíz y prevención de reincidencias.	▨ Media
Service Management	Gestión de Cambios	Control de versiones y despliegues dentro de la Fábrica de Software.	▨ Alta
Service Management	Gestión de Requerimientos	Priorización y trazabilidad de solicitudes de software.	▨ Alta
Service Management	Gestión de Disponibilidad	Garantiza la continuidad y disponibilidad de los servicios TI.	▨ Media
Service Management	Gestión de Niveles de Servicio	Controla métricas de SLA, CSAT y TTR.	▨ Alta
Service Management	Gestión de Capacidad	Analiza el rendimiento de recursos técnicos y humanos.	▨ Media

Categoría ITIL v4	Práctica	Aplicación en la tesis (uso / integración)	Grado de relevancia
Service Management	Gestión de Continuidad del Servicio	Sustentada en el plan DRP para garantizar resiliencia.	Media
Service Management	Gestión de Configuración y Activos	Control de inventarios y versiones de componentes.	Media
Service Management	Gestión de Disponibilidad del Cliente	Mide la experiencia y satisfacción del usuario.	Alta
Service Management	Gestión de Solicitudes de Servicio	Atención eficiente de peticiones estándar.	Media
Service Management	Gestión de Liberaciones	Coordina despliegues y entregas controladas de software.	Alta
Service Management	Gestión de Validación y Pruebas	Garantiza calidad antes del paso a producción.	Alta
Service Management	Gestión de Satisfacción del Usuario	Evalúa la percepción del cliente frente al servicio.	Alta

Categoría ITIL v4	Práctica	Aplicación en la tesis (uso / integración)	Grado de relevancia
Service Management	Service Desk	Núcleo operativo para recepción, registro y resolución de incidentes.	☒ Alta
Service Management	Gestión del Catálogo de Servicios	Documenta los servicios disponibles y sus condiciones de entrega.	☒ Media
Service Management	Gestión de Disponibilidad de Infraestructura	Monitoreo del rendimiento de infraestructura y plataformas.	☐ Baja

Categoría ITIL v4	Práctica	Aplicación en la tesis (uso / integración)	Grado de relevancia
Technical Management	Gestión de Despliegues	Automatiza el pipeline CI/CD de la Fábrica de Software.	▨ Alta
Technical Management	Gestión de Infraestructura y Plataformas	Proporciona soporte técnico estable y escalable.	▨ Alta
Technical Management	Gestión de Desarrollo de Software	Aplica los principios DevOps para entrega continua.	▨ Alta

2.1.3 La cadena de valor del servicio (Service Value Chain)

El Service Value Chain (SVC) constituye el núcleo operativo del Service Value System (SVS) y describe cómo las organizaciones convierten la demanda en valor mediante una secuencia de actividades interrelacionadas. Esta cadena de valor, según AXELOS (2019), proporciona un modelo flexible que puede adaptarse a las necesidades específicas de cada organización, facilitando la integración de diversas prácticas de gestión y promoviendo la colaboración entre las áreas de TI y negocio.

El SVC reemplaza la estructura lineal del ciclo de vida del servicio de ITIL v3, presentando un enfoque dinámico y adaptable. A diferencia de su predecesor, la cadena de valor no sigue una secuencia rígida, sino que permite iteraciones y combinaciones entre sus componentes, lo que se ajusta mejor a las metodologías ágiles y a los entornos DevOps, donde los servicios evolucionan de forma continua.

De acuerdo con AXELOS (2019), la cadena de valor está compuesta por seis actividades principales:

- Plan (Planificar): busca asegurar una comprensión común de la visión, el estado actual y los objetivos futuros de la organización, así como de la dirección que guiará las mejoras continuas.
- Improve (Mejorar): impulsa la optimización constante de los productos, servicios y prácticas, garantizando que la organización aprenda de la experiencia y mantenga un ciclo de mejora continua.
- Engage (Involucrar): promueve la comunicación y colaboración efectiva con todas las partes interesadas, asegurando que sus necesidades y expectativas se comprendan correctamente.
- Design & Transition (Diseñar y Transicionar): permite crear o modificar servicios de manera planificada y controlada, asegurando que cumplan los estándares de calidad y valor esperados.
- Obtain/Build (Obtener/Construir): se centra en adquirir o desarrollar los componentes, recursos y herramientas necesarios para ofrecer servicios que respondan a los requerimientos del negocio.
- Deliver & Support (Entregar y Soportar): garantiza la prestación eficiente de los servicios, incluyendo su mantenimiento y soporte operativo, de manera que se cumplan los acuerdos de niveles de servicio (SLA).

Cada una de estas actividades está interconectada y se apoya en las 34 prácticas de gestión definidas por ITIL v4. La combinación de actividades y prácticas conforma lo que se denomina un flujo de valor (Value Stream), el cual representa el conjunto de pasos necesarios para transformar una oportunidad o una demanda en un resultado que aporte valor.

Este enfoque sistémico tiene como objetivo que la organización optimice no solo sus procesos técnicos, sino también sus capacidades de gestión y colaboración. En entornos DevOps, por ejemplo, las actividades Obtain/Build, Deliver & Support y Improve se desarrollan en ciclos iterativos y automatizados, generando una entrega continua (Continuous Delivery) y una retroalimentación constante entre desarrollo y operaciones (Belalcázar, 2017).

Asimismo, la cadena de valor favorece la transparencia, trazabilidad y medición del rendimiento del servicio. Cada interacción dentro del SVC puede asociarse con indicadores de desempeño específicos, lo que permite evaluar la efectividad de las acciones y mantener un control integral de la entrega de valor.

Por ello, la Service Value Chain no solo es un modelo operativo, sino también una herramienta de gobernanza que alinea la ejecución técnica con la estrategia organizacional. En el contexto de esta tesis, este marco proporciona la base metodológica para la integración de ITIL v4 con Scrum y DevOps, permitiendo que la mesa de ayuda y la fábrica de software trabajen de forma coordinada en la resolución de incidencias, el desarrollo de requerimientos y la mejora continua de los servicios.

2.1.4 Flujos de valor (Value streams) y creación del valor

En el marco de ITIL v4, los flujos de valor (Value Streams) representan el conjunto de actividades que una organización ejecuta para transformar entradas —como demandas, oportunidades o incidentes— en resultados que generen valor para los usuarios y la empresa. Constituyen la materialización práctica de la Cadena de Valor del Servicio (Service Value Chain), ya que permiten visualizar cómo las diferentes prácticas de gestión interactúan y se encadenan para alcanzar un resultado concreto (AXELOS, 2019).

Los flujos de valor no son procesos aislados, sino combinaciones dinámicas y adaptables de actividades del SVC y de las prácticas de gestión que las soportan. Esto les otorga una gran

flexibilidad y les permite integrarse con enfoques modernos de gestión como Lean IT, Agile y DevOps, donde la mejora continua y la entrega iterativa son principios fundamentales.

De acuerdo con Kersten (2018), los flujos de valor permiten identificar cuellos de botella, desperdicios y oportunidades de optimización, al tiempo que ayudan a medir la efectividad de la organización en la entrega de resultados tangibles. Un flujo de valor bien definido contribuye a reducir el lead time (tiempo desde la solicitud hasta la entrega), aumentar la productividad y mejorar la experiencia del cliente final.

En la práctica, los flujos de valor de ITIL v4 pueden diseñarse para diversos propósitos, entre ellos:

- Entrega de servicios nuevos o modificados.
- Gestión y resolución de incidencias.
- Atención de requerimientos o solicitudes.
- Gestión de problemas y análisis de causas raíz.

Por ejemplo, en el contexto de la mesa de ayuda descrita en esta tesis, el flujo de valor para la gestión de incidencias puede representarse como:

- Engage → recepción y registro de la incidencia;
- Deliver & Support → diagnóstico y resolución;
- Improve → retroalimentación y documentación de la mejora;
- Plan → ajuste de procedimientos según los resultados obtenidos.

De forma similar, en la fábrica de software, un flujo de valor típico podría incluir las actividades Plan → Obtain/Build → Design & Transition → Deliver & Support, con integración continua y despliegue automatizado bajo la filosofía DevOps. Este esquema permite mantener la coherencia entre el desarrollo, las pruebas, el despliegue y la operación del servicio.

La principal ventaja del uso de flujos de valor es que facilita la medición de la eficiencia operativa y la optimización basada en datos. Cada flujo puede evaluarse a través de indicadores específicos, como el tiempo promedio de resolución (Mean Time to Resolve, MTTR), el cumplimiento de los acuerdos de servicio (Service Level Agreements, SLA) o la satisfacción del cliente (Customer Satisfaction, CSAT).

Así, los flujos de valor constituyen un elemento esencial en la gestión moderna de servicios de TI, ya que permiten entender cómo las actividades técnicas, organizativas y estratégicas se integran en una secuencia lógica orientada al valor. En el modelo propuesto en esta tesis, los flujos de valor sirven como base para la integración entre la mesa de ayuda (ITIL v4) y la fábrica de software (Scrum y DevOps), garantizando una trazabilidad completa desde la detección del incidente o requerimiento hasta la entrega final de la solución.

2.1.5 Métricas e indicadores de desempeño

En el marco de ITIL v4, las métricas e indicadores de desempeño constituyen una herramienta esencial para evaluar la eficacia, eficiencia y valor generado por los servicios de TI. A diferencia de versiones anteriores, ITIL v4 enfatiza un enfoque integral de medición que combina métricas técnicas, operativas y de valor para el negocio, con el propósito de garantizar una mejora continua basada en evidencias cuantitativas (AXELOS, 2021a).

El propósito de medir no es solo controlar el rendimiento, sino demostrar el valor que la gestión de servicios aporta al negocio. Para lograrlo, ITIL v4 recomienda la utilización de indicadores estratégicos, tácticos y operativos que reflejen el impacto de los servicios en términos de calidad, productividad y satisfacción del usuario final.

Tipos de métricas en ITIL v4

Según AXELOS (2021), las métricas en la gestión de servicios de TI se agrupan en tres niveles:

- Métricas tecnológicas: Evalúan la capacidad, disponibilidad y estabilidad de la infraestructura tecnológica. Se centran en indicadores de rendimiento de sistemas, redes, almacenamiento y disponibilidad de recursos. Ejemplo: porcentaje de uptime del servicio o rendimiento promedio de CPU y memoria.
- Métricas de procesos: Analizan la eficiencia y efectividad de las prácticas operativas. Permiten identificar cuellos de botella, medir tiempos de respuesta y evaluar el cumplimiento de acuerdos de servicio (SLA). Ejemplo: Mean Time to Resolve (MTTR), Mean Time Between Failures (MTBF), o porcentaje de cumplimiento de SLA.

- Métricas de valor (business metrics): Determinan el impacto de los servicios sobre los resultados estratégicos del negocio. Incluyen indicadores financieros y de experiencia del cliente, como Return on Investment (ROI), Customer Satisfaction (CSAT) y Net Promoter Score (NPS).

El uso equilibrado de estos tres niveles garantiza una evaluación completa del desempeño del servicio, permitiendo identificar tanto los resultados operativos inmediatos como los beneficios estratégicos a largo plazo.

Indicadores clave de desempeño (Key Performance Indicators, KPI)

Los KPI constituyen métricas específicas utilizadas para evaluar el grado de cumplimiento de los objetivos definidos en las prácticas de gestión. En la práctica de Gestión de Incidentes, por ejemplo, los KPI más utilizados incluyen:

- Porcentaje de incidentes resueltos en el primer contacto (First Contact Resolution Rate).
- Tiempo promedio de resolución (Average Resolution Time).
- Porcentaje de cumplimiento de los SLA.
- Tasa de reincidencia de incidentes.
- En la práctica de Gestión de Cambios, algunos KPI típicos son:
- Porcentaje de cambios implementados sin incidentes.
- Duración promedio del ciclo de cambio.
- Número de cambios urgentes realizados fuera de calendario.

De igual forma, en la Gestión de Requerimientos, los KPI se orientan al tiempo promedio de entrega y al grado de alineamiento del desarrollo con las necesidades del usuario.

Estos indicadores proporcionan una base objetiva para la toma de decisiones informada, permitiendo priorizar acciones correctivas, evaluar el desempeño de los equipos y justificar inversiones en mejora de servicios.

Relación de las métricas con la mejora continua

ITIL v4 concibe la medición como un elemento transversal al Sistema de Valor del Servicio (SVS). Cada práctica debe incorporar un esquema de medición que permita observar su evolución y contribución al valor total entregado.

El ciclo de medición propuesto por ITIL se apoya en el principio de "Progresar iterativamente con retroalimentación", donde los resultados de las métricas alimentan las fases de planificación y mejora (AXELOS, 2019).

En el contexto de esta tesis, las métricas permiten evaluar el impacto del modelo propuesto tanto en la mesa de ayuda como en la fábrica de software. Por ejemplo:

En la mesa de ayuda, se evalúan los indicadores de satisfacción (CSAT), cumplimiento de SLA, tiempo promedio de atención y reducción del MTTR.

En la fábrica de software, se analizan los tiempos de desarrollo, el número de incidencias posdespliegue y la frecuencia de entregas (deployment frequency), coherentes con las métricas de DevOps.

El análisis conjunto de estos indicadores demuestra la contribución del modelo ITIL–DevOps a la mejora del desempeño operacional y estratégico de la organización, promoviendo la eficiencia, la agilidad y la generación de valor sostenible.

2.1.6 Mejora continua y cultura organizacional del servicio

La Mejora Continua (Continual Improvement) es una práctica transversal y esencial dentro del marco de ITIL v4, cuyo propósito es garantizar que los servicios, prácticas y procesos de la organización mantengan una alineación constante con los objetivos estratégicos y las necesidades cambiantes del negocio. A diferencia de las versiones anteriores, ITIL v4 integra la mejora continua como un componente estructural del Sistema de Valor del Servicio (Service Value System, SVS), otorgándole un papel activo y no meramente reactivo (AXELOS, 2019).

El objetivo de esta práctica es fomentar una cultura de mejora sostenida, en la que cada miembro de la organización participe activamente en la identificación de oportunidades de optimización, promoviendo la innovación, el aprendizaje organizacional y la orientación al

valor. Este enfoque cultural convierte la mejora continua en un ciclo permanente que abarca desde la definición de estrategias hasta la operación diaria de los servicios.

- El modelo de mejora continua en ITIL v4

De acuerdo con AXELOS (2019), ITIL v4 plantea un modelo de mejora continua basado en siete pasos que proporcionan un marco sistemático para planificar y ejecutar acciones de mejora:

  - Identificar la visión: comprender el propósito general y los objetivos estratégicos de la organización.
  - Evaluar la situación actual: analizar el rendimiento actual de los servicios y determinar brechas con respecto al estado deseado.
  - Definir el estado futuro: establecer metas alcanzables y resultados esperados que representen una mejora tangible.
  - Determinar las acciones a ejecutar: priorizar las iniciativas necesarias para alcanzar la visión definida.
  - Implementar las mejoras: poner en práctica los cambios planificados utilizando un enfoque iterativo.
  - Evaluar los resultados: medir el impacto de las mejoras implementadas mediante indicadores definidos.
  - Mantener el ciclo de revisión: institucionalizar el aprendizaje obtenido para retroalimentar el proceso.

Este modelo se fundamenta en el principio de Plan–Do–Check–Act (PDCA) propuesto por Deming, asegurando que la mejora sea un proceso cíclico y medible. De esta manera, la organización evoluciona de forma controlada y basada en evidencias, minimizando riesgos y consolidando aprendizajes.

- Relación entre la mejora continua y la cultura organizacional

El éxito de la mejora continua depende directamente de la cultura organizacional. En este sentido, ITIL v4 promueve la creación de entornos donde la comunicación abierta, la colaboración y el reconocimiento del aprendizaje sean valores fundamentales. Según Abdelkebir et al. (2017), las organizaciones que adoptan prácticas de mejora iterativa fortalecen su madurez operativa, fomentan la innovación y logran una respuesta más ágil ante los cambios tecnológicos.

Además, la mejora continua impulsa la transición de una cultura centrada en los procesos a una cultura centrada en el valor, donde cada práctica se evalúa por su contribución a los resultados del negocio y a la satisfacción del cliente. Este cambio cultural es esencial para sostener el éxito del modelo ITIL–DevOps implementado en esta tesis, ya que promueve la sinergia entre la mesa de ayuda y la fábrica de software, consolidando un ciclo constante de aprendizaje y optimización.

- Vinculación con la gestión ágil y DevOps

La práctica de mejora continua se relaciona estrechamente con los principios de Agile y DevOps, los cuales promueven ciclos iterativos de entrega y retroalimentación. ITIL v4 adopta estos principios para fortalecer la cooperación entre desarrollo y operaciones, eliminando silos y fomentando la integración continua (AXELOS, 2020).

En este contexto, la mesa de ayuda actúa como una fuente permanente de información sobre la calidad del servicio, mientras que la fábrica de software se convierte en el motor de mejora a través de la automatización, pruebas continuas y despliegues iterativos. Ambos componentes, al operar bajo la práctica de mejora continua, garantizan que los servicios evolucionen con base en datos reales y retroalimentación constante.

Así, la mejora continua no solo contribuye a la eficiencia técnica, sino que también fortalece la resiliencia organizacional, permitiendo que los equipos se adapten a entornos de cambio acelerado sin perder control sobre la calidad y el valor entregado.

- Beneficios esperados de la mejora continua

La implementación efectiva de la mejora continua bajo el marco ITIL v4 proporciona beneficios tangibles para las organizaciones, entre los que destacan:

    - Incremento en la satisfacción del cliente y en la percepción de valor.
    - Reducción del tiempo promedio de resolución de incidentes y errores recurrentes.
    - Mayor eficiencia en la utilización de recursos y reducción de costos operativos.
    - Creación de un entorno organizacional orientado al aprendizaje y la innovación constante.

- Consolidación de una cultura de servicio alineada a los objetivos estratégicos del negocio.

En consecuencia, la mejora continua no solo es una práctica operativa, sino un principio rector que refuerza todos los elementos del SVS. Su aplicación sistemática garantiza la sostenibilidad del modelo propuesto en esta tesis y el crecimiento progresivo de la madurez organizacional en la gestión de servicios de TI.

2.1.7 Mesa de Ayuda: Función, Estructura y Gestión del Soporte Técnico

En el enfoque de ITIL 4, la mesa de servicio funciona como el vínculo operativo entre los usuarios y la organización que provee los servicios. Su responsabilidad no se limita a atender fallas técnicas: también centraliza la comunicación, canaliza incidentes y solicitudes, y genera información útil para mejorar la experiencia del usuario y el desempeño del soporte. Por esta razón, en el modelo propuesto se le reconoce como un componente estratégico y como el punto institucional de contacto para registrar, dar seguimiento y coordinar las necesidades relacionadas con los servicios de TI (AXELOS, 2019).

La estructura moderna de una mesa de ayuda incluye múltiples niveles de atención (soporte de primer, segundo y tercer nivel), canales multicanal (teléfono, correo electrónico, portal web, chatbots) y herramientas de gestión como sistemas ITSM (por ejemplo, Jira Service Management, ServiceNow o Freshservice). Su objetivo principal es garantizar la resolución oportuna de incidentes y solicitudes, reducir los tiempos de espera y elevar la satisfacción del usuario final.

En empresas orientadas al servicio, como aquellas que proveen plataformas de beneficios para empleados (por ejemplo, Kred C&S), la mesa de ayuda es crítica para asegurar la disponibilidad y operatividad de aplicaciones clave. Si los usuarios no pueden acceder a sus beneficios o tienen problemas al validar cupones o realizar canjes, la percepción del servicio se ve afectada negativamente. Por ello, una gestión eficiente de la mesa de ayuda no solo mejora los indicadores técnicos, sino que también contribuye a la retención y satisfacción del cliente.

Las mejores prácticas de ITIL v4 recomiendan integrar la práctica de Mesa de Servicio (Service Desk) con otras prácticas de gestión esenciales como Gestión de Incidentes, Gestión de Problemas, Gestión de Cambios y Gestión del Conocimiento, con el fin de optimizar la

coordinación y la entrega de valor dentro del sistema de gestión de servicios. Por ejemplo, una alta frecuencia de consultas similares puede indicar un problema subyacente que requiere análisis causa raíz (gestión de problemas), mientras que la documentación de soluciones recurrentes puede enriquecer la base de conocimiento para empoderar tanto a usuarios como a técnicos (gestión del conocimiento).

Además, la medición del desempeño de la mesa de ayuda se apoya en indicadores clave como:

- Tiempo medio de respuesta y de resolución (MTTR).
- Tasa de resolución en el primer contacto (FCR).
- Nivel de satisfacción del usuario (CSAT).
- Nivel de cumplimiento de acuerdos de nivel de servicio (SLAs).

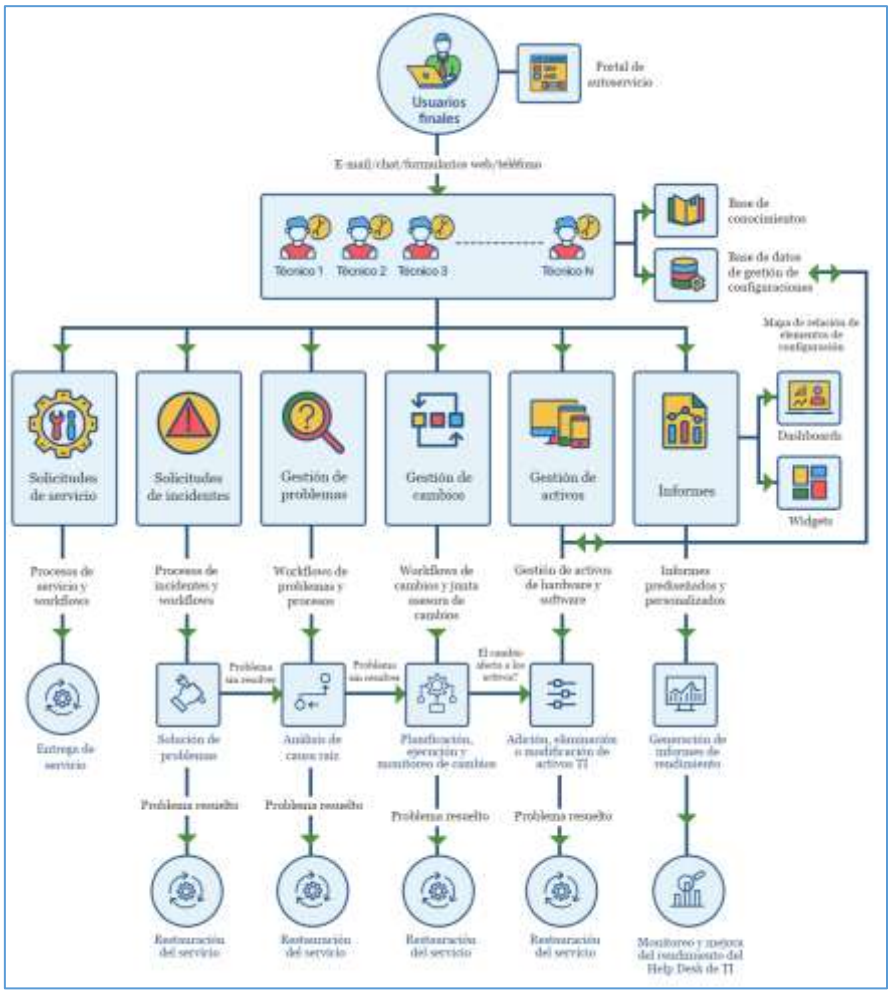
El enfoque moderno de la mesa de ayuda incorpora automatización, inteligencia artificial y autoatención. A través de asistentes virtuales, formularios inteligentes y respuestas automatizadas, se busca resolver un volumen significativo de solicitudes sin intervención humana directa, lo cual libera recursos para tareas más complejas. Esta evolución se alinea con el principio de "optimizar y automatizar" de ITIL v4 (Redondo, 2019).

Finalmente, la mesa de ayuda cumple un rol clave en la retroalimentación hacia el negocio. Las interacciones con los usuarios generan datos valiosos sobre patrones de uso, necesidades emergentes y oportunidades de mejora. Esta información, si se canaliza adecuadamente, puede influir en el diseño de nuevos servicios, la priorización de desarrollos y la toma de decisiones estratégicas.

Por tanto, la mesa de ayuda moderna no es solo un centro de soporte, sino una plataforma de interacción, aprendizaje y evolución organizacional. Su gestión efectiva contribuye directamente a la estabilidad operativa, la experiencia del cliente y el posicionamiento competitivo de la organización.

Figura 4

Diagrama de flujo del proceso de gestión de servicios del Help Desk de TI



Nota. Adaptado de Características del software de mesa de ayuda y servicio TI [Infografía], por ManageEngine ServiceDesk Plus, s.f.-a (https://www.manageengine.com/latam/service-desk/caracteristicas-mesa-de-ayuda.html).

2.1.8 Integración ITIL v4 y Scrum: Sinergia para la Agilidad y Estabilidad del Servicio

La integración de ITIL v4 y Scrum representa una convergencia entre dos enfoques complementarios que permiten a las organizaciones combinar la estabilidad operativa con la agilidad en la entrega de valor. Mientras ITIL v4 proporciona una estructura para la gobernanza, el control de calidad y la mejora continua de los servicios de TI, Scrum aporta velocidad, flexibilidad y una orientación intensa hacia el cliente a través de entregas iterativas (AXELOS, 2019; Schwaber & Sutherland, 2020).

Esta sinergia resulta especialmente valiosa en entornos donde la operación y el desarrollo deben alinearse para responder de manera efectiva a cambios tecnológicos, regulatorios o de negocio. En empresas como aquellas dedicadas a ofrecer beneficios para empleados, esta integración permite abordar simultáneamente la gestión eficaz de incidentes desde la mesa de ayuda y la mejora continua del producto desde la fábrica de software.

Desde una perspectiva operativa, ITIL v4 ofrece prácticas como la gestión de incidentes, problemas, cambios, SLAs y conocimiento, las cuales aseguran la estabilidad, disponibilidad y calidad del servicio. Por otro lado, Scrum estructura el trabajo en ciclos cortos y definidos (sprints), promoviendo la colaboración entre equipos multifuncionales, la inspección frecuente y la adaptación constante. Esta estructura permite incorporar los aprendizajes operativos (por ejemplo, causas recurrentes de fallas) como insumos para nuevos desarrollos o ajustes del producto.

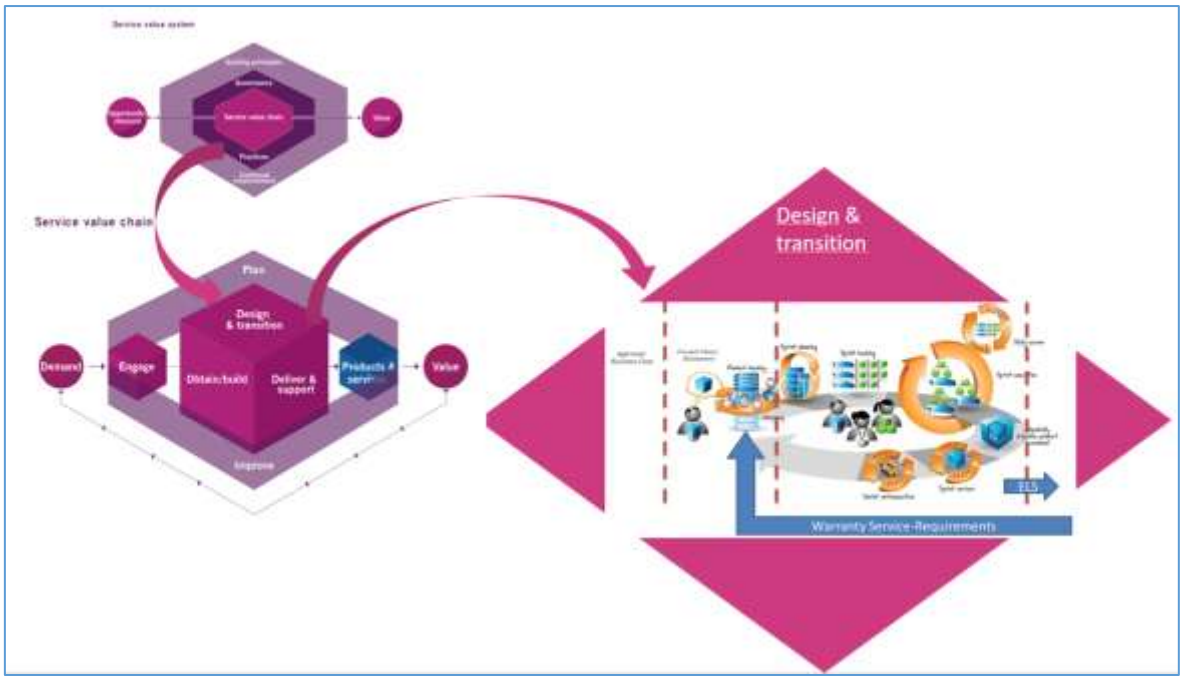
Un ejemplo práctico de esta integración se da cuando una incidencia detectada en la mesa de ayuda es documentada como historia de usuario y priorizada en el Product Backlog para ser abordada en el siguiente sprint. De igual forma, un cambio aprobado por el comité CAB puede planificarse en un sprint futuro, con seguimiento de su impacto durante la revisión del sprint (Sprint Review) y ajustes procesables discutidos en la retrospectiva (Sprint Retrospective).

Además, la información proveniente de los SLAs, reportes operativos y análisis de incidentes puede alimentar la planificación de implementaciones, mejorar los criterios de aceptación de historias de usuario y enriquecer la base de conocimiento accesible por el equipo Scrum. La figura del Scrum Master puede colaborar activamente con roles como el gestor de cambios o el coordinador de incidentes, fomentando un lenguaje común entre equipos de soporte y desarrollo (Sachs, 2021).

Esta integración también promueve una visión de valor centrada en el cliente, alineando la entrega técnica con los resultados esperados del negocio. Al adoptar un marco común de principios, como los de ITIL v4 (por ejemplo, "enfocarse en el valor" y "colaborar y promover la visibilidad") y de Scrum (como "transparencia" y "adaptación"), las organizaciones construyen una cultura de colaboración, mejora continua y aprendizaje colectivo.

En conclusión, la convergencia de ITIL v4 y Scrum permite que los equipos operativos y de desarrollo trabajen con objetivos compartidos, utilizando prácticas estandarizadas pero adaptativas, lo que mejora el flujo de trabajo, reduce los tiempos de respuesta y eleva la calidad del servicio entregado. Esta integración es una estrategia poderosa para organizaciones que buscan equilibrio entre estabilidad y agilidad en su encaminamiento digital (Bradley, 2022; Pressman & Maxim, 2020).

Figura 5
Integración del Sistema de Valor del Servicio ITIL 4 con metodología Scrum en la actividad de Diseño y Transición



Nota. La sección derecha fue desarrollada por los autores para integrar Scrum y Warranty Service Requirements. Adaptado de ITIL® 4: Foundation Edition, por AXELOS, 2019.

2.1.9 Conclusiones del Marco Teórico

A lo largo de este marco teórico se ha demostrado que la convergencia entre ITIL v4 y Scrum ofrece una solución integral y coherente para enfrentar los desafíos contemporáneos en la gestión de servicios de TI y el desarrollo de software. ITIL v4, con su enfoque basado en principios rectores, prácticas adaptables y el Sistema de Valor del Servicio (SVS), permite estructurar y gobernar los servicios tecnológicos con una orientación clara hacia la creación

de valor. Por su parte, Scrum proporciona un método ágil, iterativo y centrado en el cliente para entregar soluciones funcionales de manera rápida y flexible.

Las prácticas clave de ITIL v4 —en especial la gestión de incidentes, problemas, cambios, conocimiento y SLAs— han demostrado ser fundamentales para sostener la continuidad operativa, asegurar la calidad del servicio y fomentar la mejora continua. A través de ceremonias estructuradas como las revisiones post-incidentes y los comités de cambios, y mediante el monitoreo de indicadores de desempeño, se logra una operación más predecible, controlada y orientada a resultados.

Por otro lado, las ceremonias de Scrum (Sprint Planning, Daily Scrum, Sprint Review y Retrospective) no solo fomentan la transparencia y la colaboración, sino que también refuerzan una cultura organizacional de aprendizaje continuo. Estas prácticas permiten transformar rápidamente la retroalimentación operativa en oportunidades de mejora del producto o del servicio.

La integración efectiva de ITIL v4 y Scrum favorece la construcción de un entorno colaborativo entre los equipos de soporte técnico y desarrollo, alineando sus objetivos y procesos mediante un lenguaje común y principios compartidos. Esta sinergia resulta particularmente beneficiosa en organizaciones como aquellas dedicadas a ofrecer soluciones de beneficios para empleados, donde la excelencia operativa y la velocidad de innovación son factores competitivos clave.

En conclusión, este marco teórico sustenta la viabilidad y el valor de implementar un modelo de mejora en la atención de la mesa de ayuda y el desarrollo de software basado en ITIL v4 y Scrum. La aplicación articulada de ambos enfoques no solo mejora la eficiencia operativa y la calidad del servicio, sino que también potencia la capacidad de adaptación y respuesta de las organizaciones ante los desafíos actuales.

2.2 Estado del Arte

El artículo “Towards Cloud Computing Management Model Based on ITIL Processes” propone una estructura metodológica para adaptar los procesos tradicionales de ITIL a los desafíos actuales de la computación en la nube. Esta investigación parte del reconocimiento de que, si bien ITIL ha sido exitoso en la gestión de infraestructuras de TI tradicionales, su aplicación directa a entornos de cloud computing requiere una transformación que conserve los principios de control, disponibilidad y calidad, al mismo tiempo que incorpore flexibilidad, elasticidad y escalabilidad, características fundamentales de los servicios en la nube.

Los autores inician contextualizando el crecimiento exponencial de los servicios basados en la nube (IaaS, PaaS y SaaS), que han obligado a los departamentos de TI a reformular sus esquemas de gestión. Las prácticas convencionales de administración de sistemas, servidores locales y redes on-premise resultan ineficaces cuando los recursos tecnológicos son entregados bajo demanda, alojados en centros de datos distribuidos y contratados como servicios en plataformas públicas, privadas o híbridas.

En este marco, ITIL sigue siendo una referencia clave, pero su implementación en la nube debe considerar nuevos aspectos como el autoservicio, la automatización de tareas rutinarias, la provisión dinámica de recursos, la medición granular del consumo y la seguridad compartida entre proveedor y cliente. El artículo establece que no se trata de descartar ITIL, sino de reinterpretarlo y adaptarlo para que continúe siendo relevante y útil en estos nuevos escenarios.

El modelo propuesto se estructura sobre los cinco dominios del ciclo de vida del servicio de ITIL v3 (y en parte v4): estrategia, diseño, transición, operación y mejora continua. En cada etapa, los autores explican cómo se deben redefinir las actividades cuando se opera en un contexto cloud. Por ejemplo, en la etapa de diseño del servicio, ya no basta con definir especificaciones técnicas y acuerdos de niveles de servicio (SLA); ahora es necesario considerar acuerdos sobre elasticidad, disponibilidad por zonas geográficas y tolerancia a fallos en arquitecturas multicliente.

Asimismo, en la fase de operación, cobra especial importancia la automatización de la provisión, monitoreo y escalado de servicios. El modelo plantea el uso de herramientas como scripts de infraestructura como código (IaC), monitoreo con inteligencia artificial (AIOps),

y dashboards que integren alertas, consumo, rendimiento y cumplimiento de SLA en tiempo real. Estos elementos permiten mantener los principios de ITIL —como disponibilidad, continuidad, capacidad y seguridad—, pero ejecutados con herramientas modernas adaptadas al cloud.

Una de las grandes contribuciones del artículo es la definición de un metamodelo de gestión que traduce procesos clásicos de ITIL (como gestión de incidencias, gestión de cambios, gestión del catálogo de servicios) en funciones adaptadas a proveedores de nube como Amazon Web Services, Microsoft Azure o Google Cloud. Por ejemplo, la gestión de incidentes incorpora automatización para redireccionar alertas hacia bots o flujos de trabajo sin intervención humana. La gestión del cambio se adapta al paradigma DevOps mediante pipelines que validan y despliegan cambios en contenedores distribuidos. La gestión del catálogo se convierte en un portal digital dinámico que permite al usuario interno autoprovisionarse servicios con permisos definidos por políticas.

El modelo incluye un componente de mejora continua, donde se recopilan métricas del comportamiento del servicio en la nube (latencia, throughput, consumo de CPU/memoria, errores HTTP, etc.) y se analizan mediante herramientas de analítica avanzada. Esto refuerza la capacidad de adaptación del servicio a nuevas condiciones de carga, errores emergentes y demandas estacionales del negocio. Aquí se plantea la integración de prácticas de ITIL con modelos ágiles mostrando cómo las evaluaciones retros pueden traducirse en historias de usuario o ítems del backlog de mejora técnica.

En relación con la mesa de ayuda, el artículo enfatiza que la atención a usuarios debe cambiar radicalmente cuando se adoptan servicios cloud. El primer nivel de soporte se vuelve más automatizado, gracias a la implementación de asistentes virtuales, respuestas programadas y redireccionamiento contextual. Además, el segundo nivel actúa como facilitador del autoservicio, guiando al usuario en la resolución sin intervención directa. Las bases de conocimiento evolucionan para integrarse con portales interactivos de aprendizaje y FAQs dinámicos alimentados por inteligencia artificial.

Este modelo de soporte, alineado con ITIL y adaptado al entorno cloud, también propone cambios en los roles. Por ejemplo, surge la figura del cloud service manager, quien es responsable de mantener relaciones con el proveedor de servicios en la nube, verificar el cumplimiento de los SLA y traducir las necesidades del negocio en especificaciones técnicas

de servicios contratados. Del lado del cliente, el personal de soporte debe adquirir competencias en integración, ciberseguridad en entornos multitenant, y administración de contratos con múltiples proveedores.

Finalmente, el artículo concluye que ITIL no está obsoleto, sino que necesita evolucionar hacia una arquitectura de servicios más flexible y descentralizada. Esto se logra integrando prácticas de automatización, autoservicio y gobernanza basada en datos, lo cual es perfectamente compatible con la filosofía de ITIL si se ejecuta con un enfoque adaptativo.

Para la tesis, este artículo resulta fundamental porque valida que ITIL puede utilizarse no solo en entornos on-premise tradicionales, sino también como marco para gobernar servicios en la nube, que es precisamente el tipo de entorno donde muchas mesas de ayuda y fábricas de software modernas operan actualmente. La propuesta de un modelo integrado ITIL + Scrum puede beneficiarse de esta visión, ya que refuerza la noción de que los procesos definidos deben coexistir con prácticas ágiles y entornos tecnológicos cambiantes.

El artículo Recommending Resolutions of ITIL Services Tickets Using Deep Neural Network aborda una problemática central en la gestión de servicios de TI bajo el enfoque ITIL: la resolución eficiente y precisa de tickets de soporte. A partir de un estudio aplicado en organizaciones que utilizan sistemas estructurados de atención a incidentes, los autores proponen un modelo basado en redes neuronales profundas (DNN) para recomendar soluciones a problemas comunes reportados en la mesa de ayuda. Esta innovación se sitúa en la intersección entre ITIL y la inteligencia artificial, y tiene un impacto directo sobre la mejora del tiempo de respuesta, la reducción del esfuerzo humano y la estandarización del conocimiento técnico.

El punto de partida del artículo es la gran acumulación de tickets históricos que se generan diariamente en entornos corporativos de TI. Estos tickets, aunque diversos, presentan patrones de recurrencia y contienen valiosa información para entrenar modelos de aprendizaje automático. El enfoque tradicional —que consiste en asignar manualmente los casos y depender del conocimiento de cada agente de soporte— presenta limitaciones como demoras, decisiones subjetivas y variabilidad en la calidad de las soluciones. Frente a ello, los autores proponen automatizar parte del proceso de recomendación de soluciones mediante redes neuronales profundas entrenadas con datos históricos anonimizados.

La arquitectura propuesta se basa en una red neuronal multicapa que toma como entrada los campos textuales del ticket (título, descripción, categoría) y devuelve como salida una recomendación de resolución extraída de un conjunto de soluciones aprobadas. El modelo se entrena usando un corpus etiquetado previamente por expertos, lo que permite que la red aprenda a identificar las relaciones semánticas entre las palabras y las soluciones aplicadas anteriormente. En términos técnicos, el artículo detalla el uso de técnicas de procesamiento de lenguaje natural (NLP), embeddings semánticos y optimización con algoritmos tipo Adam para alcanzar una precisión aceptable en ambientes de prueba.

Desde la perspectiva de ITIL, esta propuesta encaja dentro del proceso de gestión de incidentes y gestión del conocimiento. En el marco ITIL v4, la gestión de incidentes busca restablecer el servicio lo más rápido posible minimizando el impacto. La propuesta de los autores fortalece esta práctica, ya que la automatización de recomendaciones permite que el primer nivel de soporte tenga acceso inmediato a posibles soluciones, sin depender de escalamiento. En paralelo, la gestión del conocimiento se ve enriquecida, ya que el sistema aprende continuamente de los nuevos tickets y ajusta sus predicciones, convirtiéndose en una base de datos viva que se retroalimenta con cada interacción.

El artículo también analiza los beneficios cuantitativos esperados: reducción del tiempo promedio de resolución (TTR), menor necesidad de intervención humana, estandarización de respuestas y aumento en la satisfacción del usuario interno. A nivel cualitativo, destaca que los agentes de soporte pueden dedicar su tiempo a casos complejos, mientras que los tickets repetitivos y de bajo impacto se resuelven automáticamente o con mínima asistencia. Este enfoque mejora la eficiencia operativa y permite una mejor asignación de recursos humanos en la mesa de ayuda.

Uno de los apartados más relevantes para esta tesis es el estudio de caso incluido en el artículo. Se presenta la implementación del modelo en una organización con más de 150,000 tickets históricos. Tras entrenar la red con una muestra representativa, se logró una tasa de precisión del 82% en la recomendación de soluciones. Además, el tiempo de primera respuesta disminuyó en un 35% y el índice de reabiertos se redujo significativamente, lo que demuestra que el modelo no solo responde rápido, sino que también lo hace con calidad.

El artículo también aborda los desafíos técnicos y éticos de la propuesta. Por un lado, la calidad del modelo depende directamente de la calidad de los datos históricos: si los registros

están incompletos, mal categorizados o contienen errores, el modelo aprenderá con sesgos o inconsistencias. Por otro lado, el sistema debe garantizar privacidad, ya que muchos tickets contienen información sensible del usuario o del entorno empresarial. En ese sentido, los autores destacan la necesidad de anonimizar los datos y establecer políticas de auditoría para la trazabilidad de las recomendaciones generadas por la inteligencia artificial.

En términos de aplicabilidad, el modelo es extensible no solo a la gestión de incidentes, sino también a solicitudes de servicio, preguntas frecuentes, configuración de dispositivos y otros procesos estándar. En combinación con herramientas de ITSM como ServiceNow o Jira Service Management, puede integrarse como una capa adicional que analiza en tiempo real los tickets y propone soluciones incluso antes de que un agente humano los revise. Esto se alinea con la visión de una mesa de ayuda proactiva y predictiva, capaz de anticiparse a las necesidades del usuario.

En relación con el desarrollo de software, este enfoque también puede vincularse con Scrum. Las historias de usuario relacionadas con incidencias recurrentes, fallos funcionales o mejoras detectadas por el sistema de tickets pueden priorizarse automáticamente en el backlog del producto. Así, el aprendizaje del sistema no solo soluciona problemas actuales, sino que alimenta el desarrollo de soluciones más robustas a futuro. Esta conexión entre la analítica operativa (ITIL) y la priorización estratégica (Scrum) es clave para una organización que desea ser verdaderamente ágil.

Para la tesis, el aporte de este artículo es valioso por varias razones. Primero, valida que ITIL puede beneficiarse enormemente de tecnologías emergentes como la inteligencia artificial, lo cual moderniza y potencia sus procesos tradicionales. Segundo, demuestra que la automatización no elimina el rol humano, sino que lo redefine, liberando a los operadores de tareas repetitivas para enfocarse en análisis y mejora. Tercero, plantea un caso de uso concreto donde la mesa de ayuda se convierte en un nodo inteligente, conectado con el desarrollo, el negocio y la infraestructura.

Finalmente, el artículo encaja perfectamente en la propuesta de un modelo integrado ITIL v4 + Scrum, ya que la automatización de la atención se puede complementar con la gestión ágil de cambios y la entrega iterativa de mejoras. Un sistema que recomienda soluciones también puede sugerir funcionalidades, documentarlas como historias de usuario y alimentarlas en ciclos ágiles. Este flujo de trabajo automatizado, supervisado y orientado a

la mejora continua constituye la base de una organización verdaderamente alineada con los principios de la agilidad y la gestión de servicios.

El artículo “ITIL: Share and Share Alike?” de Sharon Taylor, quien fuera jefa de arquitectura de ITIL durante su expansión global, ofrece una reflexión crítica y al mismo tiempo constructiva sobre la evolución de ITIL desde sus orígenes como un conjunto de buenas prácticas del gobierno británico hasta su adopción mundial como estándar de facto para la gestión de servicios de TI. Esta fuente resulta especialmente valiosa para contextualizar el marco ITIL dentro de procesos reales, identificar sus fortalezas, limitaciones y, sobre todo, comprender su transformación hacia una herramienta flexible, adaptable y colaborativa que puede integrarse con enfoques ágiles como Scrum.

El texto parte de una premisa contundente: ITIL no es un modelo de implementación única ni una norma obligatoria, sino una colección de buenas prácticas basadas en casos de éxito. A diferencia de ISO/IEC 20000, que establece requisitos para certificar organizaciones, ITIL se define como un cuerpo de conocimiento que orienta y guía, pero que debe ser personalizado por cada organización según su cultura, objetivos, madurez tecnológica y capacidades operativas. Esta distinción es clave, ya que libera a ITIL de la rigidez normativa y le otorga una gran capacidad de adaptación.

Uno de los ejes del artículo es la historia de cómo ITIL fue adoptado en distintos países y sectores, pasando de un modelo desarrollado por la CCTA (Central Computer and Telecommunications Agency) en el Reino Unido a una plataforma de colaboración global. Este crecimiento fue impulsado por la necesidad compartida entre organizaciones de distintas industrias (banca, salud, gobierno, educación) de contar con un lenguaje común y procesos estándar para gestionar sus servicios tecnológicos. A medida que ITIL se difundía, fue alimentándose de experiencias reales, generando una comunidad de práctica que retroalimentó el modelo original.

Taylor destaca que ITIL fue exitoso en gran medida por su capacidad de generar consenso entre prácticas diversas. A través de un enfoque modular, permitió que las organizaciones adoptaran solo los procesos que les resultaban relevantes, sin necesidad de implementar todo el conjunto. Este modularidad fue clave para pequeñas y medianas empresas, que muchas veces requieren estructura sin sobrecarga administrativa. Además, la autora reconoce que ITIL no es estático, sino que ha evolucionado gracias a las contribuciones de miles de

profesionales, consultores y académicos que lo han puesto a prueba y mejorado con base en evidencia empírica.

Un aporte significativo del artículo es la aclaración sobre lo que ITIL no es. No es un sistema de software, no es un paquete metodológico cerrado y no garantiza resultados si se aplica mecánicamente. En palabras de la autora, ITIL es una "plataforma de entendimiento y diálogo", cuyo valor real radica en la capacidad de la organización para interpretar y contextualizar sus principios. Esta visión desmonta mitos comunes y ayuda a evitar implementaciones fallidas basadas en recetas genéricas o consultorías que promueven modelos rígidos.

La reflexión de Taylor también aborda las críticas más frecuentes a ITIL: su presunta complejidad, su orientación burocrática y su lentitud para responder al cambio. Frente a esto, la autora señala que muchas veces los problemas no radican en ITIL en sí mismo, sino en su implementación sin sentido crítico. Las organizaciones que lo adoptan de forma literal, sin adaptarlo a su realidad, tienden a sobreprocesar, generar documentación innecesaria y frenar la innovación. Por el contrario, aquellas que lo utilizan como una guía flexible, lo combinan con métodos ágiles y promueven la autonomía del equipo, logran una mayor efectividad.

Una parte destacada del artículo analiza la transición hacia ITIL v3 y los principios que dieron origen a la versión v4. La introducción del ciclo de vida del servicio, los cinco libros (estrategia, diseño, transición, operación y mejora continua), y el enfoque en el valor para el negocio, marcaron un cambio significativo. Taylor resalta que esta evolución respondió a una necesidad real de integrar ITIL con otros marcos como COBIT, PMBOK, PRINCE2 y, sobre todo, con enfoques ágiles como Lean y Scrum. Este cambio permitió a ITIL mantenerse vigente en contextos de manejo de los recursos de internet, DevOps y entrega continua.

Desde la perspectiva de la tesis, este artículo es muy útil para justificar por qué ITIL sigue siendo un marco relevante incluso en entornos ágiles. Contrario a la creencia de que ITIL es incompatible con Scrum, el artículo refuerza la idea de que ambos pueden coexistir si se entienden sus límites y propósitos. Por ejemplo, ITIL proporciona estructura, roles y procesos para el soporte técnico, la gestión del conocimiento, la gestión de cambios y la mejora continua. Scrum, en cambio, ofrece velocidad, flexibilidad y colaboración para el desarrollo de software. Integrar ambos marcos, como propone la tesis, implica generar un

ecosistema donde la mesa de ayuda pueda nutrirse de buenas prácticas ITIL, mientras que el equipo de desarrollo pueda operar con Scrum, pero con visibilidad y trazabilidad de los requerimientos de usuario provenientes de incidentes reales.

El artículo también hace hincapié en el rol de la comunidad de práctica en la evolución de ITIL. Esta comunidad incluye profesionales de TI, líderes técnicos, académicos, desarrolladores de software y usuarios finales que, a través de foros, conferencias, y redes de colaboración, han contribuido a refinar las prácticas y compartir casos de éxito y fracaso. Este componente colaborativo es particularmente inspirador para una tesis de maestría, ya que muestra que ITIL no es solo teoría, sino un marco vivo y en constante aprendizaje.

En conclusión, "ITIL: Share and Share Alike?" no solo ofrece una historia del marco ITIL, sino una visión crítica y estratégica sobre su adopción y evolución. Aporta a la tesis de forma sustancial porque justifica la aplicabilidad de ITIL en entornos modernos, ayuda a entender los riesgos de una implementación inflexible y refuerza la posibilidad de combinarlo con prácticas ágiles. Este artículo fundamenta la viabilidad de usar ITIL como base para estructurar la mesa de ayuda, sin perder agilidad ni frenar el desarrollo de software, siempre que se promueva una implementación flexible, contextual y basada en el valor para el usuario.

El artículo "How We Deliver Our Desktop Support Services to Washington University in St. Louis the ITIL Way" describe una experiencia concreta de implementación del marco ITIL en un entorno universitario. A diferencia de otros artículos centrados en teoría o revisión de literatura, este trabajo ofrece una perspectiva empírica basada en la transformación del servicio de soporte técnico de una universidad grande, permitiendo observar directamente cómo se implementan los procesos clave de ITIL en la práctica.

El contexto del artículo es el departamento de soporte de TI de la Universidad de Washington en St. Louis, una institución con una comunidad diversa de usuarios —estudiantes, profesores, personal administrativo— y una infraestructura tecnológica distribuida. La necesidad de ofrecer servicios eficientes, escalables y predecibles llevó a los responsables de TI a adoptar ITIL como marco rector. El objetivo no era simplemente resolver problemas técnicos, sino profesionalizar el servicio, elevar la satisfacción del usuario y establecer estándares de atención que pudieran ser medidos y mejorados con el tiempo.

El artículo detalla los procesos que fueron priorizados en la implementación inicial: gestión de incidentes, gestión de solicitudes, gestión del conocimiento y gestión de niveles de servicio. El enfoque adoptado fue progresivo, comenzando con la estandarización del proceso de atención de tickets. Se diseñó un flujo que partía desde la recepción del incidente (por teléfono, correo electrónico o sistema web), pasaba por una clasificación automatizada por tipo y prioridad, y culminaba con la resolución por el nivel de soporte correspondiente. Esta estandarización permitió disminuir errores, reducir el tiempo promedio de atención y facilitar la recolección de datos.

Una de las prácticas más exitosas fue la implementación del modelo de niveles de soporte (Nivel 1, Nivel 2, Nivel 3), donde cada grupo tenía roles, herramientas y umbrales de intervención definidos. El Nivel 1 resolvía incidentes comunes (reseteo de contraseñas, instalación de software), el Nivel 2 atendía requerimientos técnicos específicos (configuraciones de red, periféricos, seguridad), y el Nivel 3 se reservaba para incidentes complejos o escalados al área de infraestructura o proveedores externos. Esta división redujo la sobrecarga del personal técnico especializado y elevó la tasa de resolución en el primer contacto (FCR).

Otro eje importante fue el monitoreo mediante SLAs (acuerdos de niveles de servicio). La universidad estableció tiempos máximos de respuesta y resolución para cada tipo de ticket según su prioridad. Estos parámetros fueron compartidos con los usuarios, lo cual aumentó la transparencia del servicio. La medición de cumplimiento de SLAs permitió detectar cuellos de botella, justificar recursos adicionales, y establecer metas de mejora continua.

El artículo resalta el rol de la gestión del conocimiento como catalizador del cambio. Se creó una base de conocimientos accesible tanto para el personal de soporte como para los usuarios, con artículos, guías, procedimientos y soluciones documentadas. Esta estrategia redujo la carga del soporte Nivel 1 y empoderó a los usuarios para resolver incidentes menores por sí mismos. La base de conocimientos fue gestionada con criterios de revisión periódica, retroalimentación de los usuarios y control de versiones.

En cuanto al componente humano, el equipo de soporte fue sometido a capacitación en ITIL Foundation, lo cual unificó el lenguaje y entendimiento del marco. Además, se implementaron reuniones semanales para revisión de métricas, análisis de tendencias, resolución de problemas repetitivos y propuestas de mejora. Se fortaleció la comunicación

entre niveles de soporte y se crearon canales de retroalimentación para que los agentes pudieran escalar observaciones sobre procesos ineficientes o necesidades recurrentes.

El caso de la Universidad de Washington en St. Louis es relevante para la tesis por varios motivos. En primer lugar, muestra cómo ITIL puede ser aplicado en organizaciones complejas con múltiples tipos de usuarios y requerimientos tecnológicos diversos. Segundo, ilustra cómo los procesos de ITIL no solo estandarizan la operación, sino que también habilitan una cultura de mejora continua y enfoque en el cliente. Tercero, demuestra que es posible medir el éxito de una implementación ITIL con indicadores concretos como reducción del tiempo de resolución, aumento en la satisfacción del usuario y mejora en la resolución en el primer contacto.

Desde el punto de vista metodológico, el artículo también aporta una estrategia clara de adopción gradual, comenzando por los procesos más visibles (incidentes, solicitudes) y luego ampliando a procesos más maduros como problemas, cambios y configuración. Esto respalda la recomendación de la tesis de implementar ITIL en fases, comenzando con lo más crítico para la operación de la mesa de ayuda.

Además, el artículo da evidencia de que la articulación entre los diferentes niveles de soporte y la gestión del conocimiento permite reducir el volumen de tickets escalados, lo que libera al equipo de desarrollo de interrupciones operativas innecesarias. Esta idea encaja directamente con la propuesta de integrar ITIL (mesa de ayuda) con Scrum (desarrollo), evitando que problemas operativos se conviertan en fuentes de distracción para el equipo de software.

Finalmente, el artículo subraya que ITIL no es incompatible con la cultura universitaria, que a menudo valora la autonomía, la descentralización y la flexibilidad. Por el contrario, ITIL permitió establecer claridad de roles, canales de comunicación eficientes, y métricas de evaluación sin sacrificar la adaptabilidad del servicio. Esto refuerza el argumento de que ITIL puede ser útil no solo en empresas tradicionales, sino también en entornos ágiles e innovadores.

El artículo “E-Government ITIL-Based Service Management Case Study” examina un caso de estudio concreto en el que una entidad gubernamental implementa ITIL para mejorar sus servicios digitales y de soporte a ciudadanos. El estudio se basa en la implementación de un sistema de gestión de servicios ITIL en un entorno de gobierno electrónico (e-Gov), y tiene

como objetivo principal demostrar cómo las mejores prácticas de ITIL pueden mejorar la eficiencia, la transparencia y la calidad en la entrega de servicios públicos basados en tecnología.

En su introducción, el artículo plantea el desafío creciente de los gobiernos por ofrecer servicios digitales con niveles de calidad comparables a los del sector privado. A medida que los ciudadanos esperan respuestas rápidas, disponibilidad constante y resolución efectiva de sus solicitudes, las plataformas electrónicas gubernamentales se enfrentan a la necesidad de contar con procesos bien definidos, medibles y controlables. En ese contexto, ITIL es identificado como un marco útil para estructurar la entrega de servicios de TI y estandarizar la atención a incidentes, requerimientos y problemas.

El estudio de caso se enfoca en la implementación de ITIL en un portal de servicios digitales dirigido a ciudadanos y empresas. La entidad gubernamental enfrentaba problemas comunes como la falta de trazabilidad de incidentes, escasa documentación, demoras en los tiempos de respuesta, duplicidad de esfuerzos entre áreas, y ausencia de métricas para evaluar la eficacia del soporte. A partir de un diagnóstico inicial, se decidió estructurar los procesos de gestión de incidentes, solicitudes de servicio, gestión de cambios, conocimiento y niveles de servicio siguiendo el marco ITIL v3.

Uno de los principales logros del proyecto fue la creación de un centro de servicios unificado, una especie de mesa de ayuda pública donde los ciudadanos podían registrar sus requerimientos, hacer seguimiento de su estado y obtener respuestas desde un canal centralizado. Este centro fue estructurado en niveles de atención (Nivel 1, Nivel 2 y Nivel 3), con flujos bien definidos de escalamiento y resolución. Cada nivel tenía herramientas, competencias y autorizaciones específicas, lo cual permitió disminuir la sobrecarga y mejorar la eficiencia.

El proceso de gestión de incidentes fue automatizado mediante un sistema de tickets que permitía clasificar, asignar y priorizar las solicitudes. Se establecieron tiempos objetivo de respuesta (SLA) diferenciados por tipo de solicitud, y se capacitó al personal de atención para operar bajo criterios de ITIL. Esto generó una mejora sustancial en la tasa de resolución en el primer contacto, el tiempo promedio de atención y la percepción ciudadana del servicio.

Otro aspecto clave del estudio fue la incorporación de la gestión del conocimiento. Se creó una base de datos de soluciones (known errors y workarounds), alimentada por los propios

agentes del centro de servicios y supervisada por un comité técnico. Esta base permitió reducir los tiempos de resolución, estandarizar respuestas y capacitar más rápidamente al personal nuevo. Además, sirvió como herramienta de autoayuda para los usuarios, que podían consultar artículos antes de escalar su incidente.

La gestión del cambio también fue abordada con seriedad. Cada modificación a la plataforma de servicios electrónicos debía pasar por un proceso formal que evaluaba su impacto, riesgos, recursos necesarios y plan de retroceso. Aunque esto agregaba un paso adicional, evitaba interrupciones inesperadas y permitía que los ciudadanos experimentaran una mayor estabilidad y continuidad del servicio.

En términos de métricas, el artículo presenta datos relevantes: tras seis meses de implementación, se logró reducir en un 40% los tiempos promedio de atención, en un 60% los incidentes reabiertos, y en un 25% las quejas ciudadanas relacionadas con TI. Además, se incrementó la transparencia del servicio mediante tableros públicos que mostraban la evolución de los indicadores.

Desde una perspectiva organizacional, el artículo destaca que uno de los factores clave del éxito fue el compromiso de la alta dirección. El liderazgo institucional promovió la cultura de servicios, respaldó financieramente el proyecto y fomentó la formación en ITIL de todos los niveles del personal. También se habilitaron canales de retroalimentación para ajustar el modelo según las necesidades emergentes de los ciudadanos y del equipo técnico.

Para la tesis, este artículo resulta especialmente valioso porque traslada el modelo ITIL a un entorno no empresarial, demostrando su aplicabilidad en escenarios diversos como la administración pública. Muestra cómo una mesa de ayuda puede profesionalizarse y convertirse en un instrumento estratégico, capaz de ofrecer valor real al usuario, ya sea ciudadano, funcionario o entidad. El hecho de que los mismos procesos —gestión de incidentes, niveles de atención, SLAs, base de conocimiento— se implementen con éxito fuera del sector privado, valida su utilidad universal.

Asimismo, refuerza la idea de que ITIL es un marco adaptable, y que no necesita ser implantado en su totalidad para generar mejoras. La entidad del estudio comenzó con los procesos más críticos y expandió gradualmente el alcance, combinando automatización, capacitación, monitoreo e integración con otras áreas de TI. Esto respalda la recomendación

de la tesis de implementar el modelo de forma incremental, alineando cada etapa con las capacidades existentes de la organización.

Por último, el artículo da evidencia de que el éxito de una mesa de ayuda no depende exclusivamente de herramientas tecnológicas, sino de procesos, roles, cultura y gobernanza. Esta visión es completamente coherente con el modelo propuesto en la tesis, que busca articular ITIL en la mesa de ayuda y Scrum en el desarrollo, con una interfaz de colaboración efectiva, retroalimentación constante y foco en el valor para el usuario.

2.3 Bases Legales

2.3.1 Ley N° 28051 - Ley de Prestaciones Alimentarias en Beneficio de los Trabajadores Sujetos al Régimen Laboral de la Actividad Privada

La Ley N.° 28051 regula el régimen de prestaciones alimentarias otorgadas a trabajadores sujetos al régimen laboral de la actividad privada. Esta norma establece las condiciones bajo las cuales pueden entregarse vales, cupones u otros mecanismos destinados a la adquisición de alimentos, así como sus límites y efectos laborales o tributarios. En el contexto de esta investigación, dicha ley resulta relevante porque Kred C&S participa en el sector de soluciones de beneficios corporativos, donde la gestión tecnológica debe asegurar continuidad, trazabilidad y calidad en servicios asociados a beneficios para colaboradores (Ley N.° 28051, 2003).

Estas empresas operan como emisores autorizados de instrumentos de valor no dinerario que pueden ser usados exclusivamente en comercios afiliados para la adquisición de alimentos o productos específicos.

Ámbitos que regula la ley:

- Definición de vales
    - Vale de alimentación: Medio de pago entregado por el empleador al trabajador para compra de alimentos.
    - Vale de consumo: Similar, pero aplicable a una gama más amplia de productos, no solo alimentación.
- Forma de entrega
    - Pueden ser físicos o electrónicos (tarjetas, códigos, etc.)
    - No son parte de la remuneración regular ni base de cálculo de beneficios laborales.
- Uso permitido
    - Solo pueden ser usados en comercios registrados ante la SUNAT como parte de la red de aceptación de la empresa emisora.
- Exclusión del cálculo de beneficios

- No se considera para el cálculo de CTS, vacaciones, gratificaciones ni aportes a la seguridad social.

- Condiciones tributarias
    - Las empresas pueden deducir los vales entregados como gasto tributario, siempre que se cumpla con los topes legales y el sistema de emisión autorizado.

2.3.2 Ley N° 29733 - Ley de Protección de Datos Personales

Dentro del marco jurídico peruano, la Ley N.° 29733 reconoce la protección de los datos personales como un derecho fundamental y establece obligaciones para las entidades públicas y privadas que los recopilan, utilizan, almacenan o transfieren. Este régimen también puede comprender a responsables o encargados ubicados fuera del territorio nacional cuando dirigen bienes o servicios a personas situadas en el Perú o realizan seguimiento de su comportamiento, conforme al alcance establecido por la ley y su reglamento vigente (Decreto Supremo N.° 016-2024-JUS, 2024; Ley N.° 29733, 2011).

Los principios que guían el tratamiento de datos personales son: legalidad, consentimiento, finalidad, proporcionalidad, calidad, seguridad, disposición de recurso y nivel de protección adecuado para transferencias internacionales. Estos principios se reflejan en controles técnicos y organizativos y deben evidenciarse en políticas, contratos y registros.

2.3.3 Ley N° 30096 - Ley de Delitos Informáticos

Dentro del marco jurídico peruano, se constituye ley N.° 30096 para enfrentar conductas ilícitas ejecutadas mediante tecnologías de la información y la comunicación. Su alcance comprende acciones que comprometen sistemas o datos informáticos, así como otros bienes jurídicos protegidos, e incluye supuestos relacionados con la vulneración de la confidencialidad, la integridad o la disponibilidad de la información y con distintas modalidades de fraude informático (Ley N.° 30096, 2013).

Entre sus núcleos de la norma destacan el acceso ilícito (incluido el exceso de privilegios), los atentados contra la integridad de datos y de sistemas, la interceptación de transmisiones no públicas, el fraude informático (con supuestos como la suplantación de interfaces o páginas y la colaboración dolosa para transferencias patrimoniales), la suplantación de

identidad, y el abuso de dispositivos diseñados para delinquir. Contiene además una exención de responsabilidad para pruebas de seguridad autorizadas (por ejemplo: pentesting) debidamente acordadas y prevé agravantes cuando interviene una organización criminal, abusando de una posición privilegiada, y se afectan servicios esenciales o media el uso de inteligencia artificial.

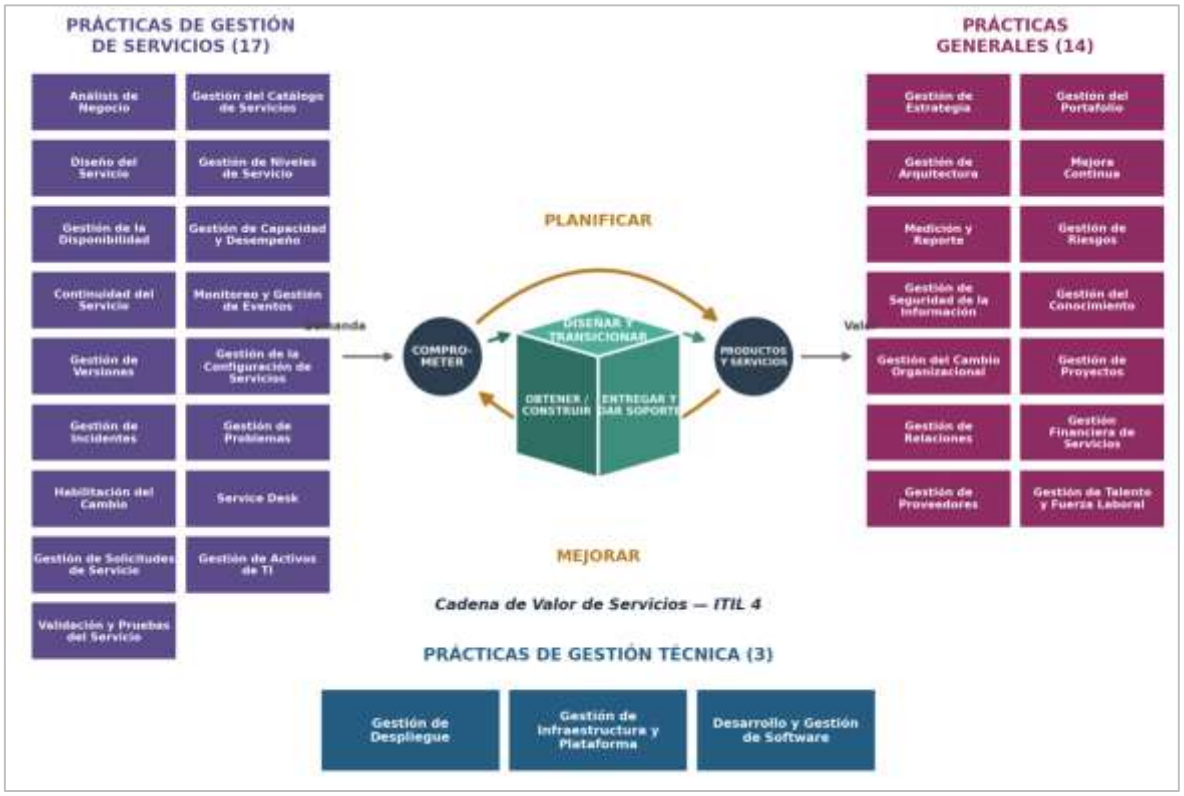
2.4 Marco Metodológico

2.4.1 ITIL – Gestión de Servicios de TI

De acuerdo con AXELOS (2019), ITIL 4 organiza la gestión de servicios mediante el Sistema de Valor del Servicio, estructura que articula la gobernanza, los principios guía, las prácticas y las actividades de la cadena de valor para convertir la demanda y las oportunidades en resultados valiosos. En esta investigación, dicho marco sustenta la coordinación entre la mesa de ayuda y la fábrica de software; su integración con Scrum y con mecanismos de automatización permite acortar los ciclos de atención y entrega, fortalecer la retroalimentación y disminuir los fallos asociados a los cambios.

Figura 6

Las 34 prácticas de ITIL 4 en relación con las actividades de la Cadena de Valor de Servicios



Nota. Elaborado a partir de ITIL® 4: Foundation Edition, por AXELOS, 2019.

- Principios Guías de ITIL v4.0

Según AXELOS (2019), las decisiones y acciones de mejora en ITIL 4 deben guiarse por siete criterios: concentrarse en el valor; partir de la situación existente; avanzar de forma iterativa con retroalimentación; colaborar y transparentar el trabajo; comprender el servicio de manera integral; mantener soluciones simples y prácticas; y optimizar antes de automatizar. En el modelo propuesto, estos principios evitan una aplicación rígida del marco y permiten ajustar las prácticas a las capacidades reales de la mesa de ayuda y de la fábrica de software.

Estos principios permiten que la organización actúe con agilidad, aprenda constantemente y mejore progresivamente sus procesos y servicios. En el caso de la mesa de ayuda, orientan a los equipos a responder con rapidez a las necesidades del usuario, reduciendo desperdicios y mejorando la experiencia del cliente.

- El Sistema de Valor de Servicios (SVS)

ITIL v4 introduce el Sistema de Valor del Servicio (Service Value System - SVS), un marco integral que describe cómo todos los componentes y actividades de una organización trabajan de manera conjunta para facilitar la creación de valor. El SVS incluye:

  - Principios guía.
  - Gobernanza.
  - Cadena de valor del servicio (Service Value Chain).
  - Prácticas de gestión.
  - Mejora continua.

Este modelo permite a las organizaciones adaptar y combinar prácticas según su contexto específico, promoviendo así una gestión más modular y efectiva (Torres & Luna, 2020).

- La Cadena de Valor del Servicio

La Cadena de Valor del Servicio es el núcleo del SVS. Está compuesta por seis actividades clave que permiten transformar la demanda en valor tangible: planear, mejorar, involucrar, diseñar y transicionar, obtener/construir y entregar y respaldar. Cada una de estas actividades se puede integrar de manera flexible para formar flujos de valor específicos para cada tipo de servicio, como la gestión de incidencias o el desarrollo ágil de software.

- Comparación con versiones anteriores

A diferencia de ITIL v3, centrado en procesos y en un ciclo de vida del servicio estructurado (estrategia, diseño, transición, operación y mejora continua), ITIL v4 adopta una visión más adaptable y holística, que permite integrar marcos ágiles como Scrum, Lean, DevOps y SAFe. Esta evolución resulta fundamental en entornos donde la agilidad, la colaboración interfuncional y la entrega continua de valor son factores determinantes del éxito (Smyth, 2020).

- Enfoque basado en servicios y mejora continua

ITIL v4 conserva la esencia de sus versiones anteriores respecto al enfoque en la prestación de servicios, pero lo redefine desde una perspectiva de co-creación de valor. El service desk o mesa de ayuda sigue siendo una función clave, no solo como punto de contacto, sino como un habilitador del valor, gestionando de manera eficiente los incidentes, solicitudes y requerimientos de los usuarios (Marrone & Hammerle, 2021).

Además, se promueve una cultura de mejora continua mediante el modelo conocido como continual improvement model, que consiste en siete pasos: ¿Qué se quiere lograr?, ¿Dónde estamos?, ¿A dónde queremos llegar?, ¿Cómo llegamos?, ¿Actuamos?, ¿Llegamos?, ¿Cómo mantener el momento? Este modelo es transversal a todo el sistema y resulta clave para lograr resultados sostenibles.

2.4.2 Practicas Relevantes de ITIL v4 en la Gestión de Servicios

ITIL v4 incorpora un conjunto de 34 prácticas que abarcan aspectos generales, técnicos y de gestión del servicio. Estas prácticas reemplazan a los antiguos procesos definidos en versiones anteriores, ofreciendo un enfoque más dinámico, modular y adaptable. A continuación, se describen las prácticas más relevantes para la gestión de servicios de TI, especialmente aplicables en el contexto de una mesa de ayuda y una fábrica de software:

- Gestión de Incidentes

Tiene como objetivo restaurar un servicio interrumpido o afectado lo más rápido posible, minimizando el impacto en las operaciones del negocio. Se prioriza según urgencia y severidad, y se apoya en herramientas de automatización y categorización.

- Gestión de Problemas

Busca identificar y eliminar las causas raíz de incidentes recurrentes, previniendo su futura aparición. Utiliza técnicas como el análisis causa-raíz (RCA) y diagramas de Ishikawa. Incluye la creación de errores conocidos y soluciones provisionales (workarounds).

- Gestión del Conocimiento

Consiste en recopilar, organizar y compartir información relevante para que los equipos técnicos y los usuarios accedan a soluciones rápidas. Implementa bases de conocimiento (knowledge bases) y FAQs integradas a la mesa de ayuda.

- Gestión del Cambio

Controla y facilita los cambios en la infraestructura tecnológica, reduciendo riesgos operativos. Se diferencia entre cambios estándar, normales y de emergencia, y trabaja en coordinación con los equipos de desarrollo.

- Gestión de Niveles de Servicio (SLAs)

Define, documenta y monitorea acuerdos de nivel de servicio entre el proveedor de TI y sus clientes. Establece métricas clave como tiempo de resolución, disponibilidad y calidad, para asegurar la alineación con las expectativas del negocio.

- Gestión de la Relación

Fomenta una comunicación efectiva con los usuarios, garantizando que sus necesidades y percepciones sean comprendidas. Enlaza con prácticas de experiencia del usuario (UX) y gestión de la demanda.

- Mejora Continua

Basada en el modelo de siete pasos, asegura que todos los servicios y prácticas evolucionen con el tiempo. Impulsa la revisión periódica de KPIs, el análisis de tendencias y la implementación de acciones correctivas y preventivas.

2.4.3 Ceremonias y Prácticas Operativas en ITIL v4 para la Mesa de Ayuda

En el entorno de una mesa de ayuda, las prácticas operativas y las rutinas estructuradas son fundamentales para asegurar un servicio eficiente, estandarizado y centrado en el usuario. ITIL v4 promueve un conjunto de actividades recurrentes que permiten gestionar eficazmente las interacciones con los usuarios:

- Registro y Clasificación de Solicitudes

Todas las solicitudes deben registrarse con datos estructurados: tipo de solicitud, usuario, prioridad, descripción y categoría.

- Priorización y Enrutamiento Automático

Mediante reglas de negocio, las solicitudes son clasificadas por criticidad y derivadas al grupo más adecuado.

- Escalamiento Funcional y Jerárquico

Cuando no puede resolverse un caso en primer nivel, se escala a niveles técnicos o de decisión superiores.

- Comunicación Activa con el Usuario

El usuario debe estar informado durante todo el proceso, mejorando su percepción del servicio.

- Cierre Documentado del Caso

Se registra la solución aplicada y se realiza una encuesta de satisfacción.

- Ceremonias Internas de la Mesa de Ayuda

Se incorporan reuniones diarias, revisión de casos y sesiones de feedback para fomentar mejora continua.

- Revisión de KPIs y Análisis de Tendencias

Se analizan indicadores como MTTR, tasa de reapertura, CSAT y volumen de tickets por categoría.

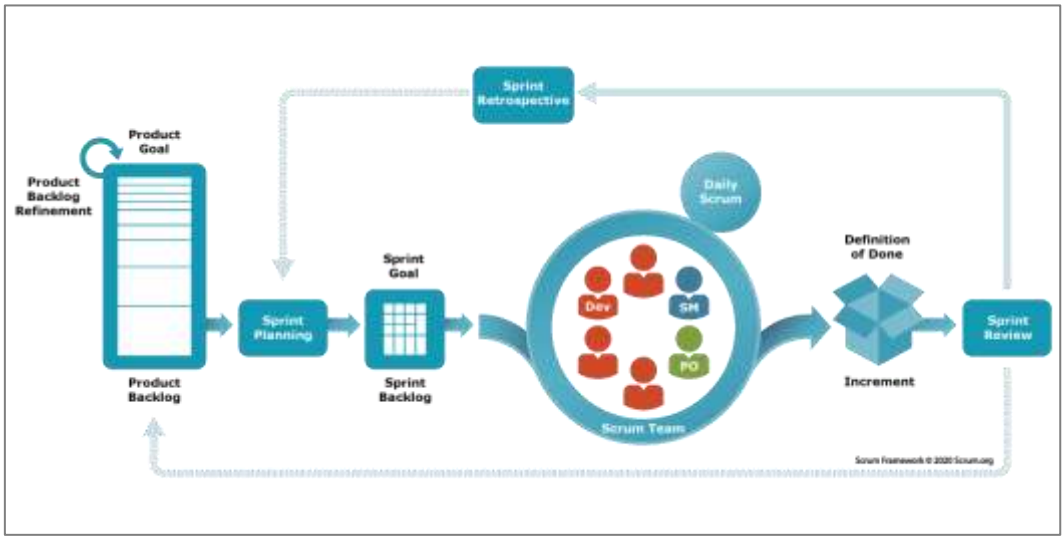
2.4.4 Scrum (marco ágil)

Marco ligero para gestionar trabajo complejo mediante roles (Product Owner, Scrum Master, Developers), eventos (Sprint Planning, Daily, Review, Retrospective) y artefactos (Product/Sprint Backlog, Increment) con transparencia, inspección y adaptación. Permite la entrega iterativa de valor, la priorización y la gestión del riesgo por ciclos cortos.

A diferencia de los modelos secuenciales, Scrum promueve la entrega incremental de valor a través de iteraciones cortas llamadas sprints, que suelen durar entre una y cuatro semanas. Cada sprint entrega un incremento funcional del producto, evaluable por los usuarios o stakeholders. Este enfoque permite reducir riesgos, validar decisiones tempranamente y adaptarse con rapidez a los cambios de contexto o requerimientos.

Scrum se alinea con los principios del Manifiesto Ágil (2001), que valora más la colaboración con el cliente, el software funcionando, la respuesta al cambio y la interacción entre personas, que los contratos rígidos, la documentación exhaustiva y los planes inflexibles (Beck et al., 2001). Este paradigma ha sido ampliamente adoptado por organizaciones de diversos sectores, y ha demostrado su eficacia especialmente en entornos de innovación tecnológica, como empresas que ofrecen soluciones digitales de bienestar y beneficios laborales.

Figura 7
Marco de trabajo Scrum: roles, eventos y artefactos del ciclo de Sprint



Nota. De The Scrum Framework Poster [Infografía], por Scrum.org, 2020 (https://www.scrum.org/resources/scrum-framework-poster).

En el contexto de una fábrica de software, Scrum permite organizar a los equipos en torno a objetivos de negocio concretos, mantener un ritmo de entrega constante y asegurar que las funcionalidades desarrolladas generen valor real para el usuario. Esto es especialmente útil cuando los requerimientos evolucionan rápidamente, como sucede en entornos empresariales donde los servicios deben ajustarse a las expectativas dinámicas de empleados, clientes o socios comerciales.

Scrum también facilita la integración con marcos de gobernanza como ITIL v4, ya que sus ciclos iterativos permiten sincronizar entregas con prácticas de gestión del cambio, validaciones de calidad, revisiones operativas y reportes de valor. Así, se genera una cultura de mejora continua, colaboración interfuncional y orientación al resultado.

- Roles en Scrum

Product Owner: Es el responsable de maximizar el valor del producto resultante del trabajo del equipo de desarrollo. Prioriza los elementos del Product Backlog en función del valor para el negocio y mantiene una comunicación constante con los stakeholders.

Scrum Master: Actúa como facilitador del marco Scrum. Se asegura de que el equipo entienda los valores, principios y prácticas de Scrum, remueve impedimentos y promueve la mejora continua.

Equipo de Desarrollo: Es un grupo autoorganizado y multidisciplinario encargado de entregar incrementos de producto que cumplan con la definición de “hecho”. Colabora estrechamente en cada iteración para alcanzar los objetivos del sprint.

- Artefactos

Product Backlog: Lista priorizada de funcionalidades, requerimientos y mejoras necesarias para el producto. Está en constante evolución.

Sprint Backlog: Conjunto de elementos del Product Backlog seleccionados para ser desarrollados en un sprint, junto con un plan para entregarlos.

Incremento: Resultado del trabajo completado durante el sprint, que cumple con los criterios de aceptación y la definición de hecho. Debe estar potencialmente listo para ser entregado al usuario.

- Ceremonias

Sprint Planning: Se realiza al inicio de cada sprint. El equipo colabora para definir qué elementos del Product Backlog se abordarán y cómo se realizará el trabajo. Esta reunión es esencial para alinear expectativas y establecer un objetivo de sprint claro.

Daily Scrum (Reunión diaria): Breve encuentro de 15 minutos donde el equipo sincroniza sus esfuerzos identifica obstáculos y ajusta su plan de trabajo. Esta ceremonia promueve la transparencia y la autogestión.

Sprint Review: Al finalizar el sprint, el equipo presenta el incremento a los stakeholders y recibe retroalimentación. Este evento fortalece la comunicación con el negocio y valida el avance en términos de valor entregado.

Sprint Retrospective: Espacio de reflexión donde el equipo analiza lo que funcionó bien, lo que podría mejorar y qué acciones adoptar para el próximo sprint. Esta ceremonia es clave para sostener una cultura de mejora continua y aprendizaje colectivo.

Estas ceremonias no solo estructuran el trabajo del equipo, sino que generan un ritmo constante de entrega y evaluación. En empresas que gestionan soluciones digitales para beneficios de empleados, estas ceremonias permiten adaptar las funcionalidades a nuevas políticas laborales, requerimientos legales o retroalimentación de los usuarios internos.

La disciplina con que se aplican estas ceremonias impacta directamente en la productividad, calidad del software y moral del equipo. Además, su regularidad crea un entorno predecible para el cambio, facilitando la integración con marcos como ITIL v4, especialmente en prácticas como la gestión del cambio, la mejora continua y la gestión de versiones (Pressman & Maxim, 2020; Schwaber & Sutherland, 2020).

- Indicadores Lean: Lead Time, Cycle Time y Métricas de Flujo

En el marco de la gestión ágil de servicios y desarrollo de software, los indicadores Lean como el Lead Time y el Cycle Time son fundamentales para comprender la eficiencia y efectividad del flujo de trabajo. Estos indicadores, originados en la manufactura esbelta (Lean Manufacturing), han sido adaptados al contexto de TI para medir tiempos de entrega, detectar cuellos de botella y mejorar la predictibilidad del rendimiento del equipo (Kniberg & Skarin, 2010).

Lead Time se refiere al tiempo total que transcurre desde que un requerimiento es solicitado por un cliente o usuario hasta que se entrega completamente. Incluye todas las etapas del flujo, como espera, validación, desarrollo, pruebas y despliegue. Es un indicador centrado en la percepción del usuario final, ya que mide la capacidad del equipo de satisfacer una necesidad desde la demanda hasta el valor entregado.

Cycle Time en cambio, mide el tiempo activo de trabajo desde que un equipo comienza a trabajar sobre un ítem hasta que lo termina. Excluye los períodos de espera y se enfoca en la eficiencia interna del proceso. Una reducción del Cycle Time indica una mayor velocidad de producción y menor desperdicio en el flujo de trabajo.

Ambos indicadores permiten evaluar la capacidad de entrega del equipo y se utilizan ampliamente en tableros Kanban, gráficos de control y herramientas como Jira, Azure DevOps y Trello. Una correlación entre un Lead Time reducido y una mejora en el Cycle Time suele reflejar un proceso optimizado y balanceado.

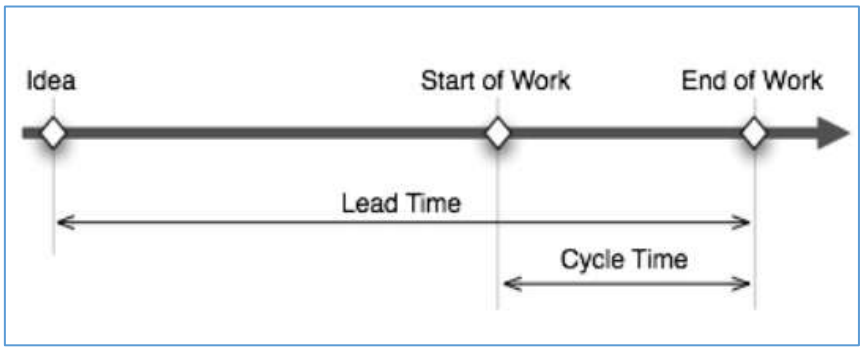
En el contexto de una mesa de ayuda, por ejemplo, el Lead Time de atención a un ticket desde su registro hasta su cierre puede reflejar la experiencia del usuario, mientras que el Cycle Time revela la eficiencia interna del soporte de primer y segundo nivel. Medir ambos indicadores permite detectar cuellos de botella, evaluar el impacto de interrupciones o cambios, y tomar decisiones basadas en datos para mejorar la calidad del servicio. Adicionalmente, otras métricas Lean relevantes incluyen:

- Work In Progress (WIP): Cantidad de tareas activas en proceso. Un WIP alto indica sobrecarga y potenciales demoras.
- Throughput: Número de ítems completados por unidad de tiempo. Mide la productividad del equipo.
- Tiempo de espera (Waiting Time): Lapso en que un trabajo está detenido antes de ser atendido. Impacta directamente en el Lead Time.

Al monitorear estas métricas, las organizaciones pueden realizar ajustes tácticos y estratégicos que permitan balancear la carga de trabajo, mejorar la fluidez y mantener un ritmo sostenible de entrega. Su integración con prácticas de ITIL v4 y Scrum fortalece la

capacidad de adaptación, la mejora continua y la entrega constante de valor al cliente (Mahnic, 2015; Redondo, 2019).

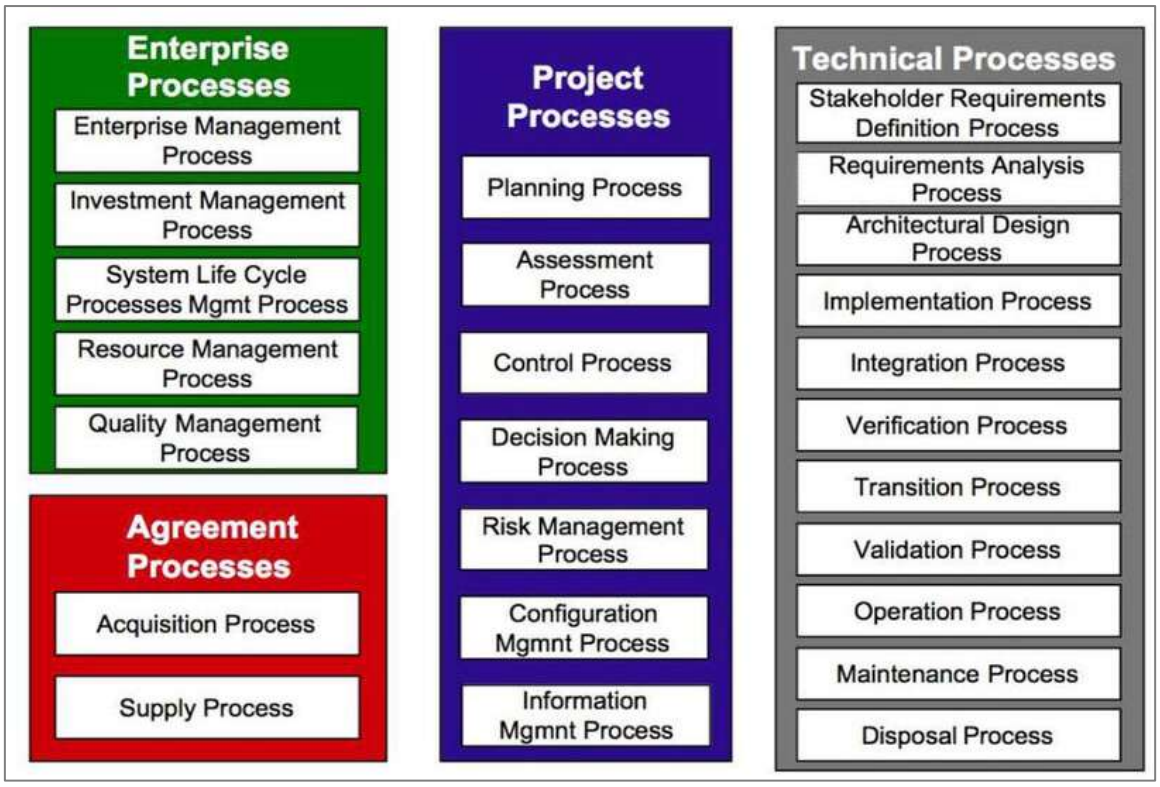
Figura 8
Métrica Aplicadas en ITIL y Desarrollo Ágil



2.4.5 ISO/IEC/IEEE 15288 (Procesos del ciclo de vida de sistemas)

Estándar de ingeniería de sistemas que abarca desde la definición de necesidades de stakeholders y requisitos de sistema hasta integración, verificación, validación, transición, operación y retiro, con procesos organizacionales y de habilitación (gestión, soporte, acuerdos). Justifica cómo la ISO 15288 gobierna el sistema completo y cómo la ISO 12207 se integra como vista de software.

Figura *9*
Clasificación de los procesos del ciclo de vida del sistema según ISO/IEC/IEEE 15288: Procesos Empresariales



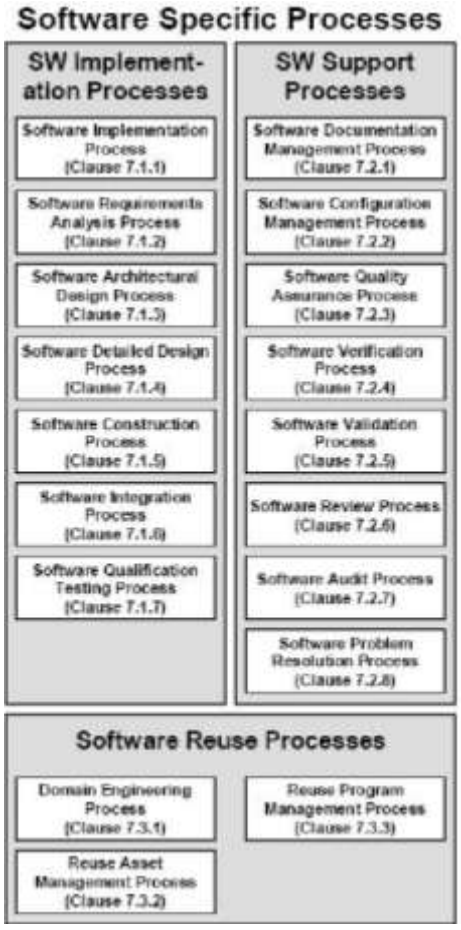
Nota. Elaborado a partir de ISO/IEC/IEEE 15288:2023 — Systems and software engineering — System life cycle processes, por International Organization for Standardization, International Electrotechnical Commission e Institute of Electrical and Electronics Engineers, 2023.

2.4.6 ISO/IEC 12207 (Procesos del ciclo de vida del software).

Estándar que define procesos y actividades para gestionar e implementar software a lo largo de su ciclo de vida (acuerdo, gestión, requisitos, diseño, construcción, verificación/validación, integración, entrega, operación y mantenimiento). Permite evidenciar entradas/salidas y puntos de control por fase vincularlo a artefactos y justificar cómo estos fluyen hacia la operación.

Figura 10

Procesos específicos de software según ISO/IEC 12207: implementación, soporte y reutilización de software



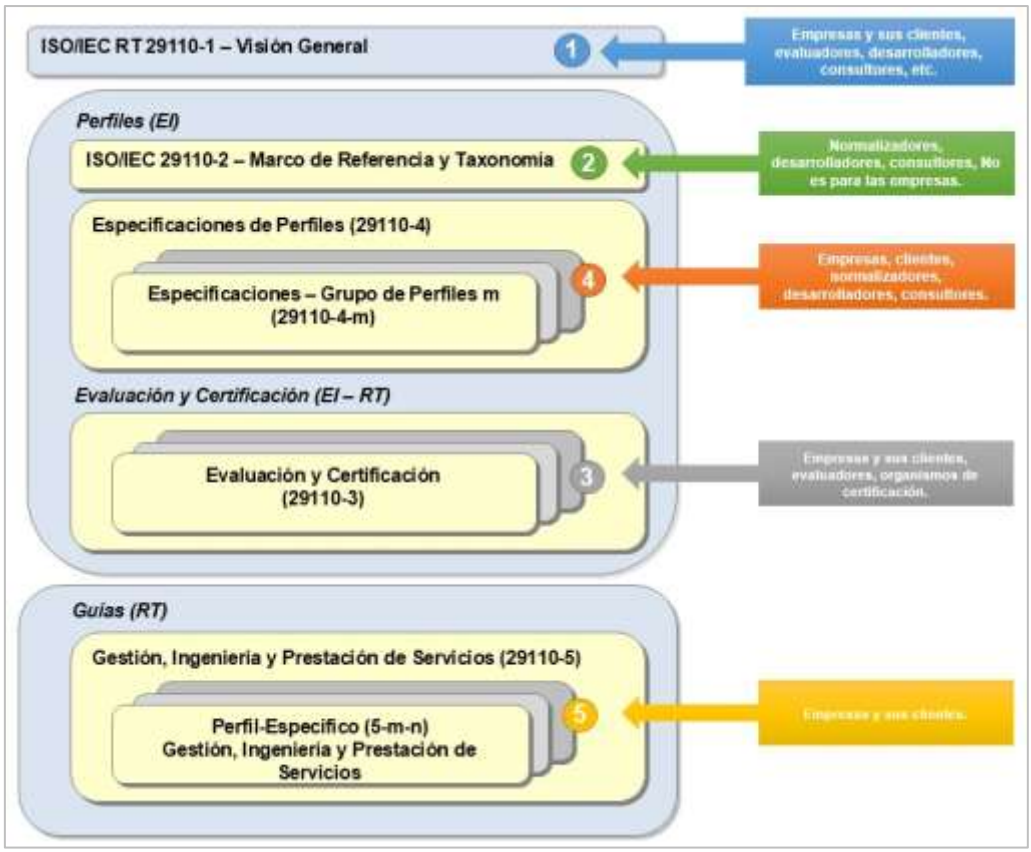
Nota. Adaptado de ISO/IEC 12207:2008 — Systems and software engineering — Software life cycle processes, por International Organization for Standardization e International Electrotechnical Commission, 2008.

2.4.7 ISO/IEC 29110 (VSE – Very Small Entities).

Conjunto de estándares para pequeñas y medianas empresas que define perfiles y procesos de desarrollo de software, incluyendo un perfil básico que puede ser implementado con métodos ágiles. Centrado en dos procesos esenciales: Gestión de Proyectos e Implementación de Software (planificación, requisitos, diseño/codificación, pruebas y entrega).

Figura 11

Arquitectura documental de la serie ISO/IEC 29110: perfiles, evaluación y certificación, y guías para Pequeñas Organizaciones (VSE)



Nota. Adaptado de ISO/IEC TR 29110-1:2016 — Systems and software engineering — Lifecycle profiles for Very Small Entities (VSEs) — Part 1: Overview, por International Organization for Standardization e International Electrotechnical Commission, 2016.

2.4.8 ISO/IEC 25010 (Calidad del producto software).

Para evaluar la calidad del producto software, la investigación adopta el modelo de ISO/IEC 25010:2023, compuesto por nueve características que se desagregan en subcaracterísticas verificables. Esta estructura permite convertir las expectativas de los interesados en requisitos no funcionales, criterios de aceptación y evidencias de prueba, facilitando la trazabilidad entre la necesidad, la validación y el resultado obtenido. La Tabla 3 corresponde a una adaptación del modelo para el contexto del proyecto (International Organization for Standardization [ISO] & International Electrotechnical Commission [IEC], 2023).

Tabla 3
ISO/IEC 25010

SOFTWARE PRODUCT QUALITY								
FUNCTIONAL SUITABILITY	**PERFORMANCE EFFICIENCY**	**COMPATIBILITY**	**INTERACTION CAPABILITY**	**RELIABILITY**	**SECURITY**	**MAINTAINABILITY**	**FLEXIBILITY**	**SAFETY**
FUNCTIONAL COMPLETENESS	TIME BEHAVIOUR	CO-EXISTENCE	APPROPRIATENESS RECOGNIZABILITY	FAULTLESSNESS	CONFIDENTIALITY	MODULARITY	ADAPTABILITY	OPERATIONAL CONSTRAINT
FUNCTIONAL CORRECTNESS	RESOURCE UTILIZATION	INTEROPERABILITY	LEARNABILITY	AVAILABILITY	INTEGRITY	REUSABILITY	SCALABILITY	RISK IDENTIFICATION
FUNCTIONAL APPROPRIATENESS	CAPACITY		OPERABILITY	FAULT TOLERANCE	NON-REPUDIATION	ANALYSABILITY	INSTALLABILITY	FAIL SAFE
			USER ERROR PROTECTION	RECOVERABILITY	ACCOUNTABILITY	MODIFIABILITY	REPLACEABILITY	HAZARD WARNING
			USER ENGAGEMENT		AUTHENTICITY	TESTABILITY		SAFE INTEGRATION
			INCLUSIVITY USER ASSISTANCE		RESISTANCE			
			SELF-DESCRIPTIVENESS					

Nota. Elaborado a partir de *ISO/IEC 25010:2023 — Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — Product quality model*, por International Organization for Standardization e International Electrotechnical Commission, 2023.

2.4.9 Gestión de Proyectos – PMBOK (PMI).

Estándar que organiza buenas prácticas en dominios como alcance, cronograma, costos, calidad, recursos, riesgos y stakeholders, adaptable a enfoques predictivos, ágiles o híbridos. Usado para estructurar el plan del proyecto (EDT/WBS, cronograma, gestión de riesgos y comunicaciones) y para demostrar control metodológico en la ejecución.

2.4.10 Organizational Project Management Maturity Model - OPM3 (PMI).

Se define como el conjunto de procesos y decisiones para seleccionar, priorizar, balancear y gobernar iniciativas (proyectos y programas) de modo que maximicen el valor estratégico dentro de las restricciones de presupuesto y capacidad organizacional. Esto implica definir criterios de priorización (valor, riesgo, urgencia, cumplimiento), establecer límites y políticas de balance (run–grow–transform, riesgo/retorno), implantar un ciclo de autorizaciones por etapas (stage-gates) y medir la entrega de beneficios con indicadores asegurando trazabilidad desde la estrategia hasta los resultados operativos.

Proporciona un marco para evaluar y mejorar la madurez con la que una organización ejecuta su estrategia a través de proyectos, programas y portafolios (PPP). Se apoya en la relación entre buenas prácticas => capacidades => resultados => KPI, y propone un camino de mejora continua que avanza por estandarizar => medir => controlar => mejorar, habilitado por la gobernanza, las personas y la tecnología. Permite diagnosticar brechas, priorizar iniciativas y demostrar la trazabilidad desde objetivos estratégicos hasta resultados medibles y sostenibles.

3. Situación Actual

Kred C&S ofrece soluciones ágiles en alimentación, incentivos y movilidad, posicionándose como el aliado ideal de las áreas de recursos humanos de las empresas que buscan aumentar el poder adquisitivo, mejorar el bienestar y la calidad de vida de sus colaboradores. Kred C&S cuenta con una amplia red de clientes y comercios afiliados en todo el territorio nacional. La empresa se enfoca en el mercado B2B2C, su cartera de productos y servicios abarca desde soluciones tradicionales, como tarjetas de alimentación, hasta plataformas digitales para la gestión de beneficios e incentivos.

3.1 Misión

Brindar soluciones innovadoras de beneficios y engagement para empleados, optimizando la gestión de recursos humanos y mejorando la calidad de vida de los colaboradores de las empresas clientes, garantizando que los beneficios ofrecidos se adapten a las necesidades de cada organización y contribuyan al desarrollo profesional y personal de los empleados.

3.2 Visión

Ser líder en el mercado de beneficios y engagement en Perú, ofreciendo soluciones tecnológicas avanzadas que generen impacto positivo en la productividad y bienestar de los empleados. Buscamos consolidarnos como un referente en innovación, confiabilidad y excelencia en la gestión de beneficios corporativos.

3.3 Objetivos Estratégicos

Kred C&S ha establecido un conjunto de objetivos estratégicos alineados con su misión de mejorar la calidad de vida de los colaboradores y promover el bienestar en las organizaciones. A continuación, se detallan estos objetivos:

Tabla 4
Objetivos Estratégicos Kred C&S



- Acelerar la Oferta de Beneficios para Colaboradores:

  Kred C&S busca fortalecer su línea de negocio enfocándose en soluciones que abordan necesidades clave como alimentación, bienestar, regalos y movilidad. Esto incluye productos como Tarjeta de Alimentación, Tarjeta de Incentivo, Tarjeta de Gasolina y Tarjeta de Regalo, diseñados para mejorar el poder adquisitivo y la satisfacción de los trabajadores.
- Enfocarse en Pequeñas y Medianas Empresas (Pymes):

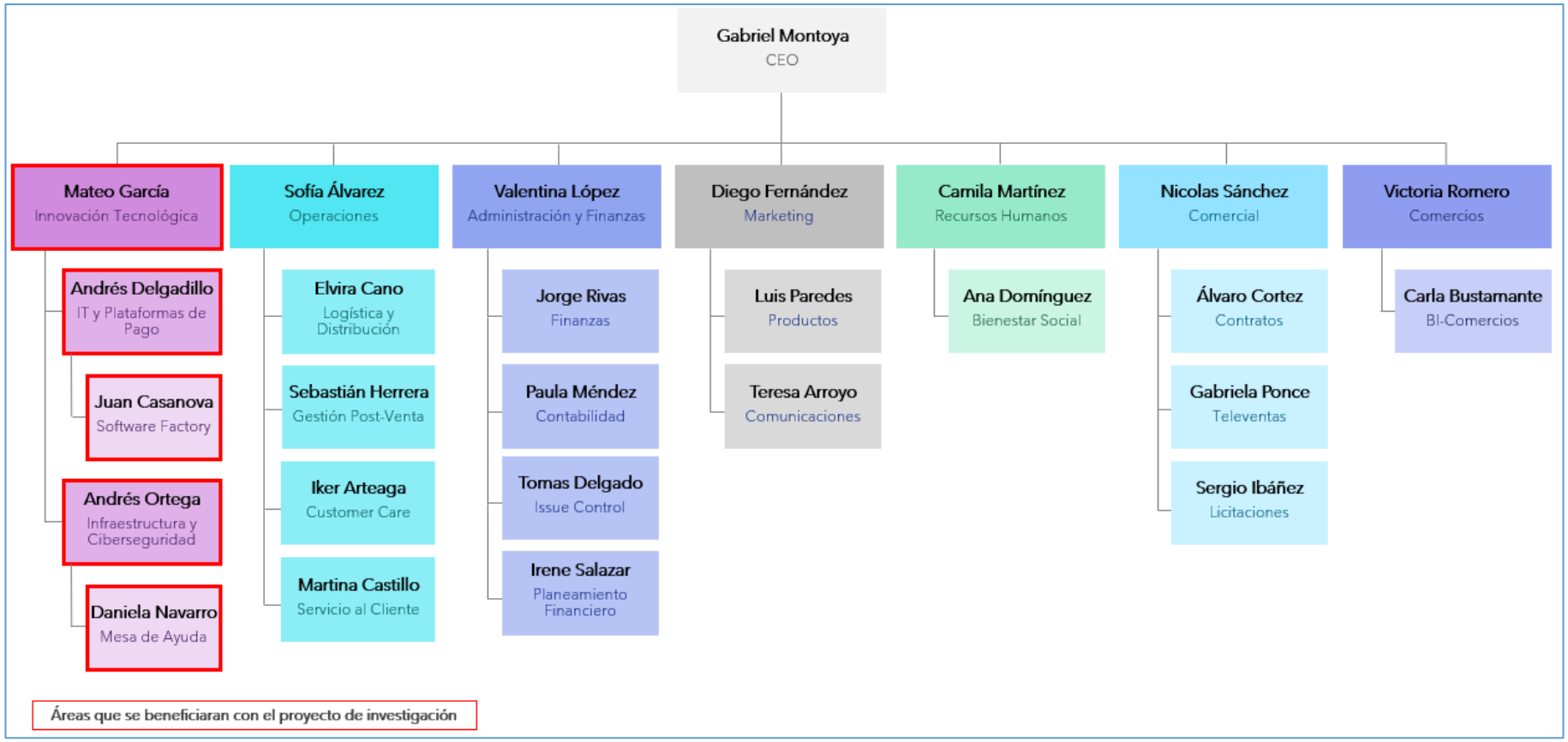
  Kred C&S tiene como objetivo aumentar su penetración en el segmento de las Pymes, reconociendo la importancia de cultivar un ambiente laboral positivo y motivador en estas organizaciones.
- Impulsar la Estratega Digital:

  Kred C&S se compromete a mejorar la experiencia de sus clientes y beneficiarios mediante la implementación de soluciones digitales innovadoras. Esto incluye el desarrollo de una aplicación móvil que ofrece beneficios y ofertas ampliadas para los colaboradores, así como una plataforma de autogestión para empresas denominada “Kred Clientes”.

- Promover la Sostenibilidad Ambiental:
  Como parte de su responsabilidad social y ambiental, Kred C&S ha lanzado tarjetas fabricadas al 100% con plástico reciclado, siendo pionera en esta iniciativa en el país.
- Mejorar la Calidad de Vida de los Colaboradores:
  La empresa se centra en ofrecer soluciones que incrementen la satisfacción laboral, impulsen el desarrollo profesional y mejoren el compromiso de los equipos de trabajo.

3.4 Organigrama

Figura 12
Organigrama Kred C&S



- CEO

  Es el máximo responsable de la gestión integral de la empresa en el país. Define y ejecuta la estrategia local alineada con los objetivos definidos a nivel regional (Latam) y Global, asegurando el cumplimiento de objetivos financieros, comerciales, operativos y de sostenibilidad.

- Sistemas

  Es el responsable regional de la gestión tecnológica, abarcando infraestructura, desarrollo de plataformas, ciberseguridad y proyectos digitales. Dirige la estrategia tecnológica para garantizar que las soluciones digitales respondan a las necesidades del negocio. Supervisa las jefaturas de IT, buscando eficiencia operativa, innovación y alineamiento con la estrategia digital global de la empresa.

  - Infraestructura y Ciberseguridad

    Encargada de asegurar la continuidad tecnológica de las operaciones. Administra servidores, redes, alojamiento en la nube, estaciones de trabajo y sistemas de respaldo y de seguridad. Además, lidera al interno la estrategia de ciberseguridad, protegiendo la información, controlando accesos y gestionando amenazas, así como también gestiona la mesa de ayuda.

  - IT y Plataformas de Pago

    Responsable del desarrollo, mantenimiento y evolución de las plataformas digitales de la empresa. Gestiona la fábrica de software y su roadmap, inicia la priorización de requisitos de software y alinea el uso de tecnologías en función a la arquitectura All-in-One definido por su sede central. Además, coordina con entidades financieras como Unibanca (VISA) asegurando la correcta integración de servicios.

- Operaciones

  Supervisa los procesos clave del negocio: desde la gestión de pedidos de tarjetas hasta su emisión, carga y distribución. Asegura el cumplimiento de los tiempos de entrega de las tarjetas de beneficios, la calidad del servicio al cliente y la eficiencia logística. Coordina con TI para asegurar la disponibilidad de los sistemas tanto para clientes como para el equipo de operaciones, asimismo la comunicación con UNIBANCA para garantizar la emisión de las tarjetas de beneficios y su entrega correcta a las empresas o a los beneficiarios en caso soliciten reposición a sus tarjetas.

- Administración y Finanzas

  Administra los recursos financieros de la empresa. Controla el presupuesto, gestiona cuentas por pagar y cobrar, lidera los cierres contables y asegura el correcto cumplimiento de normativas fiscales.

- Recursos Humanos

  Encargada de atraer, retener y desarrollar el talento en la empresa. Diseña e implementa políticas de bienestar, planes de carrera, formación, clima laboral y evaluación de desempeño.

- Marketing

  Diseña e implementa campañas de marketing enfocadas en el posicionamiento de empresa como marca innovadora y proveedora de beneficios tanto para empresas como para sus trabajadores.

- Comercial

  Es responsable de captar y fidelizar clientes empresariales con enfoque en grandes empresas y estratégicamente en pequeñas y medianas empresas (Pymes) y en los últimos años en el sector público.

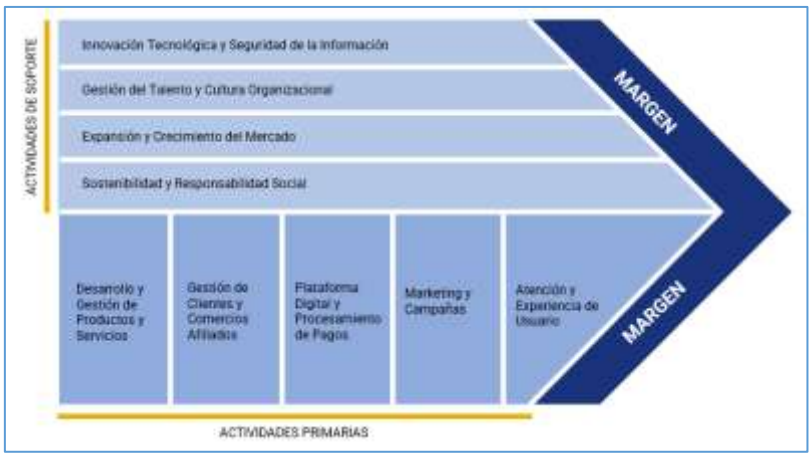
- Comercios

  Responsable de la red de comercios afiliados con los cuales se acepta las tarjetas de beneficios, así como de los convenios para las promociones que se hace disponible a los beneficiarios por medio de cuponeras o por notificaciones en la app de la empresa.

3.5 Cadena de Valor

La cadena de valor Kred C&S se estructura en actividades primarias y actividades de soporte, alineadas con su modelo de negocio B2B2C (empresa a empresa a consumidor), donde conecta clientes (empresas), consumidores (empleados) y comercios afiliados.

Figura 13
Cadena de Valor



3.5.1 Actividades Primarias (Generación Directa de Valor)

- Desarrollo y Gestión de Productos y Servicios:
    - Oferta de beneficios alineada con la normativa peruana.
        - Tarjeta de Alimentación (vales de alimentos con beneficios tributarios).
        - Tarjeta de Movilidad (vales de transporte y combustible).
        - Tarjeta de Bienestar (beneficios en salud, entretenimiento y formación).
        - Tarjeta de Incentivos (bonos y recompensas empresariales).
    - Digitalización progresiva: integración de soluciones móviles y pagos electrónicos.
    - Segmentación del mercado: enfoque en grandes empresas y pymes en crecimiento.
- Gestión de Clientes y Comercios Afiliados

    - Captación de empresas que buscan beneficios laborales exonerados de impuestos.
    - Expansión de la red de comercios en supermercados, restaurantes, bodegas, farmacias y gasolineras.
    - Convenios con gremios empresariales y asociaciones de pymes para fomentar el uso de beneficios.
- Plataforma Digital y Procesamiento de Pagos
    - Implementación de la All-in-One adaptada a medios de pago en Perú.
    - Integración con billeteras electrónicas locales (Google Pay, Apple Pay).
    - Uso de tarjetas prepago y códigos QR para transacciones rápidas en comercios afiliados.
    - Optimización del modelo financiero del Float (gestión del dinero cargado en cuentas de la empresa antes de su uso).
- Atención y Experiencia del Usuario
    - Customer Care y soporte omnicanal en español.
    - Capacitación a empresas y comercios afiliados para mejorar la experiencia de uso.

3.5.2 Actividades de Soporte (Infraestructura y Crecimiento)

- Sistemas y Seguridad de la Información
    - Aplicación de regulaciones locales de ciberseguridad y protección de datos (conforme a la Ley de Protección de Datos Personales).
    - Integración con sistemas contables peruanos para conciliación automática de pagos.
    - Implementación de procedimientos de ciberseguridad para la detección de fraudes en pagos.
- Gestión del Talento y Cultura Organizacional
    - Adopción de una cultura de bienestar para colaboradores.
    - Capacitación en tecnología, ventas y gestión comercial adaptada al mercado local.
- Expansión y Crecimiento del Mercado
    - Estrategia de penetración en pymes que actualmente tienen una baja adopción de beneficios laborales.

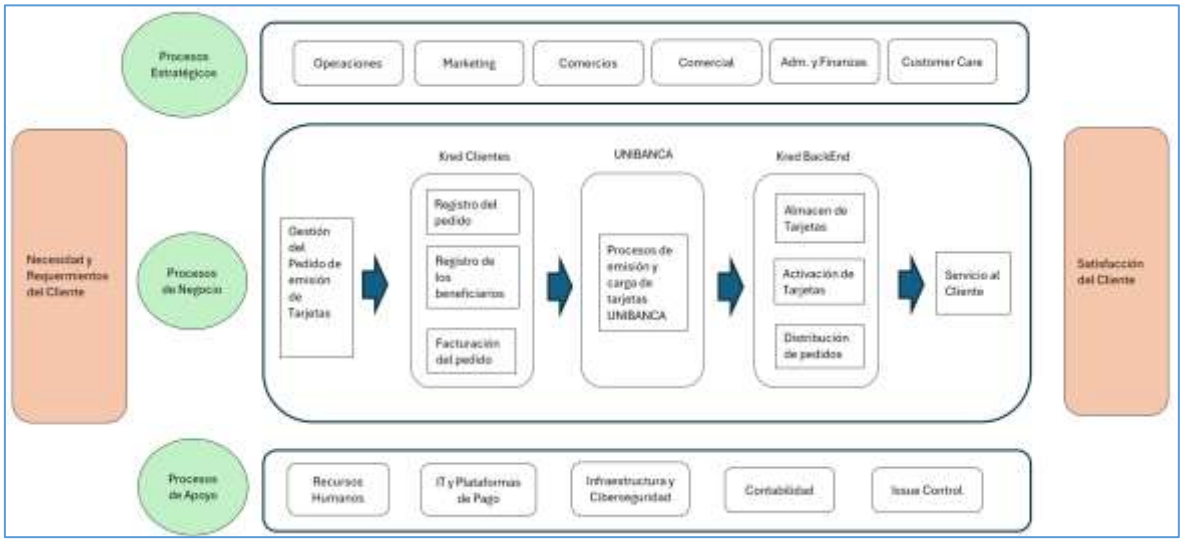
    - Alianzas estratégicas con bancos peruanos para fortalecer la oferta de tarjetas y billeteras digitales (Apple Pay, Google Pay).
- Sostenibilidad y Responsabilidad Social (ESG)
    - Uso de beneficios para programas de inclusión social y alimentación en alianza con el sector público.
    - Digitalización para reducir el uso de papel.
    - Utilización de material reciclado para la materialización de las tarjetas.

3.6 Mapa de Procesos

3.6.1 Macroproceso General (Gestión del Pedido de Tarjetas de Beneficiarios)

El siguiente mapa de proceso proporciona una visión estructurada del principal macroproceso de Kred C&S la cual engloba toda la gestión del pedido de tarjetas que realizan los clientes (empresas). La empresa se encuentra en constante innovación funcional con respecto a sus procesos de gestión de pedidos, por lo que la Gerencia de Sistemas participa brindando el soporte necesario en lo que respecta a su servicio de mesa de ayuda (Infraestructura y Ciberseguridad) así como por el servicio de fábrica de software (IT y Plataformas de Pago).

Figura 14
Mapa de Procesos – Macroproceso General



A continuación, se describe el mapa de procesos.

Procesos Estratégicos:

- Operaciones

  Este proceso supervisa la eficiencia y efectividad de las actividades de gestión de los pedidos de los clientes, asegurando que los servicios se entreguen conforme a los estándares de calidad y dentro de los plazos establecidos.

- Marketing

Proceso encargado de desarrollar e implementar estrategias para posicionar y asegurar la marca de la empresa, atraer nuevos clientes y retener a los existentes.

- Comercios

Proceso encargado de la interacción con la red de comercios afiliados en los cuales se mantiene convenio para la aceptación de las tarjetas Kred C&S.

- Comercial

Involucra la identificación y captación de nuevos clientes para la empresa, así como la gestión de relaciones con los clientes actuales ya que el ejecutivo comercial brinda un servicio de línea directa con los contactos del cliente para brindar un soporte funcional de nivel 1 a los pedidos de tarjetas que realizan, como por ejemplo de facturación, promociones, campañas, etc.

- Administración y Finanzas

Procesa encargado de la gestión financiera de la empresa, incluyendo la planificación presupuestaria, control de costos y aseguramiento de la rentabilidad la cual es informada directamente al CEO de la empresa.

- Customer Care

Proceso que se dedica a proporcionar soporte y asistencia a los clientes vía telefónica y correo electrónico, resuelve consultas y problemas relacionados principalmente a los pedidos registrados por los clientes y transacciones realizadas por los beneficiarios al utilizar sus tarjetas de beneficios buscando con ello asegurar una experiencia positiva.

Procesos de Negocio:

- Proceso de Gestión de Pedidos de Clientes (Kred Clientes)

Proceso que se encarga de recibir, validar y procesar las solicitudes de tarjetas de beneficios por parte de los clientes (empresas) asegurando que los pedidos se gestionen de manera eficiente y precisa.

- Proceso de Emisión y Carga de Pedidos (UNIBANCA)

Proceso que implica la producción física y digital de las tarjetas para los beneficiarios de la empresa que realizo el pedido. No solo se encarga de la emisión de la tarjeta sino también de la carga monetaria correspondiente indicado durante el proceso de gestión de pedidos del cliente. Asegura la integración con las entidades financiera y con VISA.

- Proceso de Almacenamiento y Entrega de Pedidos (Kred BackEnd)

Se encarga de la logística, distribución y entrega de las tarjetas a los clientes. Para los casos de los pedidos solicitados por los clientes se les entrega en sus oficinas asegurando que lleguen en tiempo y con la debida seguridad garantizando la integridad del producto de manera similar a como lo realiza una entidad bancaria con el delivery de tarjetas de crédito.

Procesos de Apoyo:

- Recursos Humanos

Proceso que gestiona la contratación, formación, desarrollo y retención del talento humano dentro de la empresa.

- IT Plataformas de Pago

Proceso que gestiona, mantiene y mejora la infraestructura tecnológica de la empresa, entre sus objetivos se encuentra desarrollar soluciones digitales que facilitan la operación y mejoran la experiencia del cliente (empresas) y el beneficiario (usuario de la tarjeta). Coordina con los diversos procesos los requisitos de software que implica cambios en la base de aplicaciones de la empresa principalmente en meses previos a la campaña navideña.

- Infraestructura y Ciberseguridad

Proceso que garantiza la seguridad de la información y la protección contra amenazas externas e internas, adicionalmente, mantiene la operatividad de los sistemas y plataformas tecnológicas acorde a la arquitectura All-in-One definida por la sede central de Kred C&S.

- Contabilidad

Proceso que gestiona los registros contables y asegura el cumplimiento de las normativas contables y fiscales tanto del país como lo solicitado por la sede central.

- Issue Control

Proceso que supervisa y gestiona cualquier problema o incidente que surja en las operaciones de la empresa, principalmente con el proceso de gestión de pedidos del cliente, implementando soluciones y medidas preventivas para evitar recurrencias, así como influyendo en cambios sobre las aplicaciones de software de la compañía.

3.6.2 Macroproceso General vs Objetivos Estratégicos

Tabla 5
Macroproceso General vs Objetivos Estratégicos

Objetivos / Procesos Generales	Acelerar la Oferta de Beneficios para Colaboradores	Enfocarse en Pequeñas y Medianas Empresas (Pymes)	Impulsar la Transformación Digital	Promover Sostenibilidad Ambiental	Mejorar la Calidad de Vida de los Colaboradores
Operaciones		X	X	X	
Marketing	X	X		X	X
Comercios	X	X		X	X
Comercial	X	X			X
Admin y Finanzas		X			
Customer Care	X				X
Proceso de Gestión de Pedidos de Clientes (Kred Clientes)		X	X		X
Proceso de Emisión y Carga de Pedidos (UNIBANCA)		X	X	X	X
Proceso de Almacenamiento y Entrega de Pedidos (Kred BackEnd)		X	X		X
Recursos Humanos		X			X
IT y Plataformas de Pago	X	X	X	X	X
Infraestructura y Ciberseguridad	X	X	X	X	X
Contabilidad		X			
Issue Control		X	X		

En base a la relación de los procesos del punto anterior identificamos que el proceso de “IT Plataformas de Pago” e “Infraestructura y Ciberseguridad” son los únicos alineados a los 5 objetivos estratégicos de la empresa. Esto demuestra la importancia transversal dentro de la compañía y el impacto que genera por lo siguiente:

- Se confirma la importancia de ambos procesos en la estrategia digital de la empresa.
- Evidencia el impacto en los servicios brindados al cliente y beneficiario al exponer soluciones tecnológicas acordes al negocio de la empresa y gestionando incidencias atendidas por una mesa de ayuda con tiempos de atención definidos por la gerencia. Optimiza los servicios de TI en función a la sostenibilidad ambiental reduciendo el uso de recursos físicos y consumiendo infraestructura de nube, adicionalmente se asegura que la emisión de tarjetas de beneficios se realice sobre productos reciclados.

- Apoya al resto de procesos en la aceleración de la oferta de beneficios, estrategias de captación de clientes y campañas de fidelización ofreciendo soluciones digitales según los requisitos de software que se le solicite.

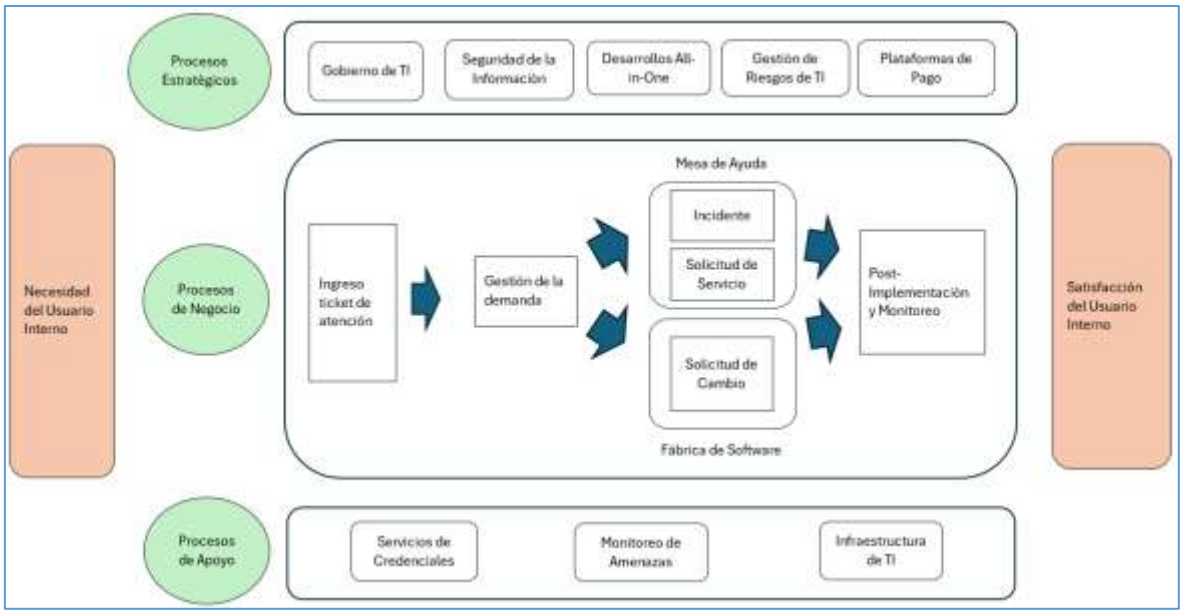
El siguiente mapa de procesos examina los procesos actuales del resultado del cruce de Macroprocesos vs Objetivos:

- Proceso de IT - Plataformas de Pago (Fábrica de Software - Gestión de Requisitos)
- Proceso de Infraestructura y Ciberseguridad (Mesa de Ayuda - Gestión de Incidencias)

3.6.3 Mapa de Procesos Específicos: IT - Plataformas de Pago e Infraestructura y Ciberseguridad

La gerencia de Sistemas de Kred C&S depende de la tecnología para gestionar el negocio por lo que para ellos es fundamental contar con una estructura de procesos de TI bien definida. A continuación, se presenta el Mapa de Procesos de las 2 áreas de TI alineado con la estrategia de la gerencia de Sistemas, dividido en Procesos Estratégicos, Clave y de Soporte.

Figura 15
Mapa de Procesos – Procesos Específicos de TI



A continuación, se describe el mapa de procesos.

Procesos Estratégicos:

- Gobierno de TI

  El proceso de Gobierno de TI establece las políticas, normas y procedimientos para garantizar que la tecnología esté alineada con los objetivos estratégicos de la empresa. Este proceso define el marco de trabajo para la toma de decisiones (comité de requisitos), asegurando que los recursos tecnológicos se usen de manera eficiente, con ello se logra optimizar la inversión en tecnología, reducir riesgos operativos y mejorar la calidad del servicio.

- Seguridad de la Información

El proceso de Seguridad de la Información protege los datos, sistemas y redes de la empresa contra amenazas internas y externas. En la empresa se manejan transacciones y datos sensibles de clientes, la seguridad de la información es fundamental para evitar fraudes, filtraciones de información y ataques.

- Desarrollos All-in-One

Como parte de la implementación de la arquitectura All-in-One desde la empresa matriz toda solución tecnológica debe estar alineada con esta plataforma. Este proceso verifica y valida que se cumpla con este principio y de ser necesario se integre con equipo de trabajo de la empresa a nivel regional y/o global.

- Gestión de Riesgos de TI

Este proceso identifica, analiza y mitiga los riesgos asociados no solamente a la infraestructura tecnológica si no al desarrollo de software. Se enfoca en la continuidad del negocio, la integridad de los sistemas y la disponibilidad de los servicios.

- Plataformas de Pago

El proceso de Plataformas de Pago permite la integración con una empresa de procesamiento, personalización y ensobrado de tarjetas con logo VISA (Unibanca) y principales servicios de recaudo del sistema financiero.

Procesos de Negocio:

- Gestión de Incidencias (Mesa de Ayuda)

El proceso de Mesa de Ayuda actúa como el primer punto de contacto para resolver incidencias, interrupciones, degradaciones del servicio y problemas tecnológicos dentro de la empresa. Actualmente cuenta con 3 tipos de atenciones:

  - Incidencia: En el contexto de esta investigación, un incidente se entiende como una alteración no planificada que interrumpe un servicio de TI o reduce su nivel de calidad, y cuyo efecto puede limitarse a un usuario o extenderse a varias áreas de la organización (AXELOS, 2019). La mesa de ayuda realiza la atención inicial en el nivel N1; cuando la solución requiere intervenir una aplicación y supera su

capacidad técnica, el ticket se escala al nivel N2, gestionado por la fábrica de software.

        - Incidente de Seguridad: Es un subtipo de Incidencia, es una afectación negativa a la Seguridad de la Información. Es gestionado por N1.
    - Solicitud de Servicio: Una solicitud de servicio es un requerimiento habitual iniciado por el usuario para obtener información, asistencia, acceso, equipamiento o software que forma parte de la oferta previamente definida por el área de TI (AXELOS, 2019). Debido a que se trata de atenciones estandarizadas y previsibles, su gestión corresponde al nivel N1 de la mesa de ayuda, de acuerdo con el catálogo de servicios y los tiempos de atención establecidos.
    - Solicitud de Cambio: Es la adición, modificación o eliminación de cualquier cosa que pueda afectar directa o indirectamente los servicios de TI. Los requisitos de software (nuevas funcionalidades o mejoras en las aplicaciones) ingresan a través de una solicitud de cambio a la Mesa de Ayuda y, tras su revisión inicial, la deriva a la fábrica de software quien a su vez la convierte en Requisito en Azure DevOps para que se gestione bajo Scrum.
- Gestión de Requisitos (Fábrica de Software)

Este proceso cubre el diseño, desarrollo y mantenimiento de plataformas digitales utilizadas por clientes, beneficiarios y comercios afiliados. La Fábrica de Software atiende todo cambio, modificación o creación de una funcionalidad en un sistema existente en la empresa que le son derivados desde mesa de ayuda como:

- Incidencia N2: Cuando mesa de ayuda deriva un ticket de incidencia en una aplicación de software que no puede atender.
- Requisitos de Software: Cuando mesa de ayuda deriva a fábrica de software una solicitud de cambio que implica nuevas funcionalidades o mejoras en las aplicaciones.

Cada área de la empresa define sus requisitos de forma interna, define el requisito en un formato en Excel y lo registra en Jira como una solicitud de cambio. Fábrica de Software recibe el requisito, lo transforma en historia de usuario, analiza y estima su implementación en un sprint de 2 semanas.

En una reunión de requisitos al inicio de cada sprint se define la criticidad del requisito versus la carga actual de la fábrica para identificar los requisitos a ser atendidos en el sprint de 2 semanas.

Durante la ejecución del sprint se pueden dar reuniones con los usuarios para aterrizar a mayor nivel de detalle el requisito, una vez concluido se planifica el despliegue en producción.

Procesos de Apoyo:

- Servicios de Credenciales

Este proceso gestiona el acceso de clientes a la plataforma de gestión de pedidos de tarjetas por medio de una implementación multifactor (MFA) la cual comprueba las credenciales del cliente (web / móvil) y también de usuarios internos por medio de servicios Okta (web).

- Monitoreo de Amenazas

Este proceso supervisa constantemente las amenazas y vulnerabilidades de la infraestructura TI en coordinación con el área de ciberseguridad de la empresa en su sede matriz.

- Infraestructura de TI

Este proceso administra el hardware, software, redes y otros activos tecnológicos necesarios para operar los sistemas de la empresa.

3.6.4 Procesos Específicos vs Objetivos

Tabla 6
Procesos Específicos vs Objetivos Estratégicos

Objetivos / Procesos Específicos	Acelerar la Oferta de Beneficios para Colaboradores	Enfocarse en Pequeñas y Medianas Empresas (Pymes)	Impulsar la Transformación Digital	Promover Sostenibilidad Ambiental	Mejorar la Calidad de Vida de los Colaboradores
Gobierno de TI		X	X	X	
Seguridad de la Información		X	X		X
Desarrollos All-in-One			X		
Gestión de Riesgos de TI	X		X		X
Plataformas de Pago			X	X	X
Gestión de Incidencias (Mesa de Ayuda)	X	X	X	X	X
Gestión de Requisitos (Fábrica de Software)	X	X	X	X	X
Servicios de Credenciales		X	X		
Monitoreo de Amenazas		X	X		X
Infraestructura de TI			X		

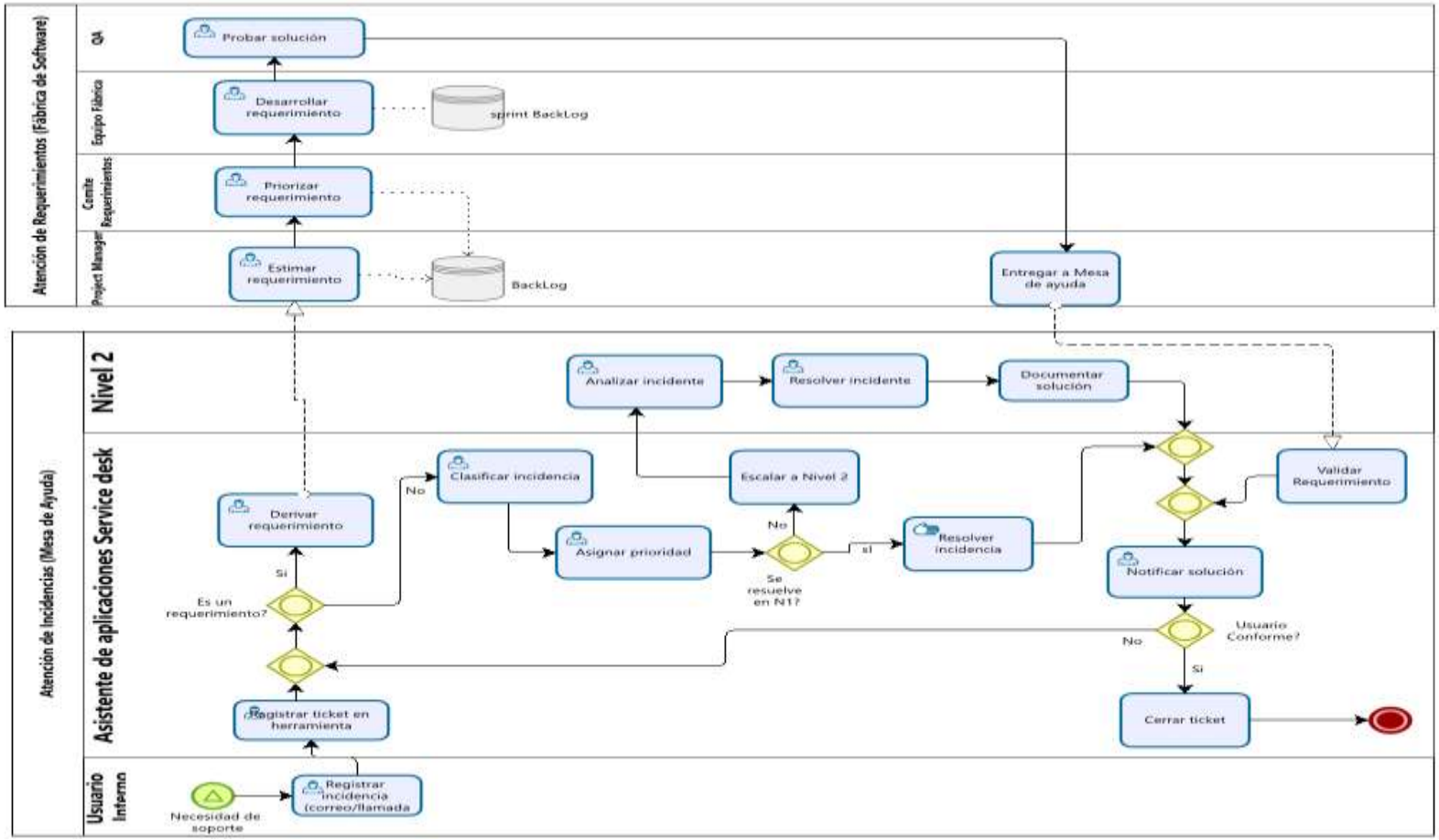
En base al catálogo se identifica 2 procesos relevantes que impacta directamente en los objetivos estratégicos.

- Gestión de Incidencias (Mesa de Ayuda)
- Gestión de Requisitos (Fábrica de Software)

Es evidente el impacto de estos 2 procesos dada la importancia sobre la cual se soporta el negocio en la empresa, su relevancia dentro de la empresa, la estrategia digital y su eficiencia operativa, siendo necesario realizar un análisis de estos procesos con el fin de identificar puntos de dolor y con ello propuestas de solución para mitigarlos.

3.7 AS-IS del Proceso de Gestión de Incidencias (Mesa de Ayuda) y Gestión de Requisitos (Fábrica de Software)

Figura 16
BPMN (As-Is) Fábrica de Software



El proceso de Gestión de Incidencias (Mesa de Ayuda) y Gestión de Requisitos (Fábrica de Software) como parte de los procesos de la Gerencia de Sistemas son de los más importantes y de mayor impacto dentro de la empresa dado lo integrado en el negocio de su plataforma tecnológica al conectar a la empresa con sus clientes, beneficiarios y comercios.

3.7.1 Descripción del Proceso de Atención de Incidencias (Mesa de Ayuda)

Nombre del Proceso	
Atención de Incidencias (Mesa de Ayuda)	
Objetivo:	Brindar soporte técnico eficiente a los usuarios internos de la empresa, asegurando la rápida solución de incidencias y mejorando la disponibilidad de los sistemas.
Alcance:	Inicia cuando un usuario registra su incidencia (solicitud de atención) a través de la plataforma Jira y finaliza cuando la incidencia es solucionada y comunicada al usuario. Promedio de atención: 1.5 horas
Recursos Involucrados:	Personal de Mesa de Ayuda (3 agentes)
	Plataforma Jira
	Estaciones de usuario
	Servidores de aplicaciones
	Infraestructura de Red

Tabla 7
Descripción del Proceso de Atención de Incidencias (Mesa de Ayuda)

Entrada	Actividad	Salida	Descripción	Responsable
Solicitud de atención	Recepcionar la solicitud de atención	Solicitud de atención - JIRA	El analista de aplicaciones y mesa de ayuda recepciona la solicitud de atención en la plataforma JIRA	Analista de aplicaciones y Mesa de ayuda
Solicitud de atención - JIRA	Clasificar la solicitud de atención (incidencia o requisito de software)	Solicitud de atención (incidencia) - JIRA	El analista de aplicaciones y mesa de ayuda clasifica si la solicitud de atención si es una incidencia o un requisito de SW	Analista de aplicaciones y Mesa de ayuda

Entrada	Actividad	Salida	Descripción	Responsable
Solicitud de atención (incidencia) - JIRA	Analizar la solicitud de atención (incidencia)	Solicitud de atención (incidencia) - JIRA	El analista de aplicaciones y mesa de ayuda analiza la solicitud de atención que contiene el detalle de la incidencia.	Analista de aplicaciones y Mesa de ayuda
Solicitud de atención (incidencia) - JIRA	Solicitar información complementaria para la solicitud de atención (incidencia)	Solicitud de atención (incidencia) - JIRA	El analista de aplicaciones y mesa de ayuda solicita más información de la incidencia.	Analista de aplicaciones y Mesa de ayuda
Solicitud de atención (incidencia) - JIRA	Recibir información complementaria para la solicitud de atención (incidencia)	Solicitud de atención (incidencia) - JIRA	Con la información recibida, el analista de aplicaciones y mesa de ayuda complementa su análisis de la incidencia.	Analista de aplicaciones y Mesa de ayuda
Solicitud de atención (incidencia) - JIRA	Clasificar el alcance del trabajo	Solicitud de atención (incidencia) - JIRA	El analista de aplicaciones y mesa de ayuda clasifica el alcance del trabajo a realizar para atender la incidencia.	Analista de aplicaciones y Mesa de ayuda
Solicitud de atención (incidencia) - JIRA	Ejecutar correcciones a la incidencia	Solicitud de atención (incidencia) - JIRA	El analista de aplicaciones y mesa de ayuda procede con la atención de la incidencia	Analista de aplicaciones y Mesa de ayuda
Solicitud de atención (incidencia) - JIRA	Elaborar informe de resultados	Informe de resultados	El analista de aplicaciones y mesa de ayuda genera el informe de resultados como un comentario en la plataforma JIRA sobre la solicitud de atención registrada.	Analista de aplicaciones y Mesa de ayuda

3.7.2 Descripción del Proceso de Atención de Requisitos (Fábrica de Software)

Nombre del Proceso	
Atención de Requisitos (Fábrica de Software)	
Objetivo:	Garantizar el desarrollo, mantenimiento y entrega de soluciones de software alineadas al requerimiento solicitado, cumpliendo con estándares de calidad, tiempos de entrega y seguridad de la información, asegurando la continuidad operativa y la transformación digital en la empresa.
Alcance:	El proceso inicia cuando un usuario registra una solicitud de atención del tipo requerimiento de software en la plataforma Jira. Luego, la solicitud es evaluada, priorizada y asignada a la Fábrica de Software. Finaliza cuando el requerimiento es desarrollado, probado, validado por el usuario e implementado en producción. Promedio de atención: 14 horas por implementación.
Recursos Involucrados:	Personal de la Fábrica de Software (6 recursos)
	Plataforma Jira
	Servidores de desarrollo, UAT, STA
	Servidores de bases de datos
	Infraestructura de red

Tabla 8
Descripción del Proceso de Atención de Requisitos (Fábrica de Software)

Entrada	Actividad	Salida	Descripción	Responsable
Solicitud de atención	Recepcionar la solicitud de atención	Solicitud de atención - JIRA	El analista de aplicaciones y mesa de ayuda recepciona la solicitud de atención en la plataforma JIRA	Analista de aplicaciones mesa de ayuda
Solicitud de atención - JIRA	Clasificar la solicitud de atención (incidencia o requisito de software)	Solicitud de atención (RQ-Software) - JIRA	El analista de aplicaciones y mesa de ayuda clasifica si la solicitud de atención si es una incidencia o un requisito de SW para lo cual lo deriva al Project Manager de la fábrica de software.	Analista de aplicaciones y Mesa de ayuda
Solicitud de atención (RQ-Software) - JIRA	Consolidar la solicitud de atención (RQ-Software)	Backlog - Solicitud de atención (RQ-Software) – Azure DevOps	El Project Manager de la fábrica de software recepciona la solicitud de atención (RQ-Software) y lo registra en Azure DevOps.	Project Manager

Entrada	Actividad	Salida	Descripción	Responsable
Backlog - Solicitud de atención (RQ-Software) – Azure DevOps	Estimar la solicitud de atención (RQ-Software)	Backlog - Solicitud de atención (RQ-Software) – Azure DevOps	El Project Manager con el equipo de fábrica se encargan de realización la estimación del esfuerzo del requisito de software.	Project Manager Equipo de Fábrica
Backlog - Solicitud de atención (RQ-Software) – Azure DevOps	Priorización de solicitudes de atención (RQs-Software)	Sprint Backlog - Solicitud de atención (RQ-Software) – Azure DevOps	Comité de requisitos prioriza los requisitos de software que se atenderán en el siguiente sprint.	Comité de requisitos
Sprint Backlog - Solicitud de atención (RQ-Software) – Azure DevOps	Desarrollar las solicitudes de atención (RQs-Software) priorizadas.	Sprint Backlog - Desarrollos de software – Azure DevOps	El equipo de fábrica desarrolla los requisitos de software.	Equipo de Fábrica
Sprint Backlog - Desarrollos de software – Azure DevOps	Ejecutar pruebas de calidad	Sprint Backlog - Desarrollos de software (Defectos) – Azure DevOps	El analista QA realiza pruebas de software y en caso se detecte defectos se retorna al equipo de fabrica con el detalle de la evidencia	Analista QA
Sprint Backlog - Desarrollos de software (Validados) – Azure DevOps	Ejecutar la validación del requisito de software con el usuario final	Sprint Backlog - Desarrollos de software (Aprobado) – Azure DevOps	El analista QA valida con el usuario final el resultado del requisito de software	Analista QA
Sprint Backlog - Desarrollos de software (Aprobado) – Azure DevOps	Entregar a mesa de ayuda	Acta de conformidad	El Project Manager procede administrativo con el cierre de la solicitud de atención (RQ-Software)	Project Manager

3.8 Puntos de Dolor del Proceso Atención de Incidencias (Mesa de Ayuda) y Atención de Requisitos (Fábrica de Software)

Dado que la empresa ofrece soluciones de beneficios y engagement para empleados de empresas, el uso de la tecnología es un pilar fundamental ya que permite la gestión de los procesos de pedidos de tarjetas, asegura la integridad de las transacciones que realizan los beneficiarios y protege la integridad de los datos sensibles de sus clientes.

Sabiendo que las necesidades de sus clientes varían frecuentemente y para Kred C&S es importante promover campañas con el fin de asegurar y captar nuevos clientes, se hace necesaria las actualizaciones de las aplicaciones core del negocio, implementando nuevas funcionalidades, por ejemplo, para la gestión del pedido de tarjetas, así como también en la usabilidad de las funcionalidades al hacer uso de la app de la empresa.

Para identificar los puntos de dolor se aplicó un enfoque combinado con los siguientes métodos:

- Modelo del proceso As-Is (sección 3.7) para visualizar las actividades reales y cuellos de botellas.
- Análisis de brechas contra indicadores de desempeño (KPI) y percepción definidos en la sección 3.8.1.
- Assement basado en entrevista a roles tanto para la mesa de ayuda como para la fábrica de software con el objetivo de confirmar los hallazgos.

El siguiente cuadro consolida dicho enfoque con los 2 primeros puntos y los vincula de forma trazable con el método aplicado. Para cada punto de dolor se incluye la evidencia del hallazgo.

Tabla 9
Puntos de dolor

Punto de dolor	Servicio	Procedimiento de Identificación	Referencia de buenas prácticas (I (TIL/Scrum)	Indicador / Métrica relacionada (Sección 3.8.1)
No se analiza sistemáticamente el historial de tickets	Mesa de ayuda	BPMN (sección 3.7.1)	Monitoring & Event Mgmt Problem Mgmt	ND

Punto de dolor	Servicio	Procedimiento de Identificación	Referencia de buenas prácticas (I (TIL/Scrum)	Indicador / Métrica relacionada (Sección 3.8.1)
Altas derivaciones a N2 y tiempos de espera	Mesa de ayuda	BPMN (sección 3.7.1); KPIs	Service Level Mgmt Incident Mgmt	% Cumplimiento de SLA (Mensual y por prioridad)
Transferencia de conocimiento insuficiente	Mesa de ayuda	KPI	Knowledge Mgmt Problem Mgmt	CSAT (Mensual) – Servicio de Mesa de Ayuda
Variabilidad en requisitos y retrabajos	Fábrica de Software	BPMN (sección 3.7.2); KPIs	Product Backlog Refinement DoR/DoD	Velocidad de Entrega de Requisitos (Implementaciones vs Objetivo)
Priorización y gobernanza de cambios débiles y sin criterio	Fábrica de Software	KPI	Change Enablement Sprint/Release Planning	NPS Interno por Sprint – Servicio de Fábrica de Software
Reaperturas frecuentes de tickets	Mesa de ayuda	KPI	Problem Mgmt Knowledge Mgmt	% de Reapertura (Calidad de cierre del ticket de incidencia)
Base de conocimiento incompleta/desactualizada y transferencia de conocimiento insuficiente	Mesa de ayuda	KPI	Knowledge Mgmt Problem Mgmt	CSAT (Mensual) – Servicio de Mesa de Ayuda
Cambios urgentes mal evaluados/gobernados	Fábrica de Software	KPI	Change Enablement Sprint/Release Planning	NPS Interno por Sprint – Servicio de Fábrica de Software
Defectos post-release elevados	Fábrica de Software	KPI	DoD y Calidad de Software Service Validation & Testing	% de Fallos Post-Implementación
Cuello de botella en QA	Fábrica de Software	KPI	Políticas de flujo de desarrollo de software Trabajo en proceso Sprint Review	NPS Interno por Sprint – Servicio de Fábrica de Software

Con los puntos de dolor identificados, se realiza un assessment con los principales roles de la mesa de ayuda y la fábrica de software, respondiendo cada rol solo si está en capacidad de confirmar lo identificado, en caso contrario se responde NA.

Tabla 10
Puntos de dolor vs Roles

Roles / Punto de dolor	Jefe de TI	Coordinador de Mesa	Agente N1	Agente N2	Product Owner	Project Manager	Desarrollador
Altas derivaciones a N2 y tiempos de espera	Sí	Sí	Sí	Sí	NA	Sí	NA
Reaperturas frecuentes de tickets	Sí	Sí	Sí	Sí	NA	Sí	NA
Transferencia de conocimiento insuficiente	Sí	Sí	Sí	Sí	NA	Sí	NA
Base de conocimiento incompleta/desactualizada y transferencia de conocimiento insuficiente	Sí	Sí	Sí	Sí	NA	NA	NA
No se analiza sistemáticamente el historial de tickets	Sí	Sí	Sí	Sí	NA	Sí	Sí
Variabilidad en requisitos y retrabajos	Sí	NA	NA	NA	Sí	Sí	Sí
Priorización y gobernanza de cambios débiles y sin criterio	Sí	NA	NA	NA	Sí	Sí	Sí
Cambios urgentes mal evaluados/gobernados	Sí	NA	NA	NA	Sí	Sí	Sí
Defectos post-release elevados	Sí	NA	NA	NA	Sí	Sí	Sí
Cuello de botella en QA	Sí	NA	NA	NA	Sí	Sí	Sí

En síntesis, la mesa de ayuda presenta altas derivaciones a N2 y tiempos de espera, reaperturas frecuentes, transferencia de conocimiento deficientes / base de conocimiento desactualizada y ausencia de análisis sistemático del historial de tickets, afectando el cumplimiento de SLA y la percepción del servicio (CSAT / NPS). La Fábrica de Software evidencia variabilidad en requisitos y retrabajos, gobernanza con priorización débil de cambios, defectos post-implementación y cuellos de botella en QA con impacto directo en oportunidad y calidad de entrega. Las derivaciones de mesa de ayuda a fábrica de software

(de incidencias N1 a N2) amplifican los impactos cuando los flujos de atención no están alineados.

Esta falta de estructura metodológica dificulta la trazabilidad de los requisitos, genera reprocesos y afecta la capacidad de respuesta del equipo de TI ante las necesidades del negocio.

3.8.1 Métricas e Indicadores Detectados al Proceso Gestión de Incidencias (Mesa de Ayuda) y Gestión de Requisitos (Fábrica de Software)

Como parte del análisis de brechas, la eficiencia en la gestión de servicios de TI es un factor clave para garantizar la competitividad y el crecimiento sostenible. En el caso de la mesa de ayuda y la fábrica de software presentan desafíos que impactan directamente en los tiempos de respuesta, la calidad de las soluciones entregadas y la satisfacción de los usuarios internos.

A continuación, se muestra el análisis de desempeño de la mesa de ayuda:

- **% Cumplimiento de SLA (Mensual y por prioridad)**:

  Medición: Calculo por periodo (mes)

  Periodos considerados en el análisis:

  - Ago-24, Set-24, Oct-24, Nov-24, Dic-24, Ene-25

  Muestra:

  - Tickets atendidos en el mes.

  Formula:

  - % Cumplimiento SLA (Mes) = (Tickets Cumplen SLA / Tickets Resueltos) * 100
  - % Cumplimiento SLA (Prioridad) = (Tickets Cumplen SLA por prioridad / Tickets Resueltos por prioridad) * 100

Tabla 11
Prioridad – SLA (Mesa de Ayuda)

Prioridad	SLA
P1	1 hora
P2	2 horas
P3	8 horas
P4	16 horas
P5	24 horas

Tabla 12

Cumplimiento de Tickets por SLA (Mesa de ayuda)

Mes	Total Tickets	Tickets P5	P5 Cumplen SLA	Tickets P4	P4 Cumplen SLA	Tickets P3	P3 Cumplen SLA	Tickets P2	P2 Cumplen SLA	Tickets P1	P1 Cumplen SLA	Tickets Incumpliendo SLA
Aug-24	368	166	166	118	116	74	19	8	2	2	0	65
Sep-24	335	151	151	108	105	66	15	7	2	3	1	61
Oct-24	423	191	191	136	136	86	28	9	3	1	0	65
Nov-24	425	192	192	136	133	87	19	9	4	1	0	77
Dec-24	438	198	198	141	138	89	16	9	3	1	0	83
Jan-25	389	176	176	125	124	79	23	8	2	1	0	64

Tabla 13

% Cumplimiento de SLA Mesa de Ayuda (Mensual y Prioridad)

Mes	Tickets Resueltos	Tickets Cumplen SLA	% Tickets Cumplen SLA	% Cumplimiento P5	% Cumplimiento P4	% Cumplimiento P3	% Cumplimiento P2	% Cumplimiento P1	% Incumplimiento SLA
Aug-24	368	303	82.3%	100.0%	98.3%	25.7%	25.0%	0.0%	17.7%
Sep-24	335	274	81.8%	100.0%	97.2%	22.7%	28.6%	33.3%	18.2%
Oct-24	423	358	84.6%	100.0%	100.0%	32.6%	33.3%	0.0%	15.4%
Nov-24	425	348	81.9%	100.0%	97.8%	21.8%	44.4%	0.0%	18.1%
Dec-24	438	355	81.1%	100.0%	97.9%	18.0%	33.3%	0.0%	18.9%
Jan-25	389	325	83.5%	100.0%	99.2%	29.1%	25.0%	0.0%	16.5%

Lectura:

- Con base a la muestra de conteo de todos los tickets resueltos en 6 meses, el promedio de los que cumplieron su meta de SLA se ubica entre 81.1% y 84.6% cuando la exigencia de la Gerencia de Sistemas es mayor al 95%.
- A nivel cumplimiento de SLA por prioridad, P5–P4 presentan cumplimiento casi total, mientras que P3 y P2 concentran la mayor parte de los incumplimientos; P1 exhibe niveles mínimos de cumplimiento (0 a 33%).

Conclusión:

- Estos resultados orientan la visibilidad operativa hacia mejoras en tiempos de resolución de P3/P2/P1 lo que implica cambios en sus procesos y procedimientos de atención.
- Se detecta riesgo operativo con los tickets P1 y P2 no atendidos a tiempo que involucra el apoyo de fábrica de software, esto evidencia falta de

comunicación, recursos en mesa de ayuda y compartir base de conocimiento entre ambos servicios para atender estos tickets críticos y bloqueantes.

- **% de Reapertura (Calidad de cierre del ticket de incidencia)**:

  Medición: Calculo por periodo (mes)

  Periodos considerados en el análisis:

  - Ago-24, Set-24, Oct-24, Nov-24, Dic-24, Ene-25

  Muestra:

  - Tickets atendidos en el mes.

  Formula

  - Porcentaje de Reapertura (Calidad de cierre) = (Tickets Reabiertos / Tickets Resueltos) * 100

  Umbral: 3% dentro de los 14 días posterior al cierre del ticket de incidencia.

Tabla 14
Total de Tickets Reabiertos por Prioridad

Mes	Total Tickets	Tickets Reabiertos P5	Tickets Reabiertos P4	Tickets Reabiertos P3	Tickets Reabiertos P2	Tickets Reabiertos P1
Aug-24	368	0	0	15	2	0
Sep-24	335	0	0	10	2	0
Oct-24	423	0	0	13	1	0
Nov-24	425	0	0	13	1	0
Dec-24	438	0	0	12	3	0
Jan-25	389	0	0	18	2	0

Tabla 15
% de Reapertura (Calidad del Cierre)

Mes	Total Tickets	% Reincidencia	% Tickets Reabiertos P5	% Tickets Reabiertos P4	% Tickets Reabiertos P3	% Tickets Reabiertos P2	% Tickets Reabiertos P1
Aug-24	368	4.6%	0%	0%	4.1%	0.5%	0.0%
Sep-24	335	3.6%	0%	0%	3.0%	0.6%	0.0%
Oct-24	423	3.3%	0%	0%	3.1%	0.2%	0.0%
Nov-24	425	3.3%	0%	0%	3.1%	0.2%	0.0%
Dec-24	438	3.4%	0%	0%	2.7%	0.7%	0.0%
Jan-25	389	5.1%	0%	0%	4.6%	0.5%	0.0%

Lectura:

- Entre Ago-24 y Ene-25 el % mensual de reaperturas oscilo entre 3.3% y 5.1% lo cual es mayor al umbral referencial de 3% con picos en agosto y enero.
- Por prioridad, la reapertura de tickets se concentra en P3 y P2 mientras que P5, P4, P1 se mantienen en 0%.

Conclusión:

- Los tickets muestran fallos en el diagnostico al ser incompletos, validación insuficiente y comunicación al usuario.
- La reapertura de tickets P2, P3 implica reprocesos que consumen capacidad y deterioran la percepción del servicio.
- Estos resultados sugieren reforzar la validación de cierre, involucrar apoyo exclusivo de nivel 2 - fábrica de software y cambios en sus procesos y procedimientos de atención.

- **NPS Interno por Periodo (mes) – Servicio de Mesa de Ayuda**:

  Medición: Periodo mensual.

  Periodos considerados en el análisis:
  - Ago-24, Set-24, Oct-24, Nov-24, Dic-24, Ene-25

  Muestra:
  - 53 trabajadores por periodo.

  Pregunta:
  - ¿Qué tan probable es que recomiendes el servicio de mesa de ayuda a un colega o área usuaria?

  Respuesta (0 – 10) alineada a la percepción del usuario con respecto a:
  - La calidad de atención de la mesa de ayuda.

  Formula:
  - NPS = %Promotores (9 y 10) - % Detractores (0 al 6)

Tabla 16
Resultado Encuesta Recomendación - Mesa de Ayuda

Sprint	Promotores (9-10)	Pasivos (7-8)	Detractores (0-6)	Muestra (trabajadores)
Aug-24	12	4	37	53
Sep-24	14	3	36	53
Oct-24	11	3	39	53
Nov-24	8	2	43	53
Dec-24	7	4	42	53
Jan-25	12	3	38	53

Tabla 17
Resultado NPS - Mesa de Ayuda

Sprint	NPS
Aug-24	-47.2%
Sep-24	-41.5%
Oct-24	-52.8%
Nov-24	-66.0%
Dec-24	-66.0%
Jan-25	-49.1%

Lectura:

- El NPS por periodo (mes) oscila entre -66% y -41.5% con promedio de -53.8%.
- En paralelo, el % de Incumplimiento de SLA varía entre 15.4% y 18.9% y el % de reaperturas entre 3.3% y 5.1%.
- Adicionalmente, en esta muestra de 6 meses se observa una asociación negativa entre NPS y el % de Incumplimiento de SLA (-0.85%) mientras que su relación con el % de reaperturas se muestra como positiva (0.48)

Conclusión:

- El resultado NPS de los 6 meses analizados nos muestra que los usuarios que utilizan la mesa de ayuda no recomendarían el servicio.
- El resultado de la asociación negativa sugiere que en el periodo analizado el incumplimiento de la SLA por parte de la mesa de ayuda tiene un mayor impacto en la percepción general de los empleados que la reapertura de tickets que presenta un resultado positivo.

- **CSAT (Mensual) – Servicio de Mesa de Ayuda**

  Medición: Post-cierre enviada automáticamente a 100% de tickets de incidencias cerrados en el mes; ventana de respuesta 72h; se excluyen duplicados/pruebas.

  Pregunta: "Califique su satisfacción general con la atención recibida a su ticket de incidencia por el servicio de mesa de ayuda"

  Escala Likert: 1 – 5

  CSAT% = ((Muy Satisfecho + Satisfecho) / Total) * 100

  Muestra: Por periodo (mes)

  - Ago-24, Set-24, Oct-24, Nov-24, Dic-24, Ene-25

Tabla 18
Resultados Encuesta Satisfacción - Mesa de Ayuda

Periodo	Total Tickets	Encuestas enviadas	Encuestas respondidas	Muy satisfecho	Satisfecho	Neutral	Insatisfecho	Muy insatisfecho
Aug-24	368	368	149	14	20	11	82	22
Sep-24	335	335	154	15	20	12	84	23
Oct-24	423	423	175	17	23	13	96	26
Nov-24	425	425	188	18	25	14	103	28
Dec-24	438	438	156	15	21	12	85	24
Jan-25	389	389	161	15	21	12	88	24

Tabla 19
CSAT (Mensual) - Mesa de Ayuda

Periodo	Total Tickets	Encuestas respondidas	Muy satisfecho	Satisfecho	CSAT	Tasa de respuestas
Aug-24	368	149	14	20	22.8%	40.5%
Sep-24	335	154	15	20	22.7%	46.0%
Oct-24	423	175	17	23	22.9%	41.4%
Nov-24	425	188	18	25	22.9%	44.2%
Dec-24	438	156	15	21	23.1%	35.6%
Jan-25	389	161	15	21	22.4%	41.4%

Lectura:

- El CSAT mensual de la mesa de ayuda se sitúa entre 22.4% y 23.1% para la muestra analizada en base a un rango de 149 a 188 opiniones por mes.

- Se evidencia una asociación negativa con respecto al % de reaperturas de tickets (-0.77) y no se evidencia una relación negativa con respecto al % de Incumplimiento de SLA (0.31).

Conclusión:

- El resultado CSAT de los 6 meses analizados nos muestra que los usuarios que utilizan la mesa de ayuda se encuentran poco satisfechos con el servicio recibido en cada mes que se les ha consultado.
- El resultado de la asociación negativa con respecto a la reapertura de tickets nos sugiere que en estos periodos la calidad y resolución efectiva de tickets tienen mayor peso que la percepción al incumplimiento de SLA en las incidencias.
- Adicionalmente, es necesario precisar que predomina las respuestas "insatisfechas" en todos los meses lo cual es coherente con el CSAT anual de la mesa de ayuda indicado en la sección 1.2.2. Esto demuestra que los procesos actuales de la mesa de ayuda no son los óptimos, se evidencia falta de preparación e incluso capacidad operativa.

A continuación, se muestra el análisis de desempeño de la atención de requisitos por parte de la fábrica de software:

- **Velocidad de Entrega de Requisitos (Implementaciones vs Objetivo)**: Durante los últimos 6 Sprint de desarrollo, el número de implementaciones entregadas ha estado por debajo del objetivo establecido por la gerencia de Sistemas (20 implementaciones por sprint), con valores fluctuantes entre 5 y 19 implementaciones atendidas, evidenciando problemas en la planificación y ejecución de los desarrollos. Este indicador se obtiene de la información del Sprint Backlog que se almacena en Azure DevOps obteniendo los valores de los tiempos que ha tomado la resolución de la atención de requisitos desde su inicio de atención hasta la puesta en producción.

Tabla 20
Requisitos atendidos por Sprint

Sprint	Objetivo de Atenciones por Sprint	Requisitos Atendidos	% de Cumplimiento
FY25-Q1-SP5	20	9	45%
FY25-Q1-SP6	20	9	45%
FY25-Q1-SP7	20	5	25%
FY25-Q2-SP1	20	16	80%
FY25-Q2-SP2	20	18	90%
FY25-Q2-SP3	20	19	95%
FY25-Q2-SP4	20	15	75%
FY25-Q2-SP5	20	12	60%

Figura 17
Velocidad de Entrega de RQs vs Objetivo por Sprint



Lectura:

- La entrega por sprint se mantiene por debajo del objetivo (20), con variación notable entre sprints lo que denota baja predictibilidad (compromiso del equipo, procesos engorrosos dentro de la fábrica de software).
- La dispersión sugiere historias mal definidas, grandes o disparejas lo cual dificulta planificar y terminarlas a tiempo.
- Cuellos de botella en las actividades de QA/Deploy, más que un problema puntual.
- Hay subutilización de la meta en sprints con bloqueos externos (dependencias entre historias de usuario, retraso en aprobaciones de PO y usuario solicitante)
- Se refleja el trabajo proveniente de un sprint anterior (carryover).

Conclusión:

- Reforzar el DoR y dividir las historias (slicing vertical) para estabilizar la cantidad terminada de historias por sprint (throughput) y mejorar la capacidad de compromiso. Esto implica cambios en procesos de priorización y la definición del sprint backlog.
- Implementar límites de WIP y una política de para la línea cuando aparece un bloqueo / defecto crítico, se priorizaría resolverlo antes de continuar y evitar el desperdicio.
- Medir y gestionar capacidad por recurso de la fábrica de software para identificar cuellos de botellas y usar forecasting basado en throughput histórico y no con estimaciones subjetivas

- **% de Fallos Post-Implementación**: Se ha identificado que en cada sprint entre 1 y 4 implementaciones presentan fallos en las aplicaciones derivadas de sus pases a producción (post-implementación) por lo que implica retrabajos y retrasos.
  Esta información se obtiene de los Excel almacenados en Azure DevOps y que se relaciona contrastando los casos de pruebas y los defectos detectados por el equipo de calidad durante las revisiones de software de los desarrollos entregados por el equipo de fábrica.

Tabla 21
% de Fallos Post-Implementación

Sprint	Requisitos Atendidos	Requisitos con fallos post-implantación	% de Retornos
FY25-Q1-SP5	9	1	11%
FY25-Q1-SP6	9	1	11%
FY25-Q1-SP7	5	1	20%
FY25-Q2-SP1	16	3	19%
FY25-Q2-SP2	18	3	17%
FY25-Q2-SP3	19	4	21%
FY25-Q2-SP4	15	3	20%
FY25-Q2-SP5	12	2	17%

Figura 18
Total de Fallos Post-Implantación por Sprint



Lectura:

- El porcentaje de fallos post-implementación se ubica en un rango elevado (entre 11 y 21%), generando retrabajo, reducción de credibilidad, confianza y satisfacción del usuario.
- Los picos de fallos coinciden con sprints donde “casi se cumple” la velocidad de entrega.
- El % de fallos evidencia brecha en DoD (pruebas, criterios de aceptación, evidencias).

Conclusión:

- Fortalecer Definition of Done y aplicar calidad desde el inicio para prevenir defectos en lugar de “arreglar después”.
- Se evidencia que al acercarse al objetivo de 20 implementaciones por sprint los defectos se incrementan lo cual indica que la aceleración sin controles aumenta los defectos.
- Revisar la capacidad de QA y/o aumentar pruebas automatizadas realizar análisis de causa raíz para reducir reincidencia.

- **NPS Interno por Sprint- Servicio de Fábrica de Software**:

  Medición: Encuesta post-sprint (2 semanas cada uno)

  Sprints considerados en el análisis:

  - FY25-Q1-SP5, FY25-Q1-SP6, FY25-Q1-SP7, FY25-Q2-SP1, FY25-Q2-SP2, FY25-Q2-SP3, FY25-Q2-SP4, FY25-Q2-SP5

  Muestra:

  - 53 trabajadores por sprint.

  Pregunta:

  - ¿Qué tan probable es que recomiendes el servicio de fábrica de software a un colega o área usuaria?

  Respuesta (0 – 10) alineada a la percepción del usuario con respecto a:

  - Velocidad de entrega de sus requisitos.
  - Afectación por los fallos que le hayan ocurrido post-implementación de sus requisitos.

  Formula:

  - NPS = %Promotores (9 y 10) - % Detractores (0 al 6)

Tabla 22
Resultado Encuesta Recomendación - Fábrica de Software

Sprint	Promotores (9-10)	Pasivos (7-8)	Detractores (0-6)	Muestra (trabajadores)
FY25-Q1-SP5	17	4	32	53
FY25-Q1-SP6	16	3	34	53
FY25-Q1-SP7	17	3	33	53
FY25-Q2-SP1	16	6	31	53
FY25-Q2-SP2	16	4	33	53
FY25-Q2-SP3	15	3	35	53
FY25-Q2-SP4	17	2	34	53
FY25-Q2-SP5	16	5	32	53

Tabla 23
Resultado NPS - Fábrica de Software

Sprint	NPS
FY25-Q1-SP5	-28.3%
FY25-Q1-SP6	-34.0%
FY25-Q1-SP7	-30.2%
FY25-Q2-SP1	-28.3%
FY25-Q2-SP2	-32.1%
FY25-Q2-SP3	-37.7%
FY25-Q2-SP4	-32.1%
FY25-Q2-SP5	-30.2%

Lectura:

- El NPS por sprint varía entre -28.3% y -37.7% con promedio de -31-6%.
- En el mismo periodo, la velocidad de entrega de sus requisitos alcanza entre 25.0% y 95.0% de cumplimiento, y él % de fallos post-implementación entre 11.11% y 21.05%.
- El peor NPS (−37.7% en el sprint FY25-Q2-SP3) coincide con el mayor % de fallos post-implementación (21.05%) pese a ser el sprint de mayor velocidad (95% del objetivo).

- En este sprint, el coeficiente NPS vs % fallos post-implementación muestra una asociación negativa (-0.26) y el coeficiente NPS vs % cumplimiento de entrega no evidencia mejora por aumento de la velocidad (-0.43).

Conclusión:

- El resultado NPS de los sprint analizados, nos muestra que los usuarios que utilizan la fábrica de software no recomendarían el servicio.
- El resultado de la asociación negativa en el sprint FY25-Q2-SP3 sugiere que la calidad post-implementación tiene un peso perceptivo mayor que el volumen de entregas para los usuarios.

- **CSAT (Mensual) – Servicio de Fábrica de Software**

  Medición: Encuesta post-sprint (2 semanas cada uno)

  Pregunta: "Califique su satisfacción general con la atención recibida de su requisito por el servicio de fábrica de software"

  Escala Likert: 1 – 5

  CSAT% = ((Muy Satisfecho + Satisfecho) / Total) * 100

  Muestra: Por sprint

  - FY25-Q1-SP5, FY25-Q1-SP6, FY25-Q1-SP7, FY25-Q2-SP1, FY25-Q2-SP2, FY25-Q2-SP3, FY25-Q2-SP4, FY25-Q2-SP5

Tabla 24

Resultado Encuesta de Satisfacción - Fábrica de Software

Sprint	Mes	Total de Requisitos Atendidos	Requisitos con fallas	Encuestas enviadas	Encuestas respondidas	Muy satisfecho	Satisfecho	Neutral	Insatisfecho	Muy insatisfecho
FY25-Q1-SP5	Oct-24	9	1	45	35	4	7	3	16	5
FY25-Q1-SP6	Nov-24	9	1	45	32	4	7	2	14	5
FY25-Q1-SP7	Nov-24	5	1	25	21	2	4	2	10	3
FY25-Q2-SP1	Dec-24	16	3	80	45	5	9	3	21	7
FY25-Q2-SP2	Dec-24	18	3	90	48	5	10	4	22	7
FY25-Q2-SP3	Jan-25	19	4	95	55	6	11	4	25	9
FY25-Q2-SP4	Jan-25	15	3	75	48	5	10	4	22	7
FY25-Q2-SP5	Feb-25	12	2	60	40	5	8	3	18	6

Tabla 25
CSAT (Mensual) - Fábrica de Software

Sprint	Mes	Encuestas enviadas	Encuestas respondidas	Muy satisfecho	Satisfecho	CSAT	Tasa de respuestas
FY25-Q1-SP5	Oct-24	45	35	4	7	31.4%	77.8%
FY25-Q1-SP6	Nov-24	45	32	4	7	34.4%	71.1%
FY25-Q1-SP7	Nov-24	25	21	2	4	28.6%	84.0%
FY25-Q2-SP1	Dec-24	80	45	5	9	31.1%	56.3%
FY25-Q2-SP2	Dec-24	90	48	5	10	31.3%	53.3%
FY25-Q2-SP3	Jan-25	95	55	6	11	30.9%	57.9%
FY25-Q2-SP4	Jan-25	75	48	5	10	31.3%	64.0%
FY25-Q2-SP5	Feb-25	60	40	5	8	32.5%	66.7%

Lectura:

- El CSAT por sprint de la fábrica de software se ubica entre 28.6% y 34.4% para la muestra analizada en base a un rango de 32 a 55 opiniones por sprint.
- Se evidencia una asociación negativa con respecto a los fallos post implementación (-0.66) y no se evidencia una relación negativa con respecto a los tiempos de entrega de requisitos (0.08).
- Adicionalmente, es necesario precisar que predomina las respuestas “insatisfechas” en todos los sprints lo cual es coherente con el CSAT anual de la fábrica de software indicado en la sección 1.2.2

Conclusión:

- El resultado CSAT de los sprints analizados nos muestra que los usuarios que utilizan el servicio de fábrica se encuentran poco satisfechos con el servicio recibido en cada sprint que se les ha consultado.
- El resultado de la asociación negativa en estos sprints nos sugiere que la percepción de los usuarios en la calidad de las entregas (evitar fallos) pesa más que el cumplimiento en los tiempos de entrega de requisitos.

En conclusión:

- Con respecto al indicador NPS, los usuarios no recomiendan tanto al servicio de mesa de ayuda como el de fábrica de software. Los usuarios esperan que se cumplan con las fechas de entrega de sus tickets y requisitos y se brinde un servicio de calidad sin tantas reaperturas de tickets y requisitos por atenciones mal realizadas.
- Con respecto al indicador CSAT, los usuarios se encuentran muy poco satisfechos con el servicio de mesa de ayuda al igual que con el de fábrica de software. La resolución efectiva de sus solicitudes de atención es muy importante dado el impacto que tienen en su operativa diaria para atender a sus clientes.
- Un factor importante es la capacidad operativa de la mesa de ayuda tal cual se indica en la sección 3.7.1 Descripción del Proceso de Atención de Incidencias (Mesa de ayuda) se cuenta con 3 agentes y un promedio de atención de 1.5 horas por ticket y de la fábrica de software como se menciona en la sección 3.7.2 Descripción del Proceso de Atención de Requisitos (Fábrica de Software) se cuenta con 6 recursos asignados y un promedio de 14 horas por requisito, lo cual es insuficiente para la demanda actual para ambos servicios.
- Finalmente, se muestra claramente la necesidad de implementar nuevos modelos de atención para la mesa de ayuda y la fábrica de software para que se puedan gestionar de manera efectiva los ciclos de atención, reducir la tasa de fallos mejorando la calidad de la entrega en ambos servicios, reducir los tiempos de respuestas, revisar la capacidad de atención y asegurar el cumplimiento de los SLAs.

3.9 Objetivos

3.9.1 Objetivo General

Optimizar la gestión de incidencias y gestión de requisitos en Kred C&S mediante la implementación una gestión basada en ITIL en la mesa de ayuda e integrándolo con mejores prácticas agiles en la fábrica de software, con el fin de mejorar los tiempos de respuesta, reducir la reincidencia de problemas y generar altos niveles de satisfacción de los usuarios.

3.9.2 Objetivos Secundarios

- Objetivo 1: Describir los procesos actuales de atención de incidencias y requisitos de software mediante el modelado de procesos y levantamiento de información operativa.

  Este objetivo busca identificar cómo funcionan actualmente los procesos clave de TI relacionados con la gestión de incidencias (mesa de ayuda) y gestión de requisitos (fábrica de software) incluyendo un modelado BPMN.

- Objetivo 2: Identificar los puntos de dolor a los procesos actuales de gestión de incidencias (mesa de ayuda) y gestión de requisitos (fábrica de software)

  Este objetivo busca identificar las deficiencias a nivel de gestión de los procesos claves de TI relacionados con la atención de incidencias y desarrollo de requisitos.

- Objetivo 3: Diseñar un modelo de gestión para la mesa de ayuda basada en ITIL, así como también un modelo de gestión para la fábrica de software basado en desarrollo de software ágil, ambas adaptadas a la realidad de la empresa.

  Construir un modelo de optimización que integre buenas prácticas de ITIL y desarrollo de software ágil, adaptado a la realidad operativa de la empresa, con enfoque en eficiencia y alineamiento estratégico.

3.9.3 Indicadores de logro de cada objetivo

Objetivo 1: Describir los procesos actuales de atención de incidencias y requisitos de software mediante el modelado de procesos y levantamiento de información operativa.

- Indicador
    - % CSAT (Anual) – Servicio de Mesa de Ayuda
    - % CSAT (Anual) – Servicio de Fábrica de Software
- Entregable
    - Diagrama de los procesos actuales (As-Is) con su respectivo análisis.

Objetivo 2: Identificar los puntos de dolor e indicadores a los procesos actuales de atención de incidencias (mesa de ayuda) y requisitos de software (fábrica de software).

- Indicador
    - NPS Interno por Periodo (mes) – Servicio de Mesa de Ayuda:
    - CSAT (Mensual) – Servicio de Mesa de Ayuda
    - NPS Interno por Sprint- Servicio de Fábrica de Software
    - CSAT (Mensual) – Servicio de Fábrica de Software
- Entregable
    - Puntos de Dolor del Servicio de Mesa de Ayuda y el Servicio de Fabrica de Software
    - Métricas e Indicadores del Servicio de Mesa de Ayuda y el Servicio de Fábrica de Software

Objetivo 3: Diseñar un modelo de gestión para la mesa de ayuda basada en ITIL, así como también un modelo de gestión para la fábrica de software basado en desarrollo de software ágil, ambas adaptadas a la realidad de la empresa.

- Indicador
    - % CSAT (Anual) – Servicio de Mesa de Ayuda
    - % CSAT (Anual) – Servicio de Fábrica de Software
- Entregable
    - Diagrama de modelo de gestión adaptado al servicio de mesa de ayuda
    - Diagrama de modelo de gestión adaptado al servicio de fábrica de software.

4. Desarrollo de la Solución

4.1 Gestión del Proyecto

4.1.1 Plan de Gestión del Alcance

Objetivo	Definir los límites, entregables y criterios necesarios para controlar el alcance del proyecto de implementación del modelo integrado ITIL v4–Scrum en la Mesa de Ayuda y Fábrica de Software.
Alcance	Incluye: diagnóstico del modelo actual, diseño del modelo ITIL v4–Scrum, implementación piloto, capacitación, evaluación, documentación y entrega final. No incluye: adquisición de hardware ni contratación de personal adicional permanente.
Responsable	• Jefe de Proyecto (responsable general) • Consultor agile • Consultor ITIL
Actividades	• Identificación de requerimientos. • Elaboración de EDT (WBS). • Validación del alcance. • Control de cambios del alcance.
Supuestos	• Acceso oportuno a la información interna. • Participación y colaboración serán activas y de alto valor de los interesados involucrados en todas las fases del proyecto de tesis a fin de reducir la brecha evolutiva del alcance. • Aprobación del plan de trabajo por parte de los interesados del proyecto en fases iniciales del mismo. • Las métricas obtenidas sobre el rendimiento de la mesa de ayuda y de la fábrica de software reflejan, de forma cercana, la realidad en la atención de solicitudes de atención. • El mapeo de las tareas, su correlación y la descripción reflejan, en lo más cercano, a la forma en que se realizan los trabajos en la empresa objeto de estudio.

	• Los niveles de satisfacción de los clientes internos captados en métricas están por debajo del promedio aceptable para una empresa perteneciente a una organización transnacional. • No esta difundida forma de desarrollo ágil para la fábrica de software e ITIL no se aplican eficientemente en la empresa objeto de estudio.
Restricciones	Plazo máximo de 6 meses.
Riesgos	Cambios de requerimientos sin control formal.

4.1.2 Alcance del Proyecto

Alcance del proyecto	Alcance del proyecto de tesis	Alcance del producto
Planificación, estimación, ejecución de una propuesta de implementación de un modelo de gestión para la mesa de ayuda y fábrica de software.	Elaboración de una propuesta de implementación de un modelo de servicio basado en herramientas ágiles y de un marco de trabajo robusto como ITIL v4.	Modelo de servicios basado en desarrollo ágil e ITIL para la mesa de ayuda y la fábrica de software.

4.1.3 Plan de Gestión de los Riesgos

Objetivo	Identificar, analizar, mitigar y monitorear los riesgos del proyecto.
Alcance	Incluye riesgos operativos, tecnológicos, humanos y de alcance.
Responsables	• Jefe de Proyecto • Consultor de riesgos
Actividades	• Identificación. • Análisis cualitativo. • Evaluación de probabilidad e impacto. • Respuestas. • Monitoreo
Supuestos	Participación de áreas

Restricciones	Tiempo máximo de mitigación: 30 días.

4.1.4 Riesgos

La Tabla 26 emplea una matriz cualitativa de probabilidad e impacto adaptada a las condiciones del proyecto. Los riesgos críticos exigen definir y aplicar medidas preventivas antes de iniciar o continuar la ejecución; los riesgos altos requieren tratamiento obligatorio y seguimiento frecuente; los riesgos medios deben evaluarse considerando el costo y la viabilidad de las respuestas, manteniendo control sobre sus variables; y los riesgos bajos pueden aceptarse con vigilancia periódica, salvo que cambien su probabilidad o impacto. La escala y los criterios de decisión fueron elaborados por los autores tomando como referencia el enfoque de evaluación cualitativa del riesgo del Project Management Institute (2021).

Tabla 26
Calificación del Riesgo

			GRAVEDAD (IMPACTO)				
			MUY BAJO	BAJO	MEDIO	ALTO	MUY ALTO
			1	2	3	4	5
PROBABILIDAD	MUY ALTA	5	5	10	15	20	25
	ALTA	4	4	8	12	16	20
	MEDIA	3	3	6	9	12	15
	BAJA	2	2	4	6	8	12
	MUY BAJA	1	1	2	3	4	5

	Riesgo muy grave. Requiere medidas preventivas urgentes. No se debe iniciar el proyecto sin la aplicación de medidas preventivas urgentes y sin acotar sólidamente el riesgo.
	Riesgo importante. Medidas preventivas obligatorias. Se deben controlar fuertemente las variables de riesgo durante el proyecto.
	Riesgo apreciable. Estudiar económicamente si es posible introducir medidas preventivas para reducir el nivel de riesgo. Si no fuera posible, mantener las variables controladas.
	Riesgo marginal. Se vigilará, aunque no requiere medidas preventivas de partida.

Nota. Elaborado a partir de *Guía de los fundamentos para la dirección de proyectos (Guía del PMBOK®)*, por Project Management Institute, 2021.

Tabla 27
Riesgos

Nro.	Descripción del riesgo	Prob	Imp	Valor riesgo	Estrategia de respuesta al riesgo	Responsable
R01	Existe el riesgo de que la propuesta no sea aceptada debido a la reticencia hacia ITIL, provocada por la percepción de su complejidad, lo que podría generar retrasos o necesidad de reformulación del enfoque.	3	5	15	Evitar: Iniciar plan de capacitación sobre los aspectos importantes de ITIL v4	Equipo TESIS
R02	Existe el riesgo de que se produzca un retraso en la implementación del proyecto debido a la complejidad de la propuesta o a una falta de entendimiento por parte de los involucrados, lo que podría afectar el cronograma y los objetivos del proyecto.	3	3	9	Mitigar: Elaborar pruebas de concepto de la propuesta de desarrollo ágil para la fábrica de software + ITIL para la mesa de ayuda	Equipo TESIS
R03	Existe el riesgo de que se presenten retrasos en la ejecución del proyecto debido a la falta de apoyo por parte de los interesados clave, lo que podría afectar la toma de decisiones, la asignación de recursos y la aceptación del cambio.	2	4	8	Mitigar: Elaborar planes de involucramiento y seguimiento a los interesados	Equipo TESIS
R04	Existe el riesgo de que la propuesta sea rechazada debido a la falta de	3	5	15	Incluir en la propuesta una lista	Equipo TESIS

	herramientas facilitadoras para su implementación, lo que podría afectar la viabilidad del proyecto.				de herramientas recomendadas. Evaluar alternativas accesibles o temporales. Mostrar cómo la inversión en herramientas mejora la efectividad del proyecto.	
R05	Existe el riesgo de que se presenten retrasos en la implementación del proyecto debido a un sub-dimensionamiento en los costos de la propuesta, lo que podría generar necesidad de ajustes presupuestarios, recortes o redefinición del alcance.	2	4	12	Mitigar validar estimaciones con expertos financieros y técnicos. Incluir márgenes de contingencia en el presupuesto. Realizar revisiones de la propuesta con enfoque en costos ocultos. Priorizar entregables claves si se requiere ajuste de alcance.	Equipo TESIS
R06	Existe el riesgo de que cambios en la legislación o en las políticas gubernamentales puedan afectar el desarrollo o la implementación del proyecto, lo que podría generar la necesidad de ajustes en los requisitos o retrasos en la ejecución	2	2	4	Mitigar: Monitorear cambios legislativos y regulatorios constantemente. Incluir en el cronograma márgenes de tiempo para ajustes si se prevé que haya cambios. Consultar con asesores legales para	Equipo RRHH

					prever cambios potenciales. Mantener una relación cercana con stakeholders gubernamentales.	
R07	Existe el riesgo de que las fluctuaciones en los tipos de cambio afecten los costos del proyecto, lo que podría generar aumentos en el presupuesto o ajustes en los pagos a proveedores internacionales	3	2		Mitigar: Utilizar coberturas cambiarias (hedging) para fijar tipos de cambio. Negociar contratos en moneda local o establecer cláusulas de ajuste por variaciones cambiarias. Monitorear los tipos de cambio y la situación económica global para anticipar posibles fluctuaciones. Planificar un margen de contingencia en el presupuesto para cubrir aumentos imprevistos.	Equipo TESIS
R08	Existe el riesgo de que la aparición de una pandemia global afecte la continuidad del proyecto, lo que podría provocar interrupciones en la operación, retrasos significativos, y aumento	1	3	3	Aceptar: Desarrollar planes de continuidad de negocio y contingencia ante situaciones de emergencia global.	Equipo RRHH

	de los costos debido a restricciones de movilidad, enfermedad del personal o cierre de proveedores				Implementar trabajo remoto y medidas sanitarias de seguridad para los equipos. Diversificar proveedores y realizar acuerdos contractuales que contemplen posibles interrupciones globales. Incluir márgenes de contingencia para cubrir sobrecostos asociados a una crisis global.	
R09	Existe el riesgo de que un desastre natural, como un terremoto o sismo, afecte la ejecución del proyecto, provocando daños a infraestructuras, interrupciones en las actividades y retrasos en el cronograma debido a la necesidad de rehabilitación y recuperación.	2	3	6	Aceptar: Desarrollar planes de emergencia y continuidad de negocio en caso de desastres naturales. Evaluar y asegurar que las infraestructuras del proyecto sean resistentes a desastres naturales (diseño antisísmico, medidas de seguridad). Asegurar el proyecto contra desastres naturales mediante seguros adecuados.	Equipo RRHH

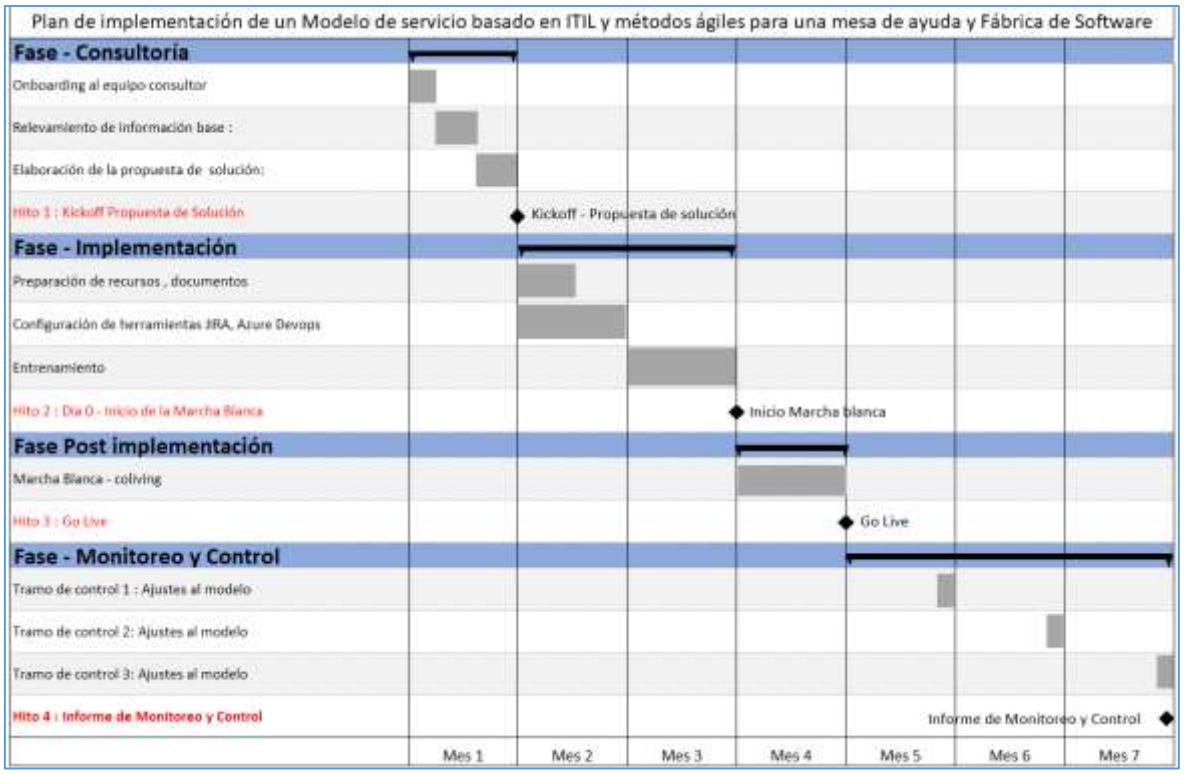
					Tener acuerdos con proveedores alternativos y redes de comunicación para mantener la continuidad en caso de interrupciones.	

4.1.5 Gestión del Cronograma

Objetivo	Definir cómo se planifica, gestiona, controla y actualiza el cronograma.
Alcance	incluye fases, actividades, dependencias, secuencias e hitos.
Responsables	• Jefe de Proyecto (planificación y control) • Equipo consultor (actualización de actividades)
Actividades	• Definir actividades. • Secuenciar actividades. • Estimar esfuerzo y duración. • Construir cronograma tipo Gantt. • Controlar el avance.

4.1.6 Cronograma

Figura 19
Cronograma



4.1.7 Hitos

Tabla 28
Hitos

Hitos	Funciones	Entregable
Presentación del Kick off del proyecto	Revisión de la investigación, extracción de los factores críticos de éxito, afinamiento de hitos y del plan de trabajo Captura, medición, validación de indicadores, establecimiento del AS IS Análisis, diseño y modelamiento del TO BE, preparación de propuesta de solución Planificación, provisión adquisición de elementos a utilizar en el nuevo modelo Presentación del plan de trabajo.	Kick off del proyecto

Hitos	Funciones	Entregable
Inicio de la Marcha blanca	Levantamiento, aprobación, aplicación, prueba y ajustes en JIRA y Azure DevOps Planificación, preparación, ejecución, regulación de sesiones y material para la aceptación del modelo	Dia 0 – inicio de la marcha blanca
GO live	Ejecución de la estrategia de implementación: co-living para ajuste del modelo implementado	Informe Go Live
Monitoreo y control	Preparación, seguimiento del backlog de ajustes al modelo implementado	Informe de control y monitoreo

4.1.8 Roles y Responsabilidades

Tabla 29
Roles y Responsabilidades

Rol	Responsabilidad
Jefe de Proyecto	Profesional con certificación PMP y conocimiento de procesos, ITIL y desarrollo ágil. Lidera la implementación del proyecto, es un facilitador para el resto del equipo. Gestiona los costos del proyecto y la rentabilidad de este.
Consultor ITIL	Profesional con certificación ITIL y conocimiento de procesos, analiza, diseña e implementa el nuevo modelo de la mesa de ayuda basándose en recomendaciones y mejores prácticas ITIL.
Consultor Desarrollo Ágil	Profesional con certificación en desarrollo ágil y conocimiento de procesos. Analiza, diseña e implementa el nuevo modelo de la fábrica de software basándose en recomendaciones y mejores prácticas ágiles y Scrum.

4.1.9 Limitaciones

- Presupuesto limitado para las herramientas a usar en la fábrica de software y mesa de ayuda.
- Ventanas de tiempo reducidas de implementación dado que la mesa de ayuda no puede ser interrumpida
- Escasos recursos especializados en servicios ITIL en el mercado nacional.

4.2 Desarrollo de la Solución

La presente sección desarrolla la propuesta integral de prestación y desarrollo de servicios que operativiza la estrategia digital mediante un modelo unificado ITIL v4 + Desarrollo Ágil.

La presente sección desarrolla una propuesta integrada de operación y desarrollo compuesto por dos modelos complementarios, el prime modelo alineado a la Mesa de Ayuda basado en ITIL4, que estandariza la gestión de incidentes en JIRA pasando por su clasificación, priorización, atención y escalamiento controlado, el segundo modelo destinado a la Fábrica de Software se fundamenta en el desarrollo ágil que regula la gestión de portafolio, gestión de requisitos, la ejecución por sprints y la operación usando Azure DevOps.

Con estos modelos logramos la integración entre ambos servicios, buscamos acelerar el time-to-value, elevar la satisfacción del usuario y alinearlo con los objetivos estratégicos de Kred C&S.

Como se pudo observar en la definición del problema, Kred C&S no cuenta con un adecuado modelo de atención para la mesa de ayuda y la fábrica de software, a continuación, presentamos el modelo para ambos servicios.

4.2.1 Modelo de gestión de la mesa de ayuda (Incidentes N1 - N2)

El modelo propuesto para la mesa de ayuda se fundamenta en las prácticas de gestión de incidentes de ITIL v4, integrando además un enfoque adaptativo que combina la operación de un Nivel 1 (N1) tradicional con un Nivel 2 (N2) orientado al desarrollo ágil de software. Esta dualidad permite atender tanto incidentes recurrentes como aquellos que requieren ajustes o desarrollos específicos en las aplicaciones de la organización.

El modelo adaptado para el servicio de Mesa de Ayuda (N1-N2) es un flujo end-to-end que inicia con el registro en Jira de la incidencia (oferta de servicio) y culmina con el cierre validado y la inducción de conocimiento a la mesa, sin N3.

El agente N1 clasifica y prioriza mediante la matriz de impacto–urgencia, aplica procedimientos estándar para casos de baja complejidad y mantiene comunicación proactiva con el usuario.

Cuando el ticket requiere cambios en software y/o acceso especializado a datos, N1 deriva a N2 (Fábrica de Software); el Project Manager toma conocimiento y, vía comité de priorización, se ordena en el backlog para su inclusión en el ciclo de desarrollo ágil.

En Fábrica, el Project Manager asigna el ticket al recurso N2, asegura su atención en el sprint y remueve impedimentos.

El equipo desarrolla, QA verifica con evidencia y el Líder Técnico vela por arquitectura, requisitos no funcionales y buenas prácticas de desarrollo.

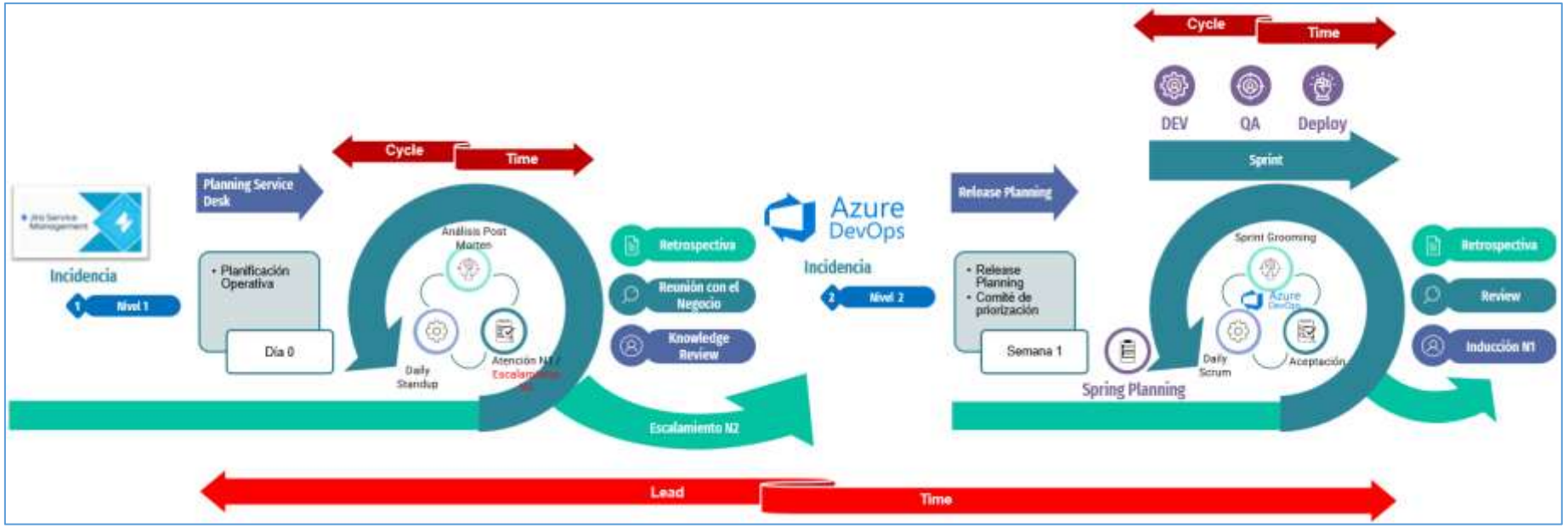
La trazabilidad se mantiene en Jira y Azure DevOps (ticket ⇄ historia/bug).

Asimismo, se precisa que no todo ingreso es incidencia:

- Las Solicitudes de Servicio se tramitan en N1 y no pasan a Scrum.
- En cambio, las Solicitudes de Cambio se derivan a Fábrica para su conversión en requisito (historia del producto backlog en Azure DevOps) y su gestión con Scrum, con lo cual se diferencia explícitamente Incidenciaz-N2 de requisito y se conserva la trazabilidad Jira↔Azure DevOps.

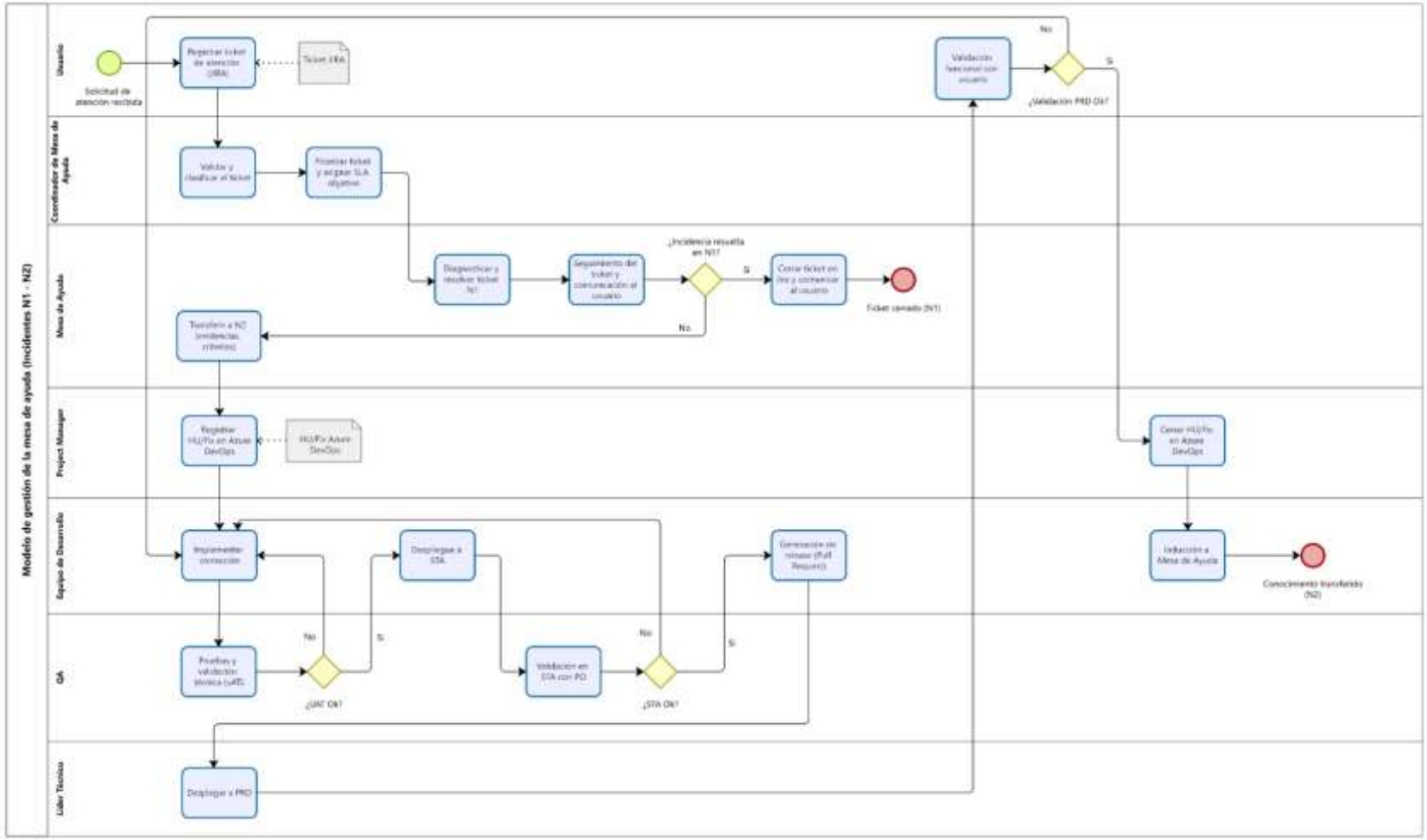
Disclaimer: ITIL 4 no prescribe un número fijo de niveles de atención; el esquema N1–N2–N3 es solo una práctica común. En esta propuesta la empresa opera con N1 (Mesa de Ayuda) y N2 (Fábrica de Software); N3 no se considera como nivel operativo separado.

Figura 20
Modelo Adaptativo Mesa de Ayuda N1 – N2



- TO-BE del modelo de gestión de la mesa de ayuda (Incidentes N1 - N2)

Figura 21
Modelo de gestión de la mesa de ayuda (Incidentes N1 - N2)



- Descripción del modelo de gestión de la mesa de ayuda (Incidentes N1 - N2)

Nombre del Proceso	
Atención de Incidencias de Mesa de Ayuda (Nivel 1 - Nivel 2)	
Objetivo:	Estandarizar la atención de incidentes desde su registro en Jira hasta su resolución en N1 o su escalamiento a fábrica de software (N2), asegurando trazabilidad y control de calidad en la corrección. Garantiza cierre operativo y transferencia de conocimiento a Mesa de Ayuda.
Alcance:	Cubre desde el registro de la incidencia y su priorización con SLA, hasta diagnóstico/solución N1 o escalamiento a N2 con atención en Azure DevOps, despliegue y validación en producción. Incluye inducción/gestión del conocimiento posterior a la liberación.
Recursos Involucrados:	Personal de Mesa de Ayuda (5 agentes)
	Fábrica de Software
	JIRA / Azure DevOps
	Estaciones de usuario
	Ambientes DEV / UAT / STA / PRD
	Infraestructura de red y conectividad

Tabla 30
Descripción del modelo de gestión de la mesa de ayuda N1 – N2

ID	Entrada	Actividad	Salida	Descripción	Responsable
1	Ticket de Atención	Registrar ticket en Jira (Incidencia)	Ticket de atención registrado en Jira	Se registra el ticket de atención en Jira como oferta de servicio “Incidencia” con los datos necesarios para iniciar la revisión.	Usuario
2	Ticket registrado	Validar y clasificar el ticket	Ticket validado	El Coordinador de Mesa de Ayuda realiza una validación mínima del ticket registrado para definir el tratamiento.	Coordinador de Mesa de Ayuda
3	Ticket validado	Priorizar ticket y asignar SLA objetivo	Ticket priorizado (Prioridad + SLA)	Priorización impacto/urgencia y asignación de SLA; se deriva a N1.	Coordinador de Mesa de Ayuda
4	Ticket priorizado	Diagnosticar y resolver ticket N1	Ticket de atención en curso (N1)	Diagnostica y resuelve en N1 las incidencias, aplicando procedimientos estándar, registrando el resultado y acciones en JIRA.	Agente N1
5	Ticket de atención en curso (N1)	Realizar seguimiento al ticket y comunicar al usuario	Ticket de atención en curso (seguimiento)	El Agente N1 actualiza el ticket con avances, control de SLA/estados.	Agente N1
6	Ticket de atención en curso	¿Incidencia resuelta en N1?	Ruta N1 / Ruta N2	Si no se resuelve en N1 o cumple criterios de complejidad, se continúa a escalamiento N2.	Coordinador de Mesa de Ayuda / Agente N1

ID	Entrada	Actividad	Salida	Descripción	Responsable
7	Resuelto en N1	Cerrar ticket en Jira y comunicar al usuario	Ticket atendido (cerrado en Jira)	El Agente N1 cierra el ticket en Jira y comunica al usuario el fin de la atención. Si aplica, se actualiza base de conocimientos y se ejecuta retrospectiva interna N1.	Agente N1
8	No resuelto / complejo	Transferir a N2 (evidencias, criterios)	Ticket de atención transferido	El Agente N1 transfiere la incidencia con información/evidencias necesarias para habilitar el registro en Azure DevOps.	Agente N1
9	Ticket transferido	Registrar escalamiento en Azure DevOps y definir plan	HU/Fix registrado en Azure DevOps (backlog)	El Project Manager acepta el escalamiento, registra HU/Fix en backlog de Fábrica y define el plan de atención/corrección para su inclusión en el ciclo ágil.	Project Manager
10	HU/Fix en backlog	Implementar corrección (desarrollo)	Corrección implementada	Desarrollo ejecuta diseño/construcción, pruebas unitarias/integración y actualiza documentación técnica si corresponde.	Equipo de Desarrollo
11	Corrección implementada	Pruebas y validación técnica	Resultado UAT	QA ejecuta pruebas, aporta evidencia y reporta resultados.	QA
12	Resultado UAT	¿UAT OK?	Ruta corrección / Ruta STA	Si no OK, retorna a desarrollo; si OK, continúa a STA.	QA
13	UAT OK	Despliegue a STA	HU/Fix desplegado STA	DEV despliega a STA según procedimiento interno.	Equipo de Desarrollo
14	STA desplegado	Validación en STA con PO	Resultado STA	Validación funcional en STA (QA con PO, según tu práctica).	QA / Product Owner
15	Resultado STA	¿STA OK?	Ruta corrección / Ruta Release	Si no OK, retorna a desarrollo; si OK, continúa a release.	QA
16	STA OK	Generación de release (Pull Request)	Release aprobado	DEV genera PR para release.	Equipo de Desarrollo
17	Release aprobado	Deploy a PRD	Cambio desplegado PRD	Despliegue a producción.	Líder Técnico
18	Cambio en PRD	Validación funcional en PRD (usuario)	Resultado PRD	Validación funcional en producción con el usuario.	Usuario
19	Resultado PRD	¿Validación PRD OK?	Ruta cierre / Ruta corrección	Si OK, cerrar HU/Fix; si no, registrar ticket de corrección (nuevo ciclo).	Usuario
20	PRD OK	Cerrar HU/Fix en Azure DevOps	HU/Fix cerrado ADO	Cierre formal del HU/Fix en ADO.	Project Manager

ID	Entrada	Actividad	Salida	Descripción	Responsable
21	HU/Fix cerrado	Inducción a Mesa de Ayuda	Conocimiento transferido a N1	Transferencia de conocimiento a N1 post release.	Equipo de Desarrollo / Líder Técnico

- Ceremonias

Tabla 31
Ceremonias Mesa de Ayuda N1 – N2

Ceremonia	Objetivo	Frecuencia	Ámbito
Planificación Operativa	Alinear la operación de N1: metas de la quincena, capacidad, reglas de prioridad (Impacto × Urgencia) y criterios de escalamiento a N2.	1 vez, inicio de mes	Mesa de ayuda
Daily Standup	Sincronizar colas, resolver bloqueos, confirmar P1–P2 y escalamientos; revisar incidentes mayores si los hubiera.	Durante el ciclo	Mesa de ayuda
Atención N1 / Escalamiento N2	Realizar diagnóstico y resolución en N1. Si el caso requiere cambio de software, base de datos o acceso especializado, ejecutar el escalamiento a N2 preparando evidencias y manteniendo la comunicación de SLA.	Durante el ciclo	Mesa de ayuda
Knowledge Review	Actualizar base de conocimiento a partir de tickets resueltos y cambios liberados.	Durante el ciclo	Mesa de ayuda
Análisis Post Mortem	Identificar causa raíz y acciones preventivas tras incidentes críticos.	Durante el ciclo	Mesa de ayuda
Retrospectiva Mesa de Ayuda	Reflexionar sobre lo ocurrido en la quincena de operación N1 y acordar mejoras de proceso, calidad de datos y coordinación con N2.	Fin del ciclo	Mesa de ayuda
Reunión con el Negocio	Revisar incidencias relevantes, expectativas de SLA, riesgos y cambios planificados con representantes del área usuaria.	1 vez, fin de mes	Mesa de ayuda
Comité de Priorización	Revisar tickets de N2 y requisitos; decidir la incorporación al backlog y la prioridad para su ejecución ágil.	Inicio del sprint	Fábrica de Software
Sprint Planning	Seleccionar trabajo del sprint y fijar objetivos claros.	Inicio del sprint	Fábrica de Software
Daily Scrum	Coordinar ejecución, visibilizar WIP y remover impedimentos.	Durante el sprint	Fábrica de Software
Sprint Grooming	Profundizar historias/bugs, depurar y ordenar el backlog del sprint, verificando DoR y dependencias.	Durante el sprint	Fábrica de Software
Aprobación de Cambio	Verificar que el requisito y los de N2 cumplen criterios de aceptación, validar con el PO	Durante el sprint	Fábrica de Software

Ceremonia	Objetivo	Frecuencia	Ámbito
Deploy	Ejecutar el despliegue planificado sin interrumpir la operación.	Durante el sprint	Fábrica de Software
Sprint Retrospectiva	Identificar mejoras del proceso y acordar acciones concretas.	Fin del sprint	Fábrica de Software
Sprint Review	Presentar incremento y recoger feedback de negocio/usuarios.	Fin del sprint	Fábrica de Software
Inducción Mesa de Ayuda	Transferir conocimiento del cambio y preparar a N1 para el soporte post-release.	Fin del sprint	Fábrica de Software

- Catálogo de Servicio:

  El catálogo de servicios expone el alcance operativo y las reglas de elegibilidad de la mesa de ayuda como punto único de contacto (SPOC). El cuadro organiza las entradas por dominios y canales de Jira y delimita su alcance a cuatro ofertas de servicio:

  - Incidencias (interrupciones o degradaciones del servicio)
  - Solicitudes de servicio (altas, accesos, configuraciones)
  - Solicitudes de cambio (requisitos que requieren intervención de la fábrica de software)
  - Incidentes de seguridad (afectaciones a confidencialidad, integridad y disponibilidad).

  Cada entrada vincula evidencias requeridas, su SLA de respuesta y su solución aplicable, con ello se habilita la trazabilidad desde la necesidad de atención hasta la entrega; logrando: enfoque de valor, visibilidad y co-creación con las áreas usuarias.

  Cuando la atención requiera intervención de N2, el catálogo especifica el criterio de escalamiento al nivel de fábrica de software y la derivación del ticket de Jira al flujo ágil (creación del “work ítem” en Azure DevOps), dejando trazabilidad con conocimiento del Comité de Priorización.

Tabla 32
Oferta de Servicio

Oferta de Servicio	Descripción
Incidentes	Interrupción no planificada o reducción de la calidad de un activo o servicio de TI que afecta a uno o más usuarios de la organización.
Solicitud de Servicio	Solicitud para obtener apoyo y/o soporte tecnológico relacionado a un activo de TI, sistemas de información o estaciones de trabajo.
Solicitud de Cambio	Solicitud para realizar o implementar una nueva aplicación o cambio específico en estos.
Incidencia de Seguridad	Evento o acción que se crea pueda afectar a la seguridad de la información del usuario o la empresa.

Tabla 33
Catálogo de Servicios

Código	Servicio	Oferta de Servicio	Regla de elegibilidad	Escalamiento N2
IN-001	Alta / Baja de usuarios	Incidentes	Solicitado por el gerente de RRHH con aprobación del gerente solicitante del alta / baja.	
IN-002	Aplicaciones de software	Incidentes	Todos los empleados tienen derecho a solicitarlo	(*)
IN-003	Backup	Incidentes	Todos los empleados tienen derecho a solicitar y recibir este servicio de manera automática.	
IN-004	Cuentas de correo	Incidentes	Solicitado por el gerente de RRHH con aprobación del gerente solicitante del alta / baja.	
IN-005	Equipo de computo	Incidentes	Solicitado por el gerente de RRHH con aprobación del gerente solicitante del alta / baja.	
IN-006	Gestión de accesos	Incidentes	Solicitado por el gerente de RRHH con aprobación del gerente solicitante del alta / baja.	
IN-007	Office	Incidentes	Todos los empleados tienen derecho a solicitar con aprobación de su gerente inmediato y el jefe de infraestructura y ciberseguridad.	
IN-008	Sistema Clientes	Incidentes	Todos los empleados tienen derecho a solicitar con aprobación de su gerente inmediato.	
IN-009	Sistema BackEnd	Incidentes	Todos los empleados tienen derecho a solicitar con aprobación de su gerente inmediato.	
SS-001	Alta / Baja de usuarios	Solicitud de Servicio	Solicitado por el gerente de RRHH con aprobación del gerente solicitante del alta / baja.	
SS-002	Aplicaciones de software	Solicitud de Servicio	Todos los empleados tienen derecho a solicitarlo	(*)
SS-003	Backup	Solicitud de Servicio	Todos los empleados tienen derecho a solicitar y recibir este servicio de manera automática.	
SS-004	Cuentas de correo	Solicitud de Servicio	Solicitado por el gerente de RRHH con aprobación del gerente solicitante del alta / baja.	
SS-005	Equipo de computo	Solicitud de Servicio	Solicitado por el gerente de RRHH con aprobación del gerente solicitante del alta / baja.	

Código	Servicio	Oferta de Servicio	Regla de elegibilidad	Escalamiento N2
SS-006	Gestión de accesos	Solicitud de Servicio	Solicitado por el gerente de RRHH con aprobación del gerente solicitante del alta / baja.	
SS-007	Office	Solicitud de Servicio	Todos los empleados tienen derecho a solicitar con aprobación de su gerente inmediato y el jefe de infraestructura y ciberseguridad.	
SS-008	Sistema Clientes	Solicitud de Servicio	Todos los empleados tienen derecho a solicitar con aprobación de su gerente inmediato.	
SS-009	Sistema BackEnd	Solicitud de Servicio	Todos los empleados tienen derecho a solicitar con aprobación de su gerente inmediato.	
SC-001	Sistema Beneficiarios	Solicitud de Cambio	Los usuarios con rol PO tienen derecho a solicitarlo con aprobación de su gerente inmediato	
SC-002	Sistema Clientes	Solicitud de Cambio	Los usuarios con rol PO tienen derecho a solicitarlo con aprobación de su gerente inmediato	
SC-003	Sistema BackEnd	Solicitud de Cambio	Los usuarios con rol PO tienen derecho a solicitarlo con aprobación de su gerente inmediato	
IS-001	Acceso no autorizado	Incidente de Seguridad	Todos los empleados tienen derecho a solicitar y recibir este servicio de manera automática.	
IS-002	DoS	Incidente de Seguridad	Solicitado por el Jefe de Ingraestructura y Ciberseguridad.	(*)
IS-003	Destrucción o daño de la información	Incidente de Seguridad	Todos los empleados tienen derecho a solicitar y recibir este servicio de manera automática.	
IS-004	Filtración de datos	Incidente de Seguridad	Solicitado por el Jefe de Ingraestructura y Ciberseguridad.	(*)
IS-005	Pishing	Incidente de Seguridad	Todos los empleados tienen derecho a solicitar y recibir este servicio de manera automática.	
IS-006	Malware	Incidente de Seguridad	Todos los empleados tienen derecho a solicitar y recibir este servicio de manera automática.	
IS-007	Robo o perdida de equipo	Incidente de Seguridad	Todos los empleados tienen derecho a solicitar y recibir este servicio de manera automática.	
IS-008	Suplantación de Identidad	Incidente de Seguridad	Todos los empleados tienen derecho a solicitar y recibir este servicio de manera automática.	
IS-009	Vulnerabilidades	Incidente de Seguridad	Solicitado por el Jefe de Ingraestructura y Ciberseguridad.	(*)

(*) Escala a N2 si: requiere cambio/corrección de software, workaround temporal o inaceptable, reportes complejos obtenidos necesariamente de una o más BDs, riesgo operativo/seguridad, dependencia técnica con un requisito.

Derivación: Vía Nivel 1 / JIRA => Nivel 2 / Azure DevOps (Formato: Historia de Usuario/Bug).

- Campos mínimos para un Ticket de Atención:

  Define los campos mínimos obligatorios para un ticket de atención y que se relaciona con los catálogos descritos en la presente sección:

Tabla 34
Ticket de Atención - Campos

Campo	Descripción	Tipos de Servicio	Tabla relacionada
Usuario Solicitante	Usuario que genera la solicitud	Incidentes / Solicitud de Cambio / Incidente de Seguridad / Solicitud de Servicio	
Área solicitante	Unidad que requiere el servicio.	Incidentes / Solicitud de Cambio / Incidente de Seguridad / Solicitud de Servicio	Personal por Área
Servicio del catálogo	Vincula la solicitud a la oferta del catálogo.	Incidentes / Solicitud de Cambio / Incidente de Seguridad / Solicitud de Servicio	Catálogo de Servicios
Descripción breve	Resumen de la solicitud	Incidentes / Solicitud de Cambio / Incidente de Seguridad / Solicitud de Servicio	
Detalle de la solicitud	Requisitos específicos, checklist, etc.	Incidentes / Solicitud de Cambio / Incidente de Seguridad / Solicitud de Servicio	
Urgencia	Rapidez requerida (alta/media/baja).	Incidentes / Solicitud de Cambio / Incidente de Seguridad	Matriz Impacto × Urgencia → Prioridad (N1)
Impacto	Afectación al negocio (alto/medio/bajo).	Incidentes / Solicitud de Cambio / Incidente de Seguridad	Matriz Impacto × Urgencia → Prioridad (N1)
Prioridad (solo lectura)	Derivada de Impacto×Urgencia.	Incidentes / Solicitud de Cambio / Incidente de Seguridad	Matriz Impacto × Urgencia → Prioridad (N1)
SLAs objetivos (solo lectura)	Derivado de Prioridad. Tiempo de cumplimiento comprometido.	Incidentes / Solicitud de Cambio / Incidente de Seguridad	Matriz Impacto × Urgencia → Prioridad (N1), Niveles de Servicio
Adjuntos	Evidencias, documentos, formularios, imágenes, etc.	Incidentes / Solicitud de Cambio / Incidente de Seguridad	
Aprobación requerida	Usuario que aprueba lo solicitado	Incidentes / Solicitud de Cambio / Incidente de Seguridad	

- Roles y Responsabilidades:

  Los cuadros de roles y responsabilidades definen la gobernanza del modelo de mesa de ayuda (N1) y su interfaz con la fábrica de software (N2). La asignación de responsabilidades respeta la segregación de funciones que se videncia en las tablas RACI.

Tabla 35
Roles - Mesa de Ayuda

Roles Mesa de Ayuda	Descripción
Usuario	Registra tickets de incidentes en JIRA, aporta contexto del defecto y es fuente de retroalimentación para mejora continua.
Agente N1	Categoriza, prioriza y resuelve tickets de incidencia, reporta en JIRA, comunica estados y expectativas. Escala a N2 de ser necesario.
Agente N2	Resuelve casos de complejidad media y alta vinculado a las aplicaciones de software de la empresa. El agente N2 es personal de la fábrica de software destinada a la atención de tickets de mesa de ayuda que involucra desarrollo de software y/o acceso a sus bases de datos.
Coordinador de Mesa de Ayuda	Gestiona colas y turnos en la mesa de ayuda, coordina incidentes mayores y prioridades con N2, impulsa mejora continua.
Jefe de Infraestructura	Gestiona la mesa de ayuda y además lidera al interno la estrategia de ciberseguridad, protegiendo la información, controlando accesos y gestionando amenazas.

Tabla 36
Roles - Fábrica de Software

Roles Fábrica de Software	Descripción
Agente N1	Identifica solicitudes registradas como 'Solicitud de cambio' en Jira y deriva al Project Manager para su registro en Azure DevOps, asegurando trazabilidad.
Product Owner	Maximiza valor, define criterios de aceptación, es interlocutor con el negocio y registra/gestiona solicitudes de cambio (requisitos) en Jira para su atención por Fábrica
Stakeholder	Avala el caso de negocio, alinea con la estrategia y aprueba el alcance
Comité de Priorización	Optimiza y prioriza el backlog de requisitos definiendo cuales se van a ejecutar en los sprints y su puesta en marcha. Conformado por el stakeholder, Product Owner, Project Manager y el Jefe de TI.
Project Manager	Dirige la fábrica de software y asume también el rol de Scrum Master, prioriza la línea base del sprint y lo presenta en el comité de priorización, remueve impedimentos, promueve métricas y mejora del proceso.

Roles Fábrica de Software	Descripción
Equipo de Desarrollo	Implementa los desarrollos de software para nuevos productos y mejoras a los ya existentes, realizan actividades de diseño, construcción, pruebas unitarias e integración; actualiza documentación técnica y participa en la revisión del producto.
QA	Define estrategia de pruebas, diseña casos (funcionales y no-funcionales), ejecuta pruebas, aporta evidencia y reporta resultados.
Líder Técnico	Asegura consistencia de arquitectura y cumplimiento de no-funcionales (seguridad, capacidad, disponibilidad); guía decisiones técnicas en el equipo de desarrollo.
Jefe de TI	Dirige el área de TI y su roadmap, además es responsable del desarrollo, mantenimiento y evolución de las plataformas digitales de la empresa.

- Matriz RACI

  La matriz RACI resume el ciclo operativo de la mesa de ayuda:

  (R) Responsable, (A) Aprobador, (C) Consultado, (I) Informado.

Tabla 37
Mesa de Ayuda - Matriz RACI

Actividad / Rol	Usuario	Agente N1	Agente N2	Coord. Mesa Ayuda	Jefe Infra	PO	PM	Equipo Dev	QA	LT
1. Registro del ticket en Jira (Incidencia)	A/R	I	I	I	I	I	I	I	I	I
2. Validación mínima y clasificación del ticket	I	C	I	A/R	C	I	I	I	I	I
3. Priorización (impacto/urgencia) y SLA por ticket	I	C	I	A/R	C	I	I	I	I	I
4. Diagnóstico y resolución N1	I	R	I	A	I	I	I	I	I	I
5. Comunicación/seguimiento al usuario	C	R	I	A	I	I	I	I	I	I
6. Decidir escalamiento a N2	I	R	C	A/R	C	I	I	I	I	I
7. Transferir a N2 (evidencias, criterios)	I	R	C	A	I	I	C	I	I	I
8. Registrar HU/Fix en Azure DevOps y definir plan	I	I	C	I	I	I	A/R	C	C	C
9. Implementar corrección (desarrollo)	I	I	C	I	I	C	A	R	C	R
10. Pruebas y validación técnica (UAT)	C	I	C	I	I	I	I	C	A/R	C

Actividad / Rol	Usuario	Agente N1	Agente N2	Coord. Mesa Ayuda	Jefe Infra	PO	PM	Equipo Dev	QA	LT
11. Despliegue a STA	I	I	C	I	I	I	C	R	C	A
12. Validación en STA con PO	I	I	I	I	I	A/R	I	I	R	I
13. Generación de release (Pull Request)	I	I	I	I	I	I	C	R	I	A
14. Deploy a PRD	I	I	I	I	I	I	C	C	C	A/R
15. Validación funcional en PRD (Usuario)	A/R	I	I	I	I	I	I	I	I	I
16. Cerrar HU/Fix en Azure DevOps	I	I	I	I	I	I	A/R	I	I	I
17. Inducción a Mesa de Ayuda	I	C	C	A	I	I	I	R	C	R

- Usuarios del Servicio de Mesa de Ayuda

  El cuadro presenta el total del personal segmentado por áreas de la empresa Kred C&S que cuentan con autorización para registrar tickets de incidencia en Jira. La estructura de personal no modifica los SLA ni el flujo de atención N1=>N2, sino que define la clasificación de las solicitudes por área para análisis posteriores.

Tabla 38
Personal por Área

Área	Usuarios
Comercios	7
Comercial	30
Recursos Humanos	5
Contabilidad	5
Marketing	5
Productos	5
Comunicaciones	5
Administración	6
Finanzas	10
Operaciones	19
Sistemas	10

- Capacidad Recomendada del Servicio de Mesa de Ayuda

  La capacidad de la mesa de ayuda se dimensiona estimando la carga de trabajo del período y dividiéndola por las horas productivas disponibles, ajustadas por shrinkage (ausencias/pausas) y la ocupación objetivo.

  Este cálculo integra el esfuerzo de atención de N1, el costo de escalar a N2 (preparación y transferencia), corrige la capacidad disponible por shrinkage y por la ocupación objetivo que mantiene la calidad del servicio.

  Formula:

Figura 22
Formula Capacidad del Servicio - Mesa de Ayuda

$$\mathbf{FTE} = \frac{\text{Tickets}_{mes} \times \left(\text{AHT}_{N1} + \%\text{Escala} \times \text{AHT}_{Escala}\right)}{\text{Horas_turno} \times \text{Días_hábiles} \times (1 - \text{shrinkage}) \times \text{Ocupación}}$$

Donde:

- Tickets x Mes = Promedio de tickets atendidos en el mes
- AHT_N1 = Tiempo promedio de atención por ticket
- %Escala = Promedio de tickets derivados a N2 en el mes
- AHT_Escala = Tiempo promedio de N1 para escalar a N2
- Horas_turno = Horas diarias laborables de la mesa de ayuda
- Días_hábiles = Días laborables en el mes de la mesa de ayuda
- Shrinkage = % de turno no disponible para atención de tickets (vacaciones, permisos, pausas, reuniones, ausencias, capacitaciones, etc.)
- Ocupación = % de horas productivas para atención de tickets

Parámetros:

- Tickets x Mes = 400 tickets.
- AHT_N1 = 1.5 horas por ticket.
- %Escala = 70/400 tickets = 0.175
- AHT_Escala = 0.5 horas
- Horas_turno = 8 horas diarias
- Días_hábiles = 21 días al mes
- Shrinkage = 0.20
- Ocupación = 0.80

Figura 23
Resultado Capacidad del Servicio - Mesa de Ayuda

$$\mathbf{FTE} = \frac{400 \times (1.5 + 0.175 \times 0.5)}{8 \times 21 \times (1 - 0.20) \times 0.80} = \mathbf{5.91} \;\Rightarrow\; \mathbf{6\ agentes}$$

Lectura:

- Dada la capacidad actual de la mesa de ayuda se requiere de 6 agentes de Nivel 1 para cubrir la demanda.
- Actualmente la mesa de ayuda cuenta con 3 agentes N1 con un promedio de atención de 1.5 horas por ticket indicado por la empresa.

- Capacidad Recomendada del Servicio de Fabrica de Software

  El siguiente análisis explica la brecha de capacidad en la fábrica de software en la atención que brinda tanto como N2 como para las atenciones de requisitos cuando es un nuevo producto o cuando se trata de mejora continua, esto a partir de dos métricas vinculantes analizadas en la sección 3.8.1:
  - Velocidad de Entrega: Implementaciones completadas por sprint frente al objetivo planteado por la gerencia (20 entregas por sprint)
  - % de fallos post-implementación: Reabiertos y retrabajo asociado

  Ambas en coherencia con ITIL v4 y con la cadencia ágil definida presentamos las siguientes fórmulas para obtener la capacidad recomendada:

  - Capacidad efectiva del equipo (h/sprint): Estima las horas realmente disponibles para ejecutar trabajo de valor por sprint descontando ceremonias y despliegue continuo.

    Formula:

Figura 24
Formula Capacidad Efectiva del Equipo de Fabrica de Software

$$\text{Cap}_{eq} = N_p \times (H_b - H_c - H_{dep})$$

Donde:

Np: Número de recursos en el equipo (FTE)

Hb: Horas brutas por recurso en el sprint.

Hc: Horas por recurso en ceremonias / coord.

Hdep: Horas por recurso en deploy continuo.

Parámetros por sprint:

Np = 6

Hb = 40 horas/sem x 2 sem = 80h/sem

Hc = 2.5 horas

Hdep = 2 horas

Cálculo:

Capacidad útil / persona: 80 – 2.5 – 2 = 75.5 horas

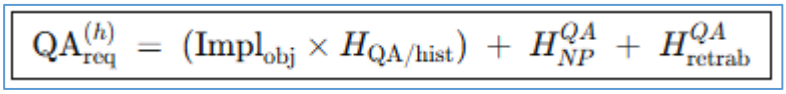
Capacidad total = 6 x 75.5 = 453 horas / sprint

Lectura: Desglose por rol:

- Desarrollador (4) = 4 recursos x 75.5 horas = 302 horas
- QA (1) = 1 QA x 75.5 horas = 75.5 horas
- Líder Técnico (1) = 1 LT x 75.5 horas = 75.5 horas

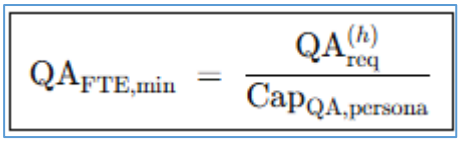
- Demanda para cumplir el objetivo (20 historias / sprint): Esto nos permite identificar la capacidad del rol QA necesario para atender el objetivo solicitado (20 requisitos por sprint).
  Formula:

Figura 25
Formula Horas de QA Requeridas por Sprint



$$\mathrm{QA}_{\mathrm{req}}^{(h)} = (\mathrm{Impl}_{\mathrm{obj}} \times H_{\mathrm{QA/hist}}) + H_{NP}^{QA} + H_{\mathrm{retrab}}^{QA}$$

Figura 26
Formula Recursos QA Mínimos para Atención del Sprint



$$\mathrm{QA}_{\mathrm{FTE,min}} = \frac{\mathrm{QA}_{\mathrm{req}}^{(h)}}{\mathrm{Cap}_{\mathrm{QA,persona}}}$$

Parámetros:

QA h/req = Horas de QA requeridas por sprint

IMPLobj: Implementaciones objetivo por sprint

Hqa/hist: Horas por historia QA

Hqa/np: Horas no planificadas QA

Hqa/retrab: Horas repruebas QA

QA fte.min = Recursos QA minímos para atender el sprint

Cap qa.persona = Capacidad útil de un QA en el sprint

Calculo:

IMPLobj: 20 implementaciones

Hqa/hist: 6 horas

Hqa/np: 35 horas

Hqa/retrab: 4 horas

Cap qa.persona = 75.5 h

QA h/req = (20 x 6) + 35 + 4 = 159 horas

QA fte.min = 159 / 75.5 = 2.11

Lectura:

- Para atender 20 historias de usuario por sprint se requiere 159 horas de QA.
- Por lo tanto, se requiere de 2 QA que aportan 75.5 horas x 2 =151 horas de QA, quedando un déficit de 8 horas por sprint que se cubriría con el apoyo del Líder Técnico o un desarrollador a modo de pruebas por pares o con automatización.
- Con 3 QA se tendría demasiada holgura.

En conclusión: La capacidad recomendada, en función a la carga actual, para el servicio de mesa de ayuda y fábrica de software se requiere adicionalmente:

- Mesa de Ayuda: 3 agentes.
- Fábrica de Software: 1 analista QA

- Matriz de Impacto x Urgencia -> Prioridad (N1)

  La matriz Impacto–Urgencia es el mecanismo que permite la priorización operativa en la mesa de ayuda N1. El impacto estima el alcance del efecto en el negocio/servicio mientras que la urgencia indica cuán rápidamente debe actuar la mesa de ayuda para evitar consecuencias mayores.

  El cruce de ambas dimensiones determina la prioridad (P1 – P4) que activa las metas de SLA y el tratamiento que le corresponde (incidente mayor, alto, medio, bajo). El resultado mantiene la trazabilidad incluso cuando el caso escala a N2.

Tabla 39
Matriz Impacto por Urgencia -> Prioridad (N1)

	Urgencia Alta	**Urgencia Media**	**Urgencia Baja**
Impacto Alto	P1	P2	P3
Impacto Medio	P2	P3	P4
Impacto Bajo	P3	P4	P4

- Niveles de Servicio

  Con la dimensión “Prioridad” se definen los acuerdos de servicios para los tickets de atención registrados por los usuarios:

  - SLA01 (primera respuesta),
  - SLA02 (resolución)
  - SLA03 (atención N2)

Tabla 40
Niveles de Servicio

		Objetivo de tiempo		
Prioridad	**Calendario**	**SLA-01 Primera Respuesta**	**SLA-02 Resolución**	**SLA-03 N2**
P1	L - V 9am - 5pm	1	4	4
P2	L - V 9am - 5pm	2	8	8
P3	L - V 9am - 5pm	8	24	16
P4	L - V 9am - 5pm	16	48	Sin objetivo
P5	L - V 9am - 5pm	Sin objetivo	Sin objetivo	Sin objetivo

Disclaimer: Por decisión de la empresa:

- Los tickets de atención con prioridad P5 no cuentan con compromiso de acuerdo de servicio.
- Para el N2 no se incluye tickets de atención con prioridad P4, P5
- La prioridad de atención de un ticket de N2 depende de la capacidad de la fábrica de software con la aprobación del Gerente de Sistemas, en caso contrario se planifica en el sprint más próximo según modelo.

- Métricas e Indicadores Propuestos

Tabla 41
Mesa de Ayuda – Métricas e Indicadores Propuestos

Métrica	Definición / Utilidad	Formula	Fuente de Datos	Frecuencia	Trazabilidad ITIL 4
Cumplimiento de SLA por Prioridad (P1–P4)	% de tickets resueltos dentro del SLA. Con esta métrica se espera tomar decisiones de capacidad o negociación de prioridad con el usuario. Los valores se obtienen por prioridad (P1 - P4) y oferta de servicio.	#tickets resueltos ≤ SLA / #tickets cerrados × 100% (segmentado por P1–P4)	Jira Azure DevOps	Mensual	**SVC:** Engage; Deliver & Support **Prácticas:** Service Level Mgmt, Incident Mgmt
MTTA (Tiempo medio de reconocimiento) por prioridad	Buscamos reducir el tiempo de respuesta y mejorar la experiencia del usuario midiendo incidentes N1 desde su creación hasta la primera respuesta del agente N1, segmentado por prioridad y oferta de servicio.	∑(*t*primer_contacto – *t*creacion) / N	Jira	Mensual	**SVC:** Engage; Deliver & Support **Prácticas:** Incident Mgmt, Monitoring & Event Mgmt
MTTR (Tiempo medio de resolución) por prioridad	Buscamos controlar y mejorar el tiempo de atención hasta la solución, midiendo sobre incidentes N1 y N2 cerrados y segmentados por prioridad y oferta de servicio.	∑(*t*resuelto – *t*creacion) N	Jira Azure DevOps	Mensual.	**SVC:** Deliver & Support **Prácticas:** Incident Mgmt, Problem Mgmt
FCR (First Contact Resolution)	Buscamos aumentar el porcentaje de resoluciones en el primer contacto con el usuario para reducir el escalamiento a N2 y el MTTR, considerando tickets N1 atendidos en primer contado, segmentando por prioridad y oferta de servicio.	#resueltos por N1 sin escalar / #elegibles en N1 × 100%	Jira	Mensual.	**SVC:** Deliver & Support **Prácticas:** Incident Mgmt, Knowledge Mgmt

Métrica	Definición / Utilidad	Formula	Fuente de Datos	Frecuencia	Trazabilidad ITIL 4
% de Escalados a N2	Buscamos evidenciar el nivel de conocimiento de N1 para orientar capacitaciones o generar mayor KB, midiendo tickets N1 que escalan a N2, segmentando por prioridad y oferta de servicio.	#tickets escalados a N2 / #tickets totales × 100%	Jira	Mensual.	**SVC:** Deliver & Support **Prácticas:** Incident Mgmt, Knowledge Mgmt
CSAT (Satisfacción del usuario)	Buscamos evidenciar y priorizar mejoras a partir de la percepción del usuario con encuestas post-cierre.	Escala Likert: 1 – 5 ((Muy Satisfecho + Satisfecho) / Total) * 100	Encuestas Jira	Mensual Anual	**SVC:** Engage **Prácticas:** Service Level Mgmt, Measurement & Reporting
Reaperturas N1	Buscamos identificar e incrementar oportunidades de mejora midiendo tickets reabiertos (posterior al cierre), segmentados por prioridad y oferta de servicio.	#tickets reabiertos / # tickets cerrados × 100%	Jira	Mensual	**SVC:** Deliver & Support; Improve **Prácticas:** Incident Mgmt, Problem Mgmt
Uso de Base de Conocimiento (Reutilización)	Buscamos identificar e incrementar las resoluciones rápidas y consistentes con el uso efectivo de KB, para reducir los escalamientos a N2, segmentado por prioridad y oferta de servicio.	% de tickets resueltos usando artículo KB	Jira Azure DevOps	Mensual.	**SVC:** Obtain/Build; Deliver & Support **Prácticas:** Knowledge Mgmt, Incident Mgmt
Tiempo de transferencia N1⇒N2 (Lead de escalamiento)	Buscamos minimizar los tiempos muertos de escalamiento de N1 a N2, con el fin de mejorar la fluidez del traspaso, segmentados por prioridad y oferta de servicio.	AVG(tcreacion_ AzureDevOps – tescala_Jira)	Jira Azure DevOps	Mensual	**SVC:** Deliver & Support **Prácticas:** Incident Mgmt

4.2.2 Modelo de gestión de requisitos de software (Mejora Continua y Bugs)

El modelo de mejora continua para la fábrica de software inicia con el registro del requisito (solicitud de cambio) o bug en Jira por el Product Owner como una solicitud de cambio, el agente N1 deriva al Project Manager para su registro en Azure DevOps con lo cual se logra la trazabilidad con JIRA.

Cada ticket genera una historia de usuario o un bug en Azure DevOps con criterios de aceptación, evidencias y enlaces recíprocos.

El backlog se gobierna con priorización por valor, impacto y dependencia y se ejecuta en sprint fijos de 2 semanas.

La segunda semana se reserva para el Release Planning del siguiente sprint, de modo que cada sprint arranca con el alcance y capacidad preacordados.

El equipo de fábrica de software ejecuta el Sprint (2 semanas) con Definition of Ready (requisitos claros y estimados) y Definition of Done (funcional, evidencias QA y documentación); el trabajo se descompone en historias de usuario / tareas en Azure DevOps, se controla el WIP para sostener el Cycle Time y se coordina diariamente en los Daily Scrum hacia un Sprint Goal orientado a lo esperado por el negocio.

Con respecto a la calidad, QA diseña y ejecuta pruebas validando criterios de aceptación en ambiente de UAT, conforme se van culminando los desarrollos dentro del sprint.

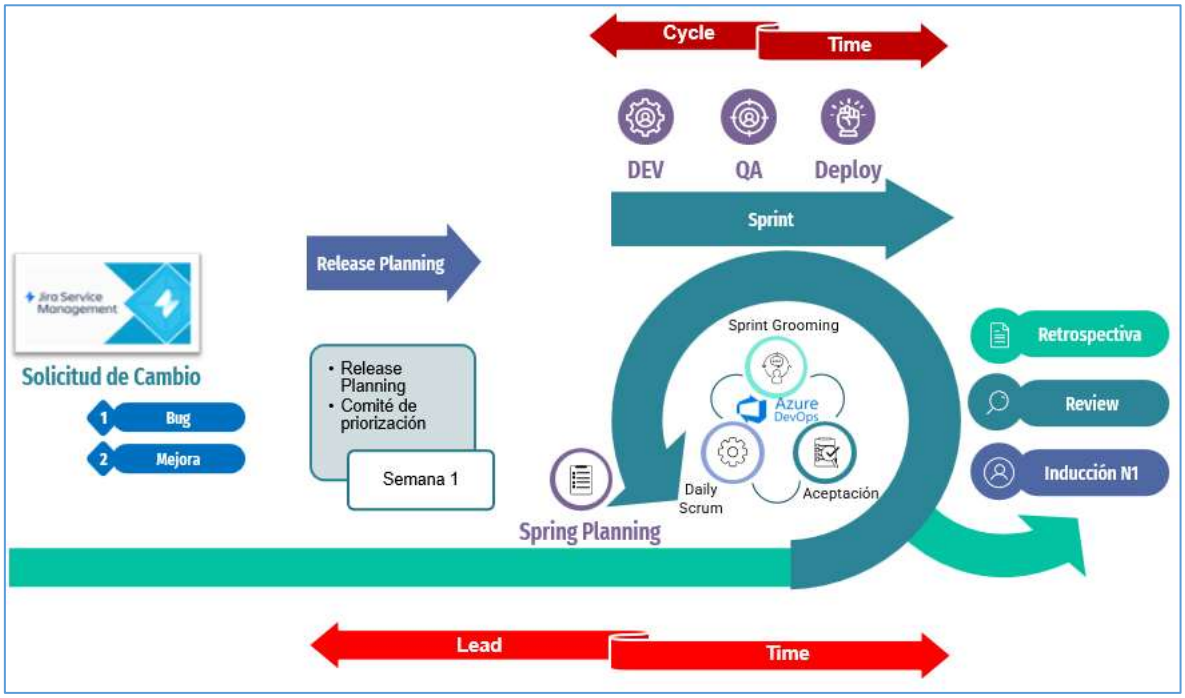
La gobernanza del despliegue contempla validación con el Product Owner y el usuario dueño del requisito para validar y aprobar el release antes de ejecutar el deploy, el cual es realizado en ventanas de tiempo nocturnas.

Tras el pase a producción se actualiza la base de conocimiento en Azure DevOps, se comunica la resolución al usuario y se ejecuta inducción a la mesa de ayuda N1 para habilitar el soporte efectivo desde el primer día con lo cual finaliza el sprint.

El desempeño de la fábrica se monitoriza mediante métricas técnicas y de satisfacción del usuario logrando que el resultado alimente la mejora continua del modelo y la toma de decisiones sobre capacidad, deuda técnica y estándares de calidad, cerrando el ciclo de co-creación de valor con el negocio.

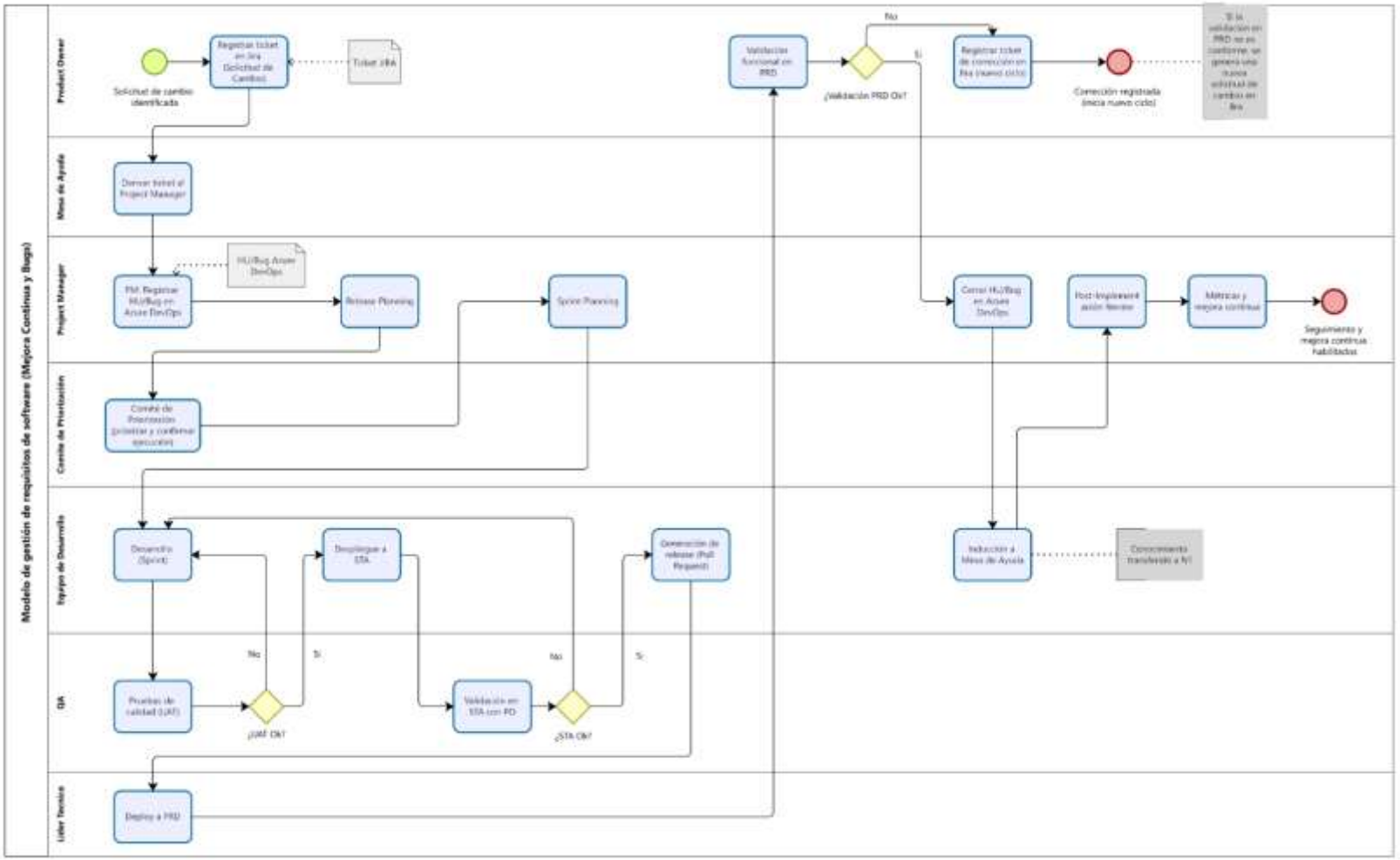
Figura 27

Modelo Adaptativo Desarrollo de Software Ágil (Mejora Continua)



- TO-BE del modelo de gestión de requisitos de software (Mejora Continua y Bugs)

Figura 28
Modelo de gestión de requisitos de software (Mejora Continua y Bugs)



- Descripción del modelo de gestión de requisitos de software (Mejora Continua y Bugs)

Nombre del Proceso	
Atención de Requisitos de Software (Mejora Continua y Bugs)	
Objetivo:	Gestionar requisitos de mejoras de software y bugs con un flujo controlado desde Jira hacia Azure DevOps, priorizado por gobernanza y ejecutado en ciclo ágil, asegurando calidad, liberación controlada y cierre formal. Incorpora manejo de no conformidad en PRD como nuevo ciclo y seguimiento post-implementación.
Alcance:	Incluye registro de solicitud de cambio/bug, derivación y creación de HU/Bug en Azure DevOps, planificación (Release Planning, Comité, Sprint Planning), ejecución (DEV+QA) y liberación (STA/PRD). Contempla cierre en Azure DevOps, inducción a N1, Post-Implementation Review y métricas/mejora continua.
Recursos Involucrados:	Personal de Mesa de Ayuda (5 agentes)
	Fábrica de Software
	JIRA / Azure DevOps
	Estaciones de usuario
	Ambientes DEV / UAT / STA / PRD
	Infraestructura de red y conectividad

Tabla 42
Descripción del modelo de gestión de requisitos de software (Mejora Continua y Bugs)

ID	Entrada	Actividad	Salida	Descripción	Responsable
1	Solicitud de cambio	Registrar ticket en Jira (Solicitud de Cambio)	Ticket registrado en Jira	El PO registra el requisito en Jira como solicitud de cambio.	Product Owner
2	Ticket registrado	Derivar ticket al Project Manager	Ticket derivado para escalamiento	El agente N1 deriva al Project Manager el ticket en JIRA para su registro en Azure DevOps manteniendo el mismo ID de JIRA para trazabilidad	Agente N1
3	Ticket derivado	Registrar en Azure DevOps	HU/Bug en Azure DevOps	Se crea una Historia de Usuario o Bug en Azure DevOps con criterios de aceptación, evidencias y su enlace a JIRA por trazabilidad.	Project Manager
4	Backlog disponible	Release Planning	Alcance/capacidad preacordados	Consolidar historias y bugs del próximo sprint; alinear capacidad, dependencias y riesgos; preparar incorporación al backlog del siguiente sprint (semana 1).	Project Manager

ID	Entrada	Actividad	Salida	Descripción	Responsable
5	Propuesta de alcance	Comité de priorización	Backlog priorizado	Ceremonia de gobierno (Negocio) donde PO, PM, Stakeholder y Jefe de TI priorizan requisitos y bugs según valor, impacto y dependencias; confirman qué pasa a ejecución en el siguiente sprint.	Comité de Priorización
6	Backlog priorizado	Sprint Planning	Sprint Goal + Sprint Backlog	Definir Sprint Goal; seleccionar HU y bugs y descomponer en tareas, verificando DoR y límites de WIP (inicio del sprint).	Project Manager
7	Sprint en ejecución	Desarrollo	Incremento implementado	Ejecutar el sprint (2 semanas) con DoR/DoD, control de WIP y coordinación diaria hacia el Sprint Goal (Daily Scrum).	Equipo de Desarrollo
8	Incremento implementado	Pruebas de calidad - Ambiente UAT	HU / Bug Validación UAT	QA diseña y ejecuta pruebas validando criterios de aceptación en UAT, conforme se culminan desarrollos dentro del sprint.	QA
9	HU / Bug Validado UAT	Despliegue Ambiente STA	HU / Bug Desplegado STA	El equipo de desarrollo despliega en STA.	Equipo de Desarrollo
10	HU / Bug Desplegado STA	Validación PO - Ambiente STA	HU/Bug aprobado	EL QA valida con el Product Owner en ambiente STA que la HU o Bug cumpla con el DoD.	QA / Product Owner
11	HU/Bug aprobado	Generación de release	Release aprobado	El equipo de desarrollo genera el Pull Request	Equipo de Desarrollo
12	Release aprobado	Deploy	Cambio desplegado en PRD	Ejecutar el deploy en producción	Lider Técnico
13	Solución en PRD	Validar con PO en PRD	Solución en PRD Validado	Validación funcional con el PO en producción	Project Manager / Product Owner
14	Solución en PRD Validado	Cerrar HU/Bug en Azure DevOps	HU/Bug cerrado en Azure DevOps	Cierre de HU/Bug en Azure DevOps.	Project Manager
15	Cambio en producción	Inducción a Mesa de Ayuda	Conocimiento transferido a N1	Transferencia de conocimiento a Mesa de Ayuda (N1) sobre la HU/Bug atendida por fábrica de software, con participación del Agente	Equipo de Desarrollo / Líder Técnico

ID	Entrada	Actividad	Salida	Descripción	Responsable
				N1 como receptor operativo	
16	PRD no conforme	Registrar ticket de corrección en Jira (nuevo ciclo)	Corrección registrada (nuevo ciclo)	Si la validación en PRD no es conforme, se registra una nueva solicitud de cambio en Jira para iniciar un nuevo ciclo de corrección.	Product Owner
17	Cierre/producción	Post-Implementación Review	Lecciones aprendidas / acciones	Revisión post-implementación para evaluar resultados, impactos y acciones derivadas.	Project Manager
18	Operación / datos	Métricas y mejora continua	Métricas y acciones de mejora	Monitoreo de métricas end-to-end y gestión de acciones de mejora continua del flujo.	Project Manager

- Ceremonias

Tabla 43
Ceremonias Desarrollo de Software Ágil (Mejora Continua)

Ceremonia	Objetivo	Frecuencia	Ámbito
Realease Planning	Consolidar requisitos y bugs del próximo sprint; alinear capacidad, dependencias y riesgos; preparar la incorporación al backlog del siguiente sprint.	Semana 1 del sprint	Fábrica de Software
Comité de priorización	Priorizar requisitos y bugs según valor, impacto y dependencia; confirmar qué pasará a ejecución en el siguiente sprint.	Semana 1 del sprint	Negocio
Sprint Planning	Definir el Sprint Goal; seleccionar requisitos/bugs y descomponer en historias de usuario, bugs y tareas, verificando DoR y límites de WIP.	Inicio del sprint	Fábrica de Software
Daily Scrum	Inspeccionar el progreso hacia el Sprint Goal, coordinar el plan de 24h y remover impedimentos.	Durante el sprint	Fábrica de Software
Sprint Grooming	Depurar y clarificar trabajo entrante; completar criterios de aceptación, evidencias y dependencias.	Durante el sprint	Fábrica de Software
Aprobación de Cambio	Verificar que el requisito y los de N2 cumplen criterios de aceptación, validar con el PO	Durante el sprint	Fábrica de Software
Deploy	Ejecutar el despliegue sin interrumpir la operación; coordinar comunicaciones.	Durante el sprint	Fábrica de Software
Sprint Retrospectiva	Identificar mejoras del proceso y acordar acciones concretas.	Fin del sprint	Fábrica de Software

Ceremonia	Objetivo	Frecuencia	Ámbito
Sprint Review	Presentar incremento y recoger feedback de negocio/usuarios.	Fin del sprint	Fábrica de Software
Inducción Mesa de Ayuda	Transferir conocimiento de los cambios liberados y preparar a N1 para el soporte post implementación.	Fin del sprint	Fábrica de Software

- Roles y Responsabilidades:

Tabla 44
Mejora Continua - Roles y Responsabilidades

Roles Fábrica de Software	Descripción
Agente N1	Identifica solicitudes registradas como 'Solicitud de cambio' en Jira y deriva al Project Manager para su registro en Azure DevOps, asegurando trazabilidad.
Product Owner	Maximiza valor, define criterios de aceptación, es interlocutor con el negocio y registra/gestiona solicitudes de cambio (requisitos) en Jira para su atención por Fábrica
Stakeholder	Avala el caso de negocio, alinea con la estrategia y aprueba el alcance
Comité de Priorización	Optimiza y prioriza el backlog de requisitos definiendo cuales se van a ejecutar en los sprints y su puesta en marcha. Conformado por el stakeholder, Product Owner, Project Manager y el Jefe de TI.
Project Manager	Dirige la fábrica de software y asume también el rol de Scrum Master, prioriza la línea base del sprint y lo presenta en el comité de priorización, remueve impedimentos, promueve métricas y mejora del proceso.
Equipo de Desarrollo	Implementa los desarrollos de software para nuevos productos y mejoras a los ya existentes, realizan actividades de diseño, construcción, pruebas unitarias e integración; actualiza documentación técnica y participa en la revisión del producto.
QA	Define estrategia de pruebas, diseña casos (funcionales y no-funcionales), ejecuta pruebas, aporta evidencia y reporta resultados.
Líder Técnico	Asegura consistencia de arquitectura y cumplimiento de no-funcionales (seguridad, capacidad, disponibilidad); guía decisiones técnicas en el equipo de desarrollo.
Jefe de TI	Dirige el área de TI y su roadmap, además es responsable del desarrollo, mantenimiento y evolución de las plataformas digitales de la empresa.

- Matriz RACI

  La matriz RACI resume el ciclo operativo de la fábrica de software cuando se requiere una mejora o se ha identificado un bug en las aplicaciones:

  (R) Responsable, (A) Aprobador, (C) Consultado, (I) Informado.

Tabla 45

Mejora Continua – Matriz RACI

Actividad / Rol	Stakeholder	Comité Priorización	PO	Agente N1	Project Manager	Equipo Dev	QA	Líder Técnico	Jefe TI
1. Registrar ticket en Jira (Solicitud de cambio)	I	I	A/R	I	I	I	I	I	I
2. Derivar ticket al PM (mantener ID Jira / trazabilidad)	I	I	I	A/R	C	I	I	I	I
3. Registrar en Azure DevOps (HU/Bug + tasks + enlace Jira)	I	I	I	I	A/R	C	C	C	I
4. Release Planning	C	I	C	I	A/R	C	C	C	I
5. Comité de Priorización	C	A/R	C	I	C	I	I	I	C
6. Sprint Planning	I	I	C	I	A/R	R	C	C	I
7. Desarrollo	I	I	I	I	C	A/R	C	C	I
8. Pruebas calidad (UAT)	I	I	C	I	I	C	A/R	C	I
9. Despliegue a STA	I	I	I	I	C	R	C	A	I
10. Validación PO en STA	I	I	A/R	I	I	I	R	I	I
11. Generación de release (Pull Request)	I	I	I	I	C	R	I	A	I
12. Deploy a PRD	I	I	I	I	C	C	C	A/R	I
13. Validar con PO en PRD	I	I	A/R	I	R	I	I	I	I
14. Cerrar HU/Bug en Azure DevOps	I	I	I	I	A/R	I	I	I	I
15. Inducción a Mesa de Ayuda (Agente N1)	I	I	I	C	C	R	C	A/R	I
16. Registrar ticket de corrección en Jira (nuevo ciclo)	I	I	A/R	I	I	I	I	I	I
17. Post-Implementation Review	C	I	C	I	A/R	C	C	C	C
18. Métricas y mejora continua	I	I	C	I	A/R	C	C	C	I

- Métricas e Indicadores Propuestos

Tabla 46
Mejora Continua – Métricas e Indicadores Propuestos

Métrica	Descripción / Utilidad	Formula	Fuente de Datos	Frecuencia
Implementaciones por sprint	Buscamos medir la productividad de entrega para planificar capacidad y compromisos de entrega; consideramo s historias y bugs terminados y desplegados, segmentados por sprint.	"Historias/bugs completados y desplegados por sprint."	Jira Azure DevOps	Sprint
Lead Time por historia	Buscamos reducir el tiempo total desde que se crea la historia / bug hasta que llega a producción. Se considera historias/bugs, segmentados por área	"Lead: creación hasta el despliegue"	Jira Azure DevOps	Sprint Mensual
Cycle Time por historia	Buscamos medir el tiempo desde que se inició el trabajo hasta el despliegue. Apuntamos a detectar cuellos y optimizar el flujo interno. Se considera historias/bugs en ejecución, segmentados por área.	"Cycle: inicio trabajo hasta el despliegue."	Azure DevOps	Sprint Mensual
% fallos post-implementación / reabiertos	Buscamos controlar el porcentaje de cambios producto de un despliegue en PRD que genere incidentes post-implantación. Apuntamos a controlar la calidad de cambios y reducir regresiones, segmentado por despliegue y sprint.	"(# items reabiertos / # items liberados) × 100%."	Azure DevOps	Sprint Mensual
% no planificado sobre capacidad del sprint	Buscamos controlar las asignaciones no previstas para proteger el Sprint Goal dado que consum la capacidad ya asignada del equipo; se mide las horas no planificadas por sprint.	"(horas no planif sprint / horas útiles sprint) × 100%."	Azure DevOps	Sprint
Cumplimiento del Sprint Goal	Buscamos medir el grado de logro del objetivo del sprint (HU), se mide el cumplimiento por sprint.	"(# objetivos logrados_HU / # objetivos plan_HU) × 100%."	Azure DevOps	Sprint
Retrabajo (h) por fallo post-implementación	Buscamos cuantificar el esfuerzo invertido en corregir defectos posteriores al despliegue. Apuntamos a costo de defectos para justificar la prevención del defecto o si es factible la automatización. Se mide en horas de fix + QA por sprint/despliegue.	"Fallos: SUMHoras de corrección + SUMHora de QA por corrección."	Azure DevOps	Sprint Mensual

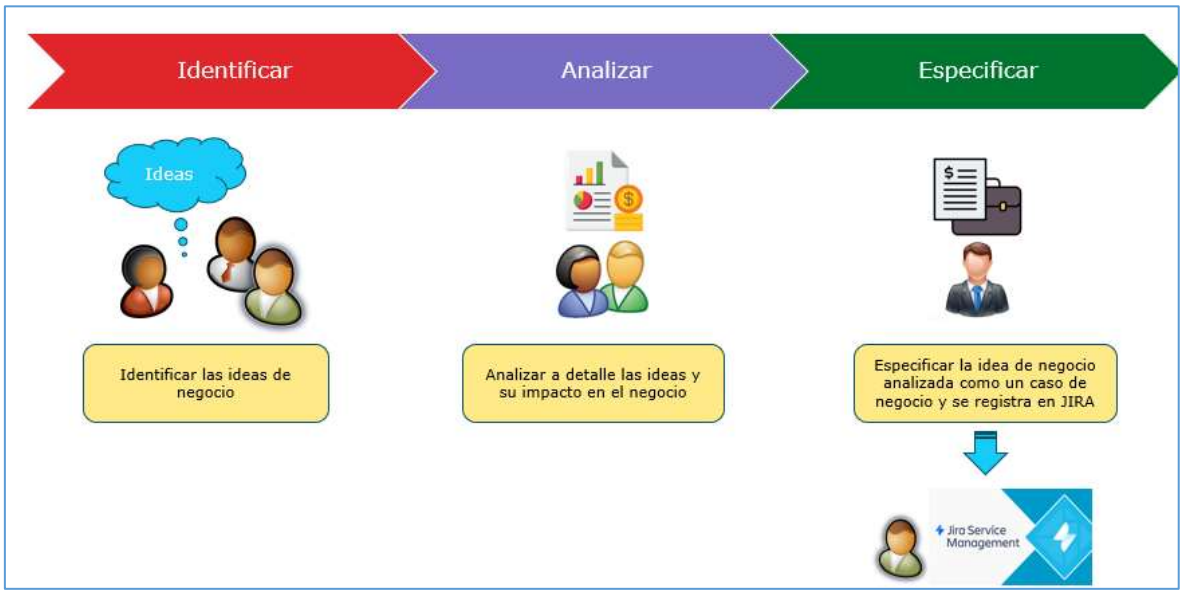
4.2.3 Modelo de gestión de requisitos de software (Nuevos Productos)

El modelo de gestión de requisitos para un nuevo producto de software inicia con el flujo de gestión de requisitos y culmina en la entrega iterativa por sprint, manteniendo la trazabilidad Jira <=> Azure DevOps y el marco metodológico descrito en la Sección 2.

La periodicidad de este modelo está relacionado principalmente a campañas organizadas por Marketing, cambios en el negocio con impacto en sus objetivos estratégicos, cuando se visiona un nuevo producto software para la empresa, migración a nuevas tecnologías o alineamiento a la arquitectura All-in-One de la sede principal.

En la etapa previa del modelo, que corresponde a la gestión de requisitos, el negocio identifica ideas (oportunidades / hipótesis de valor), las analiza en detalle (viabilidad, impacto, urgencia, económico-operativo, riesgos y dependencias) y las especifica como casos de negocio con objetivos, criterios de éxito y supuestos; cada requisito resultante se registra en Jira como una solicitud de cambio, el agente N1 deriva al Project Manager para que lo registre en Azure DevOps con lo cual se logra la trazabilidad con JIRA.

Figura 29
Flujo de Gestión de Requisitos (Nuevo Producto)



A partir de esta derivación y registro, el modelo permite el avance por tres niveles de planificación encadenados:

1. Portfolio Planning (semana 1) para seleccionar épicas y capacidad en función de estrategia, riesgo y roadmap del portafolio.
2. Product Planning (semana 2) para refinar features, definir el MVP, criterios de aceptación, no-funcionales en caso aplique y dependencias tecnológicas.
3. Release Planning (semana 3) para consolidar alcance, ventanas y criterios de go/no-go y en que sprint iniciar.

Una vez definida esta línea base, el equipo de fábrica de software ejecuta el Sprint (2 semanas) con Definition of Ready (requisitos claros y estimados) y Definition of Done (funcional, evidencias QA y documentación); el trabajo se descompone en historias de usuario / tareas en Azure DevOps, se controla el WIP para sostener el Cycle Time y se coordina diariamente en los Daily Scrum hacia un Sprint Goal orientado a lo esperado por el negocio.

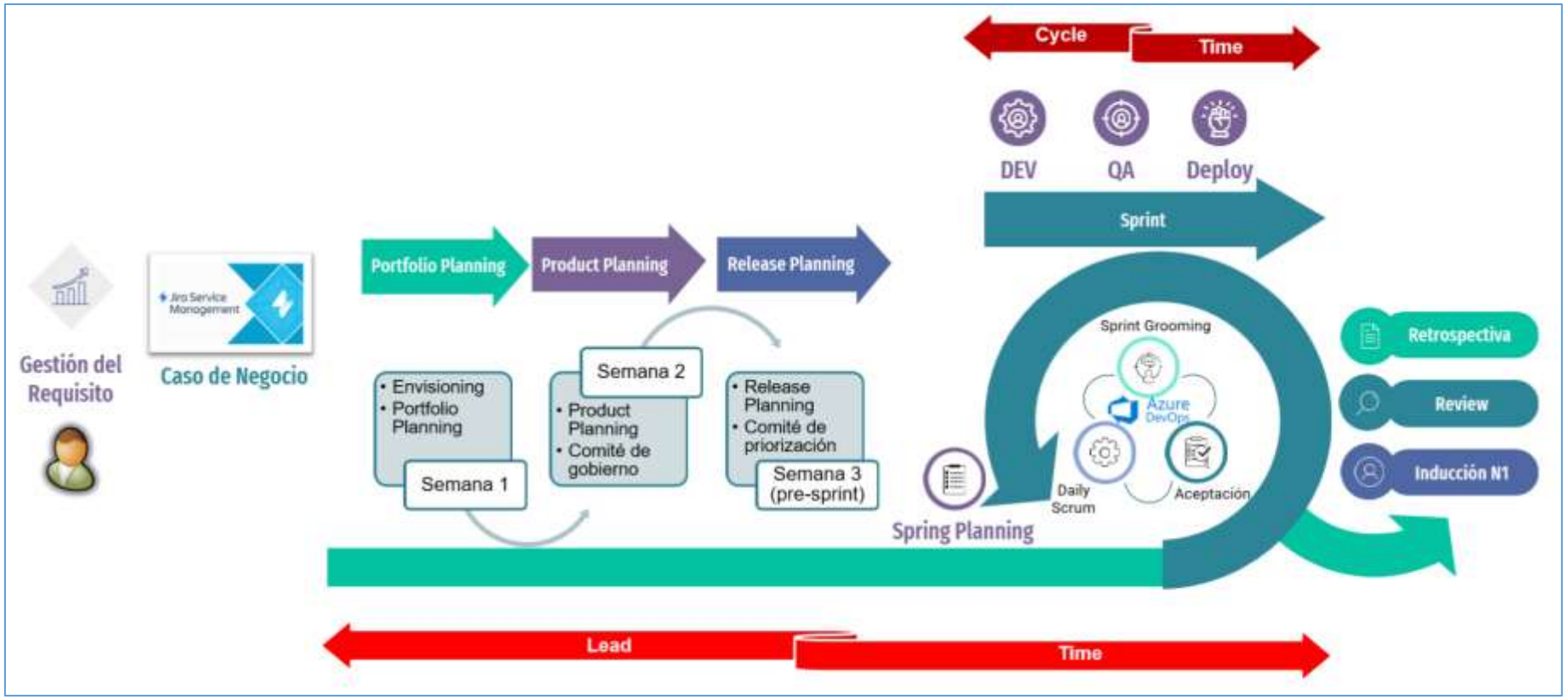
Con respecto a la calidad, QA diseña y ejecuta pruebas validando criterios de aceptación en ambiente de UAT, conforme se van culminando los desarrollos dentro del sprint.

La gobernanza del despliegue contempla validación con el Product Owner y el usuario dueño del requisito para validar y aprobar el release antes de ejecutar el deploy, el cual es realizado en ventanas de tiempo nocturnas.

Tras el pase a producción se actualiza la base de conocimiento en Azure DevOps, se comunica la resolución al usuario y se ejecuta inducción a la mesa de ayuda N1 para habilitar el soporte efectivo desde el primer día con lo cual finaliza el sprint.

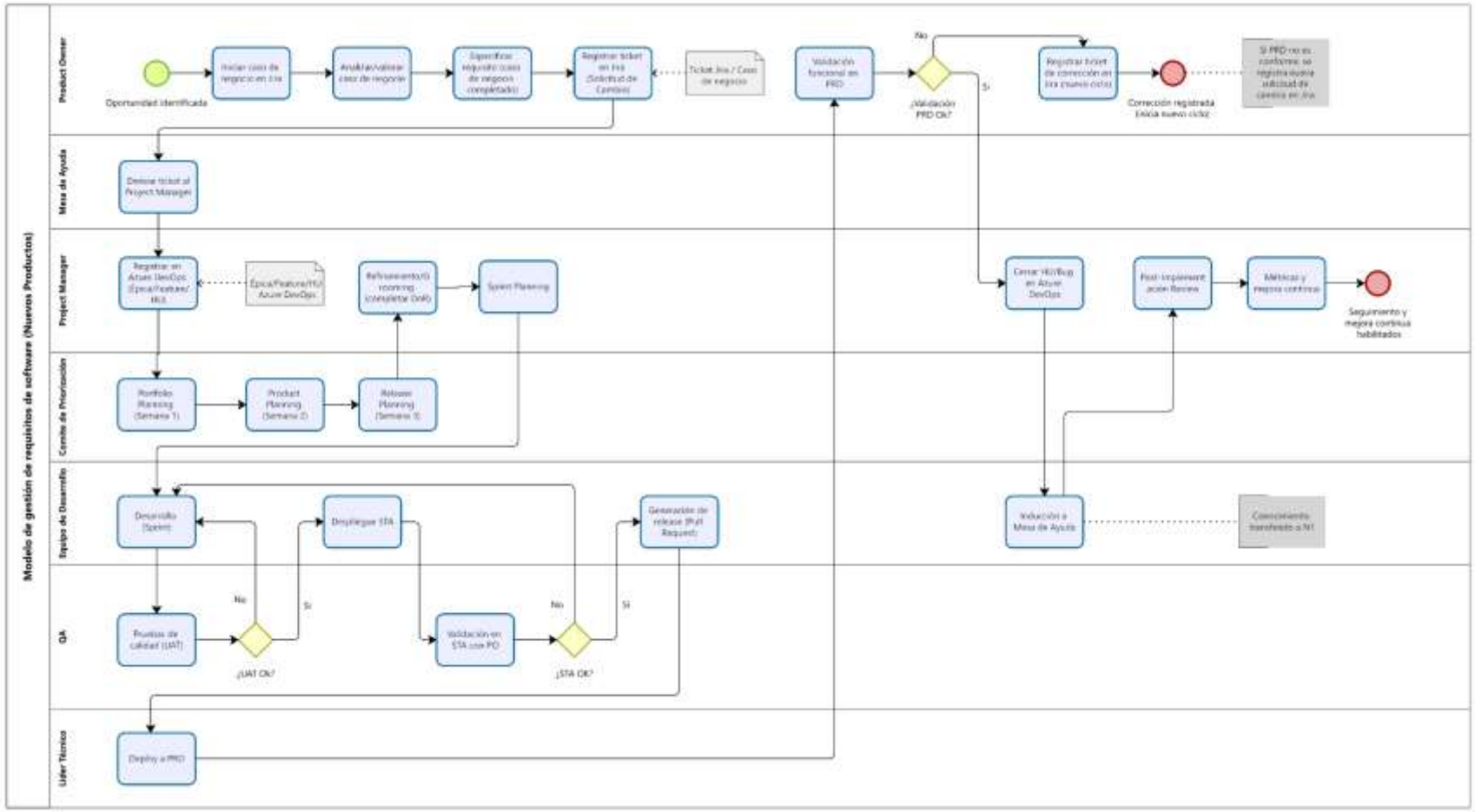
El desempeño de la fábrica se monitoriza mediante métricas técnicas y de satisfacción del usuario logrando que el resultado alimente la mejora continua del modelo y la toma de decisiones sobre capacidad, deuda técnica y estándares de calidad, cerrando el ciclo de co-creación de valor con el negocio.

Figura 30
Modelo Adaptativo Desarrollo de Software Ágil (Nuevo Producto)



- TO-BE del modelo de gestión de requisitos de software (Nuevos Productos)

Figura 31
Modelo de gestión de requisitos de software (Nuevos Productos)



- Descripción del modelo de gestión de requisitos de software (Nuevos Productos)

Nombre del Proceso	
Atención de Requisitos de Software (Nuevos Productos)	
Objetivo:	Asegurar la creación y entrega de nuevos productos desde la generación de la idea y caso de negocio hasta su construcción, liberación y validación en producción, bajo un esquema de gobernanza por etapas (portfolio/product/release). Incluye control de calidad, cierre y seguimiento post-implementación.
Alcance:	Cubre desde el caso de negocio y registro en Jira, estructuración en Azure DevOps (épica/feature/HU) y planificación por semanas (Portfolio/Product/Release Planning), hasta ejecución ágil (sprint, UAT/STA/PRD), cierre e inducción. Considera Post-Implementation Review y métricas/mejora continua.
Recursos Involucrados:	Personal de Mesa de Ayuda (5 agentes)
	Fábrica de Software
	JIRA / Azure DevOps
	Estaciones de usuario
	Ambientes DEV / UAT / STA / PRD
	Infraestructura de red y conectividad

Tabla 47
Descripción del modelo de gestión de requisitos de software (Nuevos Productos)

ID	Entrada	Actividad	Salida	Descripción	Responsable
1	Idea / oportunidad	Identificar oportunidad e iniciar caso de negocio en Jira	Caso de negocio	El negocio propone la idea y elabora un caso de negocio como punto de partida.	Product Owner
2	Caso de negocio	Analizar/valorar (beneficios, riesgos, dependencias)	Valoración documentada	Se analiza viabilidad, impacto, urgencia, riesgos y dependencias para sustentar la iniciativa.	Product Owner
3	Valoración lista	Especificar requisito (objetivo, criterios de éxito, evidencia)	Caso de negocio completado	Se especifica el caso de negocio con objetivos, criterios de éxito y supuestos.	Product Owner
4	Caso de negocio completado	Registrar ticket en Jira (Solicitud de Cambio)	Ticket registrado en Jira	El PO registra el requisito en JIRA como solicitud de cambio y adjunta el documento con el detalle del caso de negocio.	Product Owner
5	Ticket registrado	Derivar ticket al Project Manager	Ticket derivado para atención	El agente N1 deriva al Project Manager el ticket en JIRA para su registro en Azure DevOps manteniendo el mismo ID de JIRA para trazabilidad	Agente N1
6	Ticket derivado	Registrar en Azure DevOps	Epica/Feature /HU en Azure DevOps	En base al ticket y al caso de negocio, se crea una Epica con Feature e	Project Manager

ID	Entrada	Actividad	Salida	Descripción	Responsable
				Historias de Usuario en Azure DevOps con criterios de aceptación, evidencias y su enlace a JIRA para trazabilidad.	
7	Epicas en Azure DevOps	Portfolio Planning (Semana 1)	Épicas capacidades seleccionadas	Se seleccionan épicas y se mide la capacidad de atención en base a la estrategia, riesgo y roadmap del portafolio.	Comité de Priorización
8	Selección Portfolio	Product Planning (Semana 2)	Epicas, Features /MVP refinados	Se refinan epicas con sus features, se define su MVP, criterios de aceptación y dependencias funcionales y no funcionales en caso aplique.	Comité de Priorización
9	Product Planning listo	Release Planning (Semana 3)	Línea base de release + go/no-go	Se consolida alcance y criterios de go/no-go y se define en qué sprint iniciar.	Comité de Priorización
10	Línea base definida	Refinamiento/Groomi ng (completar DoR)	Historias listas (DoR)	Se define las tareas, se completa DoR/evidencias/dependen cias para dejar historias listas.	Project Manager / Equipo de Desarrollo
11	Historias listas	Sprint Planning (objetivo y selección de trabajo)	Sprint Goal + Sprint Backlog	Se define el Sprint Goal y se seleccionan historias, descomponiendo en tareas y verificando DoR y límites de WIP.	Project Manager / Equipo de Desarrollo
12	Sprint en ejecución	Desarrollo	Incremento implementad o	Ejecutar el sprint (2 semanas) con DoR/DoD, control de WIP y coordinación diaria hacia el Sprint Goal (Daily Scrum).	Equipo de Desarrollo
13	Incremento implementado	Pruebas de calidad - Ambiente UAT	HU Validacion UAT	QA diseña y ejecuta pruebas validando DoD en UAT, conforme se culminan desarrollos dentro del sprint.	QA
14	HU Validado UAT	Despliegue Ambiente STA	HU Desplegado STA	El equipo de desarrollo despliega en STA.	Equipo de Desarrollo
15	HU Desplegado STA	Validación PO - Ambiente STA	HU aprobado	EL QA valida con el Product Owner en ambiente STA que la HU cumpla con el DoD.	QA / Product Owner
16	HU aprobado	Generación de release	Release aprobado	El equipo de desarrollo genera el Pull Request	Equipo de Desarrollo
17	Release aprobado	Deploy	Cambio desplegado en PRD	Ejecutar el deploy en producción	Lider Técnico
18	Solución en PRD	Validar con PO en PRD	Solución en PRD Validado	Validación funcional con el PO en producción	Project Manager / Product Owner

ID	Entrada	Actividad	Salida	Descripción	Responsable
19	Solución en PRD Validado	Cerrar HU en Azure DevOps	HU cerrado en Azure DevOps	Cierre de HU en Azure DevOps.	Project Manager
20	Cambio en producción	Inducción a Mesa de Ayuda	Conocimiento transferido a Mesa de Ayuda (N1)	Transferencia de conocimiento a Mesa de Ayuda (N1) sobre la HU atendida por fábrica de software, con participación del Agente N1 como receptor operativo	Equipo de Desarrollo / Líder Técnico
21	Operación / datos	Métricas end-to-end y mejora continua	Métricas y acciones de mejora	Se monitorean métricas del modelo y se gestionan acciones de mejora continua del flujo de atención.	Project Manager
22	PRD no conforme	Registrar ticket de corrección en Jira (nuevo ciclo)	Corrección registrada (nuevo ciclo)	Si la validación en PRD no es conforme, se registra una nueva solicitud de cambio en Jira para iniciar un nuevo ciclo de corrección.	Product Owner
23	Cierre / Producción	Post-Implementación Review	Lecciones aprendidas / acciones derivadas	Revisión posterior al despliegue para evaluar resultados e impactos y acordar acciones derivadas.	Project Manager

- Ceremonias

Tabla 48
Ceremonias Desarrollo de Software Ágil (Nuevo Producto)

Ceremonia	Objetivo	Frecuencia	Ámbito
Portfolio Planning	Seleccionar épicas y capacidad por ciclo en función de estrategia, riesgo y roadmap del portafolio.	Semana 1	Negocio
Product Planning	Refinar features, identificar MVP, criterios de aceptación funcionales; alinear dependencias y restricciones.	Semana 2	Negocio
Release Planning	Consolidar requisitos, plan de comunicaciones/go-no go para el siguiente sprint.	Semana 3	Negocio
Comité de Priorización	Ordenar requisitos de nuevo producto (épicas/features) según valor, impacto, riesgo y dependencias; decidir qué entra a planeación.	Semana 3	Negocio
Sprint Planning	Definir Sprint Goal; seleccionar historias y descomponer en tareas verificando DoR y límites de WIP.	Inicio del sprint	Fábrica de Software
Sprint Grooming	Depurar y clarificar trabajo entrante; completar criterios/evidencias/dependencias para dejar historias listas (DoR).	Durante el sprint	Fábrica de Software
Daily Scrum	Inspeccionar progreso al Sprint Goal y adaptar el plan de 24 h; remover impedimentos.	Durante el sprint	Fábrica de Software
Pruebas de Calidad (UAT)	Validar criterios de aceptación en UAT, generar evidencias y reportar resultados.	Durante el sprint	Fábrica de Software
Aprobación de Cambio	Validar funcionalmente antes del despliegue productivo, conforme a criterios definidos.	Durante el sprint	Fábrica de Software
Deploy	Ejecutar el despliegue sin interrumpir la operación; coordinar comunicaciones.	Durante el sprint	Fábrica de Software
Inducción Mesa de Ayuda	Transferir conocimiento de los cambios liberados y preparar a N1 para el soporte post release.	Fin del sprint	Fábrica de Software
Sprint Review	Presentar incremento y recoger feedback.	Fin del sprint	Fábrica de Software
Sprint Retrospectiva	Identificar mejoras del proceso y acordar acciones concretas.	Fin del sprint	Fábrica de Software
Post-Implementación Review	Revisar resultados e impactos post despliegue y acordar acciones derivadas.	Posterior al despliegue	Negocio / TI

- Roles y Responsabilidades:

Tabla 49
Modelo Nuevo Producto – Roles

Rol	Descripción
Product Owner (PO)	Consolida las actividades de negocio del modelo: inicia y completa el caso de negocio, registra la solicitud de cambio en Jira y realiza la validación funcional (STA/PRD) según el flujo definido. Miembro formal del Comité de Priorización.
Agente N1 (Mesa de Ayuda)	Deriva el ticket registrado en Jira al Project Manager para su registro/gestión en Azure DevOps y participa como receptor operativo en la inducción post-release.
Project Manager (PM)	Gestiona la ejecución del trabajo en la fábrica: registra/estructura el trabajo en Azure DevOps (épica/feature/HU), coordina la planificación/seguimiento del ciclo ágil y gestiona el cierre operativo del trabajo en ADO según el modelo. Miembro formal del Comité de Priorización.
Comité de Priorización	Órgano único de gobierno del modelo: prioriza y toma decisiones de avance (Portfolio/Product/Release Planning) según estrategia, valor, impacto, dependencias y capacidad. Integrado formalmente por Stakeholder, Product Owner, Project Manager y Jefe de TI.
Equipo de Desarrollo	Implementa el desarrollo: diseño, construcción, pruebas unitarias/integración, generación de release (Pull Request) y soporte técnico requerido durante el ciclo.
QA	Diseña y ejecuta pruebas, genera evidencias y reporta resultados; valida en UAT según el flujo definido.
Líder Técnico	Asegura consistencia técnica/arquitectura, guía decisiones técnicas y lidera/asegura el despliegue a producción (PRD) según el modelo; participa en la inducción.
Stakeholder	Asegura alineación con la estrategia del negocio y participa en decisiones de priorización/avance del modelo. Miembro formal del Comité de Priorización.
Jefe de TI	Alinea decisiones con el roadmap TI, capacidad y riesgos; participa en el gobierno del modelo. Miembro formal del Comité de Priorización.

- Matriz RACI

  La matriz RACI resume el ciclo operativo de la fábrica de software cuando se requiere implementar un nuevo producto de software.

  (R) Responsable, (A) Aprobador, (C) Consultado, (I) Informado.

Tabla 50
Modelo Nuevo Producto – Matriz RACI

Actividad / Rol	Comité Priorización	PO	N1	PM	Equipo Dev	QA	Líder Técnico	Stakeholder	Jefe TI
1. Identificar oportunidad e iniciar caso de negocio en Jira	I	A/R	I	I	I	I	I	I	I
2. Analizar/valorar (beneficios, riesgos, dependencias)	C	A/R	I	C	I	I	I	C	C

Actividad / Rol	Comité Priorización	PO	N1	PM	Equipo Dev	QA	Líder Técnico	Stakeholder	Jefe TI
3. Especificar requisito (objetivo, criterios de éxito, evidencia)	C	A/R	I	C	I	I	I	C	C
4. Registrar ticket en Jira (Solicitud de Cambio)	I	A/R	I	I	I	I	I	I	I
5. Derivar ticket al Project Manager	I	I	A/R	C	I	I	I	I	I
6. Registrar en Azure DevOps (Épica/Feature/HU)	I	C	I	A/R	C	C	C	I	I
8. Portfolio Planning (Semana 1)	A/R	C	I	C	I	I	I	C	C
9. Product Planning (Semana 2)	A/R	C	I	C	I	I	C	C	C
10. Release Planning (Semana 3)	A/R	C	I	C	I	I	C	C	C
11. Refinamiento/Grooming (completar DoR)	I	C	I	A	R	C	C	I	I
12. Sprint Planning (objetivo y selección de trabajo)	I	C	I	A	R	C	C	I	I
13. Desarrollo	I	I	I	C	A/R	C	C	I	I
14. Pruebas de calidad (UAT)	I	C	I	I	C	A/R	C	I	I
15. Despliegue a STA	I	I	I	C	R	C	A	I	I
16. Validación PO (STA)	I	A/R	I	C	I	R	I	I	I
17. Generación de release (Pull Request)	I	I	I	C	R	I	A	I	I
18. Deploy (PRD)	I	I	I	C	C	C	A/R	I	I
19. Validar con PO en PRD	I	A/R	I	C	I	I	I	I	I
20. Cerrar HU en Azure DevOps	I	I	I	A/R	I	I	I	I	I
21. Inducción a Mesa de Ayuda	I	I	C	C	R	C	A/R	I	I
22. Métricas end-to-end y mejora continua	I	C	I	A/R	C	C	C	I	I
23. Registrar ticket de corrección en Jira (nuevo ciclo)	I	A/R	I	I	I	I	I	I	I
24. Post-Implementación Review	I	C	I	A/R	C	C	C	C	C

- Métricas e Indicadores Propuestos

Tabla 51
Modelo Nuevo Producto – Métricas e Indicadores Propuestos

Métrica	Descripción / Utilidad	Formula	Fuente de Datos	Frecuencia
Time-to-Market (TTM)	Buscamos medir el tiempo desde la aprobación de una idea hasta el primer despliegue en producción	"Aprobación de la idea ⇒ primer despliegue."	Jira Azure DevOps	Mensual
Predictibilidad (Plan vs. Implementado)	Buscamos medir el porcentaje de requisitos planificados se implementan en el sprint. Apuntamos a mejorar la confianza del negocio con el servicio de fábrica de software.	"Entregas implementadas / planificadas × 100%."	Azure DevOps	Sprint
Calidad de despliegue	Buscamos controlar la calidad de cambios para reducir regresiones derivados de despliegues (postimplementación), se mide despliegues por sprint.	"# despliegues con incidente / # despliegues × 100% (defectos críticos en 30 días)."	Jira Azure DevOps	Sprint
Burndown/ Burn-up del sprint	Buscamos controla la evolución diaria de trabajo pendiente vs completado durante el sprint.	"Gráfica: tendencia de alcance vs. progreso."	Azure DevOps	Diario en sprint

4.3 Gestión del Cambio

Como parte de la solución de optimizar la gestión de incidencias y requisitos de software en la empresa mediante la adopción de buenas prácticas internacionales: ITIL en la mesa de ayuda y desarrollo ágil en la fábrica de software tercerizada, implica no solo un rediseño de procesos operativos, sino también una transformación organizacional en términos de estructuras, roles, formas de trabajo y herramientas tecnológicas preparando a la organización, sus colaboradores y sus procesos para asumir, adoptar y sostener el cambio de manera progresiva, estructurada y sostenible.

4.3.1 Ámbito de Aplicación del Cambio

- Empresarial:

  El cambio propuesto es de carácter organizacional, ya que implica transformar la Gerencia de Sistemas en la empresa. De ser una unidad por momentos reactiva que brinda soporte técnico y desarrollo de software, a pasar a convertirse en un facilitador estratégico del negocio, cuyo valor estará vinculado a indicadores de desempeño como tiempos de entrega, calidad de servicio, innovación y satisfacción del usuario interno.

  Esta transformación afecta la forma en que la Gerencia de Sistemas interactúa con las áreas de negocio, promueve la estandarización de procesos, establece métricas de control y contribuye con el cumplimiento de los objetivos estratégicos de la empresa, tales como acelerar la oferta de beneficios a clientes / beneficiarios y continuar con la implementación de la estrategia digital.

- Grupal:

  Las áreas directamente impactadas por el cambio son:

  - Mesa de Ayuda (Jefatura de Infraestructura y Ciberseguridad): Adoptará prácticas ITIL como gestión de incidentes, solicitudes de servicio, niveles de prioridad, escalamiento y mejoras en los acuerdos de nivel de servicio (SLA).
  - Fábrica de Software (Jefatura IT y Plataformas de Pago): Pasará a trabajar bajo principios desarrollo ágil integrándose con ITIL.
  - Áreas de Negocio (Finanzas y Productos, Ventas, Customer Care, Operaciones): Cambiarán su relación con TI, de emitir requisitos informales

basado en una plantilla en Excel a colaborar con validaciones, retroalimentación y priorización funcional / negocio.

- Individual:

  A nivel de personas, el cambio afecta a distintos perfiles:

  - Analista de Aplicaciones y Mesa de ayuda: Técnicos de soporte: deberán usar herramientas como Jira, seguir procedimientos ITIL y priorizar con base en SLA.
  - Equipo de Fábrica (Desarrolladores y Analista QA): deberán adoptar metodologías ágiles, herramientas CI/CD y colaborar más estrechamente con operaciones.
  - Usuarios clave (Product Owner): Registraran requisitos correctamente y validaran entregables.
  - Jefaturas de la Gerencia Sistemas: Asumirán un rol de facilitadores y propiciarán buenas prácticas.

4.3.2 Tipo de Cambio Implicado

Tabla 52
Tipos de Cambio

Tipo de Cambio	¿Aplica?	Descripción
Estructural	Si	Considerar nuevos roles (Gestor de Incidencias, Owner de Producto, QA Automatizador, Líder Técnico), se mejorará el comité de priorización de requisitos. También se define una nueva estructura de toma de decisiones basada en SLAs y escalamiento.
Cultural	Si	Se busca cambiar el mindset de los colaboradores hacia una cultura de colaboración, transparencia, mejora continua y responsabilidad compartida. Esto implica abandonar el trabajo en silos, fomentar dailies, post mortem de incidentes, y visibilizar logros técnicos dentro de la organización.
Tecnológico	Si	Ahondar en el consumo de Jira Service Management (gestión de tickets), Azure DevOps (automatización), Selenium (pruebas de software) y tableros de métricas en Excel o Power BI.
Producto	No	El cambio no altera los productos comerciales de la empresa sino los procesos internos de mesa de ayuda y desarrollo de software que dan soporte a esos productos.
Procesos	Si	Se mejorarán los flujos de atención de incidencias y requisitos de software.

4.3.3 Factores Críticos de Éxito

- Liderazgo visible y comprometido: La participación del CEO y CIO refuerza la importancia de la implementación y les da legitimidad frente a los colaboradores.
- Apoyo institucional de RR.HH.: Necesario para diseñar planes de capacitación, gestión del cambio individual, reconocimiento y seguimiento del clima organizacional.
- Capacitación técnica y funcional: El equipo de mesa de ayuda y de fábrica de software deben de obtener un conocimiento regular de ITIL y desarrollos ágiles.
- Implementación en fases: Un despliegue escalonado permite validar los procesos, corregir errores y generar quick wins.
- Comunicación constante: Comunicar los avances ayudan a mantener el compromiso y disminuir incertidumbre.
- Análisis de resultados: KPIs que permitirán evaluar el impacto real del cambio en tiempo, calidad, errores y satisfacción.

4.3.4 Riesgos del Cambio

Tabla 53
Riesgos del Cambio

Riesgo Potencial	Nivel de Riesgo	Medidas de Mitigación Propuestas
Resistencia al cambio por parte de la mesa de ayuda y/o fábrica de software	Alto	Incluir al personal desde la fase de diagnóstico, mostrar beneficios directos, permitir participación en decisiones y hacerlos coparticipes de los avances.
Sobrecarga operativa en fases iniciales	Medio	Implementar por etapas, redistribuir funciones temporalmente entre los integrantes.
Baja adopción de herramientas nuevas	Medio	Elaborar guías paso a paso, obtener soporte de la sede central de la empresa y mantener feedback constante.
Desalineación entre prioridades de TI y negocio	Alto	Establecer comité de "pre" priorización solo de IT, validar requisitos antes de desarrollarlos y mantener comunicación frecuente.
Falta de métricas para evaluar impacto	Medio	Diseñar e implementar KPIs desde la fase de diagnóstico para medir antes, durante y después del cambio.

5. Evaluación Costo Beneficio

5.1 Estructura de Costos

La estructura de costos del proyecto está enfocada netamente al provisionamiento de recursos humanos especializados para la planificación, preparación, elaboración e implementación de un modelo de gestión para la mesa de ayuda y la fábrica de software.

Los recursos humanos planificados y costeados no serán dedicados al 100%, sino en función a la demanda típica que se tiene por etapa. Es por ello, que la estructura de costos está disgregada por las fases por la que debe pasar el proyecto para implementar los modelos:

- **Fase de Consultoría**: Ajuste, actualización del entendimiento, onboarding del equipo consultor y del jefe de Proyecto sobre las últimas actualizaciones al modelo actual, refinamiento y determinación de los modelos AS IS y TO BE, así como la determinación de las brechas identificadas en las áreas de Mesa de Ayuda y Fábrica de Software.
- **Fase de Implementación**: Preparación de recursos logísticos, documentos a utilizar tanto estratégicos, normativos como operativos, configuración de herramientas tecnológicas a utilizar en el nuevo modelo de servicio, así como la capacitación a los recursos que laboran en la mesa de ayuda y la fábrica de software. Planificación de la estrategia de difusión y comunicación a la organización.
- **Fase Post implementación**: Planificación y ejecución de la marcha blanca y estrategia de co-living entre el AS IS y el TO BE de la mesa de ayuda y Fábrica de Software. Soporte y actualización de la configuración en las herramientas preconfiguradas. Evaluación y gestión del cambio. Reforzamiento de la estrategia de difusión y comunicación a las áreas involucradas. Reforzamiento de los interesados del proyecto en lo que respecta a participación.
- **Fase Monitoreo y Control**: Revisión, toma de datos actuales, preparación de ajustes al modelo en función a los resultados mensuales obtenidos luego de la implementación. Corrección de incidentes relacionados a la aplicación del nuevo modelo. Elaborar métricas preventivas sobre el rendimiento mensual en la atención de tickets de la mesa de ayuda. Revisión de las buenas prácticas y lecciones aprendidas y retrospectivas sobre el proyecto.

Tabla 54
Estructura de Costos

Fase del Proyecto	Cantidad recursos	% dedicación	Cantidad Horas	Costo x Hora	Subtotal
Fase - Consultoría (1 mes)					
Jefe de Proyecto	1	100%	160	S/ 120	S/ 19,200
Consultor desarrollo métodos ágiles	1	50%	80	S/ 80	S/ 6,400
Consultor ITIL -Mesa de Ayuda	1	50%	80	S/ 80	S/ 6,400
Subtotal					S/ 32,000
Fase implementación (2 meses)					
Jefe de Proyecto	1	50%	160	S/ 120	S/ 19,200
Consultor desarrollo métodos ágiles	1	50%	160	S/ 80	S/ 12,800
Consultor ITIL -Mesa de Ayuda	1	50%	160	S/ 80	S/ 12,800
Subtotal					S/ 44,800
Fase Post implementación (1 mes)					
Jefe de Proyecto	1	50%	80	S/ 120	S/ 9,600
Consultor desarrollo métodos ágiles	1	50%	80	S/ 80	S/ 6,400
Consultor ITIL -Mesa de Ayuda	1	50%	80	S/ 80	S/ 6,400
Subtotal					S/ 22,400
Monitoreo y Control (3 meses)					
Jefe de Proyecto	1	20%	96	S/ 120	S/ 11,520
Consultor desarrollo métodos ágiles	1	20%	96	S/ 80	S/ 7,680
Consultor ITIL -Mesa de Ayuda		20%	96	S/ 80	S/ 7,680
Subtotal					S/ 26,880
Total costo de recursos					S/ 126,080
Costos de Gestión (15%)					S/ 18,912
Costos por contingencia (10%)					S/ 12,608
Total Gastos por Gestión y Contingencia					S/ 157,600
Utilidad del proyecto (30%)					S/ 37,824
Costos total del proyecto					**S/ 195,424**

Consideraciones a la estructura de costos

- Las licencias de software necesarias para la realización del proyecto como JIRA, Azure DevOps, herramientas ofimáticas y de productividad para la implementación del modelo planteado serán asumidas por la empresa. Los costos aproximados, según estudios de mercado vigentes, referentes a dicho conceptos son los siguientes:

Tabla 55
Costos Licencias

Concepto	Periodo	Monto
Licencias JIRA Service Management (10 usuarios)	7 meses	S/ 2,800
Licencia Azure DevOps (3 usuarios)	7 meses	S/ 2,000
Herramientas de documentación (Lucidchart, Draw.io Premium, Notion Pro)	7 meses	S/ 1,500
Hosting, dominio, servidores cloud (Azure/AWS)	7 meses	S/ 2,100
Subtotal Herramientas		**S/ 8,400**

- Los costos relacionados a capacitación adicional en ITIL V4 o Scrum serán asumidas por la empresa. El rol de capacitación y formación que el proyecto desempeñará estará enfocado en el modelo propuesto y no en qué consiste ITIL y Scrum.

Tabla 56
Costos Capacitación

Concepto	**Monto (S/.)**
Capacitación interna en ITIL v4 (10 pers.)	S/ 5,500
Talleres SCRUM / Agile Coaching	S/ 5,500
Soporte técnico externo (outsourcing puntual)	S/ 3,000
Auditoría técnica externa final	S/ 10,500
Subtotal Servicios	**S/ 24,500**

- Los gastos administrativos, materiales impresos, logísticos y de operación que involucre el equipo de proyecto serán asumidos por la empresa.
- Los gastos relaciones a la difusión, socialización, campaña y afianzamiento de campaña para comunicar a la empresa sobre el cambio a realizar en la mesa de ayuda y la fábrica de software serán asumidos por la Empresa.
- Los costos promedio por hora de los jefes de proyecto, Consultor en ITIL, Consultor en Scrum están basados en el sueldo promedio en el mercado peruano, según estudio de las principales empresas de gestión de talento y de bolsa de empleo como Michel Page, Price WaterHouse Cooper y Deloitte, asimismo como funciones del Ministerio de Trabajo y Promoción del Empleo.
- Los costos operativos del proyecto, luego de su implementación son los siguientes. El licenciamiento se prorratea del costo de las herramientas y el hosting por 7 meses,

dividiéndose el mismo por esa cantidad de meses. Asimismo, el costo de capacitación se prorratea a un costo de S/.1000 por mes de cara a implementar un programa de refuerzo anual de ITIL y SCUM y las bondades de su sinergia.

Tabla 57
Costos operativos del proyecto

Concepto	Cantidad	Subtotal
Licencia Jira, Azure DevOps Herramientas documentación Hosting	1	S/ 1,200
Refuerzos - Capacitación ITIL - SCRUM Capacitación anual, 12,000	1	S/ 1,000
Total Costo Operativo		**S/ 2,200**

5.2 Beneficios financieros

En base a las conclusiones de la sección “3.8.1 Métricas e Indicadores Detectados al Proceso Gestión de Incidencias (Mesa de Ayuda) y Gestión de Requisitos (Fábrica de Software)” se identificaron entre los puntos de dolor la capacidad operativa para la atención de ambos servicios la cual se pudo cuantificar durante la propuesta de solución plasmado en la sección “4.2 Desarrollo de la Solución” identificando la necesidad de integrar 3 agentes para la mesa de ayuda y 1 analista QA para la fábrica de software.

Aumento de la eficiencia operativa de la Mesa de Ayuda en un 33%

La mesa de ayuda, con el modelo de atención actual, ejecuta 400 tickets a un promedio de 1.5 horas por tickets (indicado en 3.7.1 Descripción del Proceso de Atención de Incidencias (Mesa de Ayuda)). Con el modelo propuesto, al bajar el tiempo de atención de 1.5 horas a 1 hora, la capacidad aumenta a 600 tickets por mes. De no cambiar el modelo, se tendría que invertir S/.5,625 para llegar a 600.

Por consecuencia, el modelo logra alcanzar eficiencia operativa de S/.5,625 con respecto al modelo anterior.

Tabla 58
Costo / Beneficio Mesa de ayuda

EFICIENCIA DE LA MESA DE AYUDA				
Modelo	**Tk atendidos x mes**	**Tiempo (horas) Atención promedio**	**Costo x Hora**	**Costo total atención tickets**
AS IS	400	1.5	S/ 18.75	S/ 11,250
Diferencia	200	1.5	S/ 18.75	S/ 5,625
	600	1.5	S/ 18.75	S/ 16,875
TO BE	600	1	S/ 18.75	S/ 11,250
Eficiencia				S/ 5,625

Aumento de la eficiencia operativa de la Fábrica de Software en un 29%

De la misma manera, la capacidad de la fábrica se ve incrementada por el nuevo modelo de atención en un 29% debido a se reduce en 4 horas el tiempo promedio de atención de historias de usuario / bugs.

Con el modelo actual se implementan 12 historias de usuario a un ritmo de 14 horas por cada uno (indicado en 3.7.2 Descripción del Proceso de Atención de Requisitos (Fábrica de Software)). El modelo nuevo reduce los tiempos de 14 a 10 horas, lo cual permite implementar 8 historias de usuario más que el modelo anterior. Al multiplicarlo por los costos por hora y el tiempo de demora se tiene una eficiencia de S/. 4,960.

Tabla 59
Costo / Beneficio– Fábrica de Software

EFICIENCIA DE LA FABRICA DE SOFTWARE				
Modelo	**HU implementadas x mes**	**Tiempo (horas) Atención promedio**	**Costo x Hora**	**Costo total atención tickets**
AS IS	12	14	S/ 62.00	S/ 10,416
Eficiencia	8	14	S/ 62.00	S/ 6,944
	20	14	S/ 62.00	S/ 17,360
TO BE	20	10	S/ 62.00	S/ 12,400
Eficiencia				S/ 4,960

En base a la eficiencia obtenida por ambos servicios se obtiene:

- Eficiencia Mensual = Eficiencia Mesa de Ayuda + Eficiencia Fábrica de Software
- Eficiencia Mensual = S/ 5,625 + S/ 4,960
- Eficiencia Mensual = S/ 10,585

5.3 Rentabilidad y factibilidad del proyecto

El presupuesto total del proyecto asciende a S/ 195,424, distribuidos en los primeros siete meses de ejecución, cubriendo los costos de recursos humanos, gestión, contingencia y utilidad. A partir del mes 8, se han considerado costos operativos recurrentes por licenciamiento de herramientas tecnológicas, estimados en S/ 2,200 mensuales.

En cuanto a los beneficios, se proyectan ahorros monetizados mensuales desde el mes 5, derivados de la mejora en la eficiencia operativa, reducción de tiempos de resolución de tickets, mayor integración entre áreas técnicas, disminución de retrabajos, y mejora en la satisfacción de los usuarios internos. Estos beneficios han sido valorados en S/ 10,585 de Eficiencia Mensual, con una proyección a 48 meses.

Utilizando una tasa de descuento del 0.8% mensual (equivalente al 9.6% anual), se obtuvieron los siguientes indicadores financieros:

- **Valor Actual Neto (VAN): S/ 117,279.56**
- **Tasa Interna de Retorno (TIR): 3.30% mensual** (aproximadamente 47.1% anual)
- **Mes de rentabilidad del proyecto: Mes 26**

Tabla 60
Retorno de Inversión

Mes	Ingresos (S/.)	Egresos (S/.)	Flujo Neto (S/.)	Flujo Acumulado (S/.)
Mes 1	S/ 0	S/ 45,343	-45343	-45343
Mes 2	S/ 0	S/ 31,740	-31740	-77083
Mes 3	S/ 0	S/ 31,740	-31740	-108823
Mes 4	S/ 0	S/ 31,740	-31740	-140563
Mes 5	S/ 10,585	S/ 12,739	-2154	-142717
Mes 6	S/ 10,585	S/ 20,059	-9474	-152191
Mes 7	S/ 10,585	S/ 22,064	-11479	-163670
Mes 8	S/ 10,585	S/ 2,200	8385	-155285

Mes	Ingresos (S/.)	Egresos (S/.)	Flujo Neto (S/.)	Flujo Acumulado (S/.)
Mes 9	S/ 10,585	S/ 2,200	8385	-146900
Mes 10	S/ 10,585	S/ 2,200	8385	-138515
Mes 11	S/ 10,585	S/ 2,200	8385	-130130
Mes 12	S/ 10,585	S/ 2,200	8385	-121745
Mes 13	S/ 10,585	S/ 2,200	8385	-113360
Mes 14	S/ 10,585	S/ 2,200	8385	-104975
Mes 15	S/ 10,585	S/ 2,200	8385	-96590
Mes 16	S/ 10,585	S/ 2,200	8385	-88205
Mes 17	S/ 10,585	S/ 2,200	8385	-79820
Mes 18	S/ 10,585	S/ 2,200	8385	-71435
Mes 19	S/ 10,585	S/ 2,200	8385	-63050
Mes 20	S/ 10,585	S/ 2,200	8385	-54665
Mes 21	S/ 10,585	S/ 2,200	8385	-46280
Mes 22	S/ 10,585	S/ 2,200	8385	-37895
Mes 23	S/ 10,585	S/ 2,200	8385	-29510
Mes 24	S/ 10,585	S/ 2,200	8385	-21125
Mes 25	S/ 10,585	S/ 2,200	8385	-12740
Mes 26	S/ 10,585	S/ 2,200	8385	-4355
Mes 27	**S/ 10,585**	**S/ 2,200**	**8385**	**4030**
Mes 28	S/ 10,585	S/ 2,200	8385	12415
Mes 29	S/ 10,585	S/ 2,200	8385	20800
Mes 30	S/ 10,585	S/ 2,200	8385	29185
Mes 31	S/ 10,585	S/ 2,200	8385	37570
Mes 32	S/ 10,585	S/ 2,200	8385	45955
Mes 33	S/ 10,585	S/ 2,200	8385	54340
Mes 34	S/ 10,585	S/ 2,200	8385	62725
Mes 35	S/ 10,585	S/ 2,200	8385	71110
Mes 36	S/ 10,585	S/ 2,200	8385	79495

Estos resultados confirman que el proyecto es **financieramente viable y rentable**. A pesar del incremento sostenido en los costos operativos a partir del mes 8 y del retraso en el inicio de los beneficios (desde el mes 5), el flujo de caja acumulado demuestra una recuperación progresiva de la inversión inicial, alcanzando el punto de equilibrio en el **mes 26**. A partir de este hito, el modelo empieza a generar beneficios netos constantes, lo que valida su sostenibilidad en el largo plazo.

Tabla 61
Flujo de Caja CAPEX / OPEX

Año	CAPEX (S/.)	OPEX Total (S/.)	Ingresos Totales (S/.)	Flujo Neto (S/.)
Año 0	S/ 195,424	–	–	**-195,424**
Año 1	–	S/ 104,343	S/ 63,510	**-40,833**
Año 2	–	S/ 26,400	S/ 127,020	**100,620**
Año 3	–	S/ 26,400	S/ 127,020	**100,620**

El proyecto requiere una inversión inicial de S/ 195,424 (CAPEX). Durante el primer año, el flujo operativo aún es negativo. Sin embargo, a partir del segundo año se observan flujos netos positivos sostenidos. Esto evidencia que el proyecto recupera su inversión en el transcurso del segundo año de operación, lo que confirma su viabilidad económica.

En conclusión, la implementación de los nuevos modelos basado en ITIL v4 y Scrum no solo responde a criterios técnicos y operativos sólidos, sino que también representa una **inversión estratégica y rentable** para la organización, aportando valor financiero tangible y alineándose con los objetivos de eficiencia, calidad y mejora continua en la gestión de servicios de TI.

Conclusiones

- Cumplimiento del objetivo central: Se diseñó y documentó un modelo adaptado a ITIL para la mesa de ayuda (N1–N2) y dos modelos de fábrica de software con enfoque ágil (mejora continua y nuevo producto) alineados con el marco teórico. Esta propuesta resuelve el problema planteado, optimiza la atención de la mesa de ayuda y la gestión de requisitos de software, al reducir tiempos de respuesta y resolución, disminuir retrabajos postimplementación y mejorar la satisfacción del usuario, con trazabilidad y medición reproducible de los resultados.

- Cumplimiento del objetivo 1: Se logró identificar los procesos vigentes de la empresa, se documentó el flujo end-to-end tal como opera actualmente (AS-IS, desde la creación del ticket hasta su cierre o pase a producción según corresponda) apoyado con la identificación de la cadena de valor de la organización y el mapa de procesos a nivel de macroprocesos. La caracterización de los dos procesos específicos con mayor impacto en los objetivos estratégicos permitió comprender con precisión su funcionamiento actual.

- Cumplimiento del objetivo 2: Se logró identificar y documentar los puntos de dolor en la mesa de ayuda y la fábrica de software mediante un enfoque trazable que combinó el modelo AS-IS, el análisis contra KPs y un assessment por roles que participan en ambos servicios, dejando evidencia por cada punto de dolor. Estos hallazgos establecieron una base clara para presentar una propuesta de mejora.

- Cumplimiento del objetivo 3: Se logró definir 3 modelos integrados; uno para la mesa de ayuda y dos para la fábrica de software (mejora continua y nuevos productos) con flujos, roles y ceremonias definidos, así como también métricas que permitan la trazabilidad entre Jira y Azure DevOps, desde el registro hasta el cierre / puesta en producción. Su adopción habilita un marco operable y gobernable alineando la ejecución a objetivos de negocio y cerrando el ciclo de co-creación de valor con el negocio.

Recomendaciones

- Ajuste de capacidad en Nivel 2 (Conclusión del Objetivo 2 y 3): Ajustar la capacidad del equipo de calidad en periodos de mayor demanda y priorizar la automatización de verificaciones en salidas relevantes, con el fin de disminuir retrabajos y evitar fallos posteriores a la puesta en producción.
- Mejora del proceso de admisión en Nivel 1 (Conclusión del Objetivo 1 y 2): Mejorar las plantillas provistas en Jira y aplicar sin restricción "Impacto × Urgencia => Prioridad" complementándolo con una revisión semanal de casos en riesgo de incumplir SLA para reducir escalamientos y elevar la resolución al primer contacto.
- Preservar la gobernabilidad de cambios y trazables (Conclusión del Objetivo Central y 3): Propiciar la aprobación formal de cambios con un registro sencillo, disponer de procedimientos de rollback cuando corresponda y revisar los resultados de cada despliegue midiendo la tasa de cambios exitosos y los defectos posteriores a la implementación.
- Cierre de conocimiento obligatorio (Conclusión del Objetivo 2): Después de cada despliegue en producción, realizar una breve inducción a N1 y crear o actualizar la base de conocimiento; auditar mensualmente su reutilización y su impacto en la resolución al primer contacto.
- Control del no planificado en Nivel 2 (Conclusión del Objetivo 3): Establecer un límite visible (por ejemplo, un 15–20% de la capacidad del sprint) para el trabajo no planificado y, si se supera, activar acciones correctivas enfocadas en mejorar la admisión y priorización.
- Métricas con revisión en cadencia (Conclusión del Objetivo Central y 3): Operar un tablero mensual de gobernabilidad con indicadores clave (Lead/Cycle Time, FCR, % reabiertos, tasa de cambios exitosos, CSAT, NPS) y revisar tendencias para acordar acciones en las retrospectivas de la mesa de ayuda y de la fábrica de software.
- Comunicación con el negocio (Conclusión del Objetivo 3): Institucionalizar la reunión mensual con el área usuaria para revisar SLA,s riesgos y trabajo no planificados; registrando acuerdos y con retroalimentación para el Comité de Priorización tanto para el servicio de mesa de ayuda como para el de fábrica de software.

Referencias

Abdelkebir, S., Maleh, Y., & Belaissaoui, M. (2017). An agile framework for ITS management in organizations: A case study based on DevOps. En *Proceedings of the 2nd International Conference on Computing and Wireless Communication Systems* (pp. 1–8). Association for Computing Machinery. https://doi.org/10.1145/3167486.3167556

AXELOS. (2019). *ITIL® 4: Foundation Edition*. The Stationery Office.

AXELOS. (2020). *ITIL® 4: Create, Deliver and Support*. The Stationery Office.

AXELOS. (2021a). *ITIL® 4: Drive Stakeholder Value*. The Stationery Office.

AXELOS. (2021b). *ITIL® 4: High Velocity IT*. The Stationery Office.

Beck, K., Beedle, M., van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., Grenning, J., Highsmith, J., Hunt, A., Jeffries, R., Kern, J., Marick, B., Martin, R. C., Mellor, S., Schwaber, K., Sutherland, J., & Thomas, D. (2001). *Manifesto for Agile Software Development*. Agile Manifesto. https://agilemanifesto.org/

Belalcázar, A. (2017). Incorporation of good practices in the development and deployment of applications through alignment of ITIL and DevOps. En *2017 International Conference on Information Systems and Computer Science (INCISCOS)* (pp. 224–230). IEEE. https://doi.org/10.1109/INCISCOS.2017.31

Bradley, T. (2022). *IT service management: A guide for ITIL® foundation exam candidates* (3a ed.). BCS Learning & Development Limited.

Decreto Supremo 016-2024-JUS, Reglamento de la Ley N.° 29733, Ley de Protección de Datos Personales. (2024, 30 de noviembre). *Diario Oficial El Peruano*. https://busquedas.elperuano.pe/dispositivo/SE/2349653-1

International Organization for Standardization, & International Electrotechnical Commission. (2008). *ISO/IEC 12207:2008 — Systems and software engineering — Software life cycle processes*. https://www.iso.org/standard/43447.html

International Organization for Standardization, & International Electrotechnical Commission. (2016). *ISO/IEC TR 29110-1:2016 — Systems and software*

engineering — Lifecycle profiles for Very Small Entities (VSEs) — Part 1: Overview. https://www.iso.org/standard/62711.html

International Organization for Standardization, & International Electrotechnical Commission. (2023). *ISO/IEC 25010:2023 — Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — Product quality model*. https://www.iso.org/standard/78176.html

International Organization for Standardization, International Electrotechnical Commission, & Institute of Electrical and Electronics Engineers. (2023). *ISO/IEC/IEEE 15288:2023 — Systems and software engineering — System life cycle processes*. https://www.iso.org/standard/81702.html

Kersten, M. (2018). What flows through a software value stream? *IEEE Software, 35*(4), 8–11. https://doi.org/10.1109/MS.2018.2801538

Kniberg, H., & Skarin, M. (2010). *Kanban and Scrum: Making the most of both*. C4Media.

Ley 28051. (2023), Ley de prestaciones alimentarias en beneficio de los trabajadores sujetos al régimen laboral de la actividad privada. Congreso de la República del Perú. https://www.leyes.congreso.gob.pe/Documentos/Leyes/28051.pdf

Ley 29733. (2011), Ley de Protección de Datos Personales. Congreso de la República del Perú. https://www.leyes.congreso.gob.pe/Documentos/Leyes/29733.pdf

Ley 30096 (2013), Ley de Delitos Informáticos y sus modificatorias. Congreso de la República del Perú. https://leyes.congreso.gob.pe/Documentos/ExpVirPal/Texto_Consolidado/30096-TXM.pdf

Mahnic, V. (2015). Improving software development throughput using Lean metrics. *Journal of Systems and Software, 105*, 145–154. https://doi.org/10.1016/j.jss.2015.04.073

ManageEngine ServiceDesk Plus. (s.f.-a). *Características del software de mesa de ayuda y servicio TI* [Infografía]. Recuperado el 8 de febrero de 2025, de https://www.manageengine.com/latam/service-desk/caracteristicas-mesa-de-ayuda.html

ManageEngine ServiceDesk Plus. (s.f.-b). *Evolución de ITIL: Una profundización en la historia de ITIL*. Recuperado el 8 de febrero de 2025, de https://www.manageengine.com/latam/service-desk/itsm/evolucion-de-itil.html

Marrone, M., & Hammerle, M. (2021). The value of ITIL practices for IT service management. *Information Systems Management, 38*(1), 24–33. https://doi.org/10.1080/10580530.2020.1857340

Pressman, R. S., & Maxim, B. R. (2020). *Ingeniería del software: un enfoque práctico* (9a ed.). McGraw-Hill Education.

Project Management Institute. (2021). *Guía de los fundamentos para la dirección de proyectos (Guía del PMBOK®)* (7a ed.).

Redondo, C. (2019, 29 de abril). *ITIL® 4: la conversión Agile. Paradigma Digital*. Recuperado el 8 de febrero de 2025, de https://www.paradigmadigital.com/techbiz/que-es-itil-v4/

Sachs, R. (2021). *Modern IT Governance with ITIL 4*. O'Reilly Media.

Schwaber, K., & Sutherland, J. (2020). *The Scrum Guide: The definitive guide to Scrum: The rules of the game*. Scrum Guides. https://scrumguides.org/

Scrum.org. (2020). *The Scrum Framework Poster* [Infografía]. https://www.scrum.org/resources/scrum-framework-poster

Smyth, D. (2020). *Implementing ITIL® 4 Foundation: Key concepts and practices*. IT Governance Publishing.

Torres, A., & Luna, M. (2020). Gestión de Servicios de TI: Implementación de ITIL v4 en organizaciones públicas. *Revista Tecnología y Sociedad, 18*(2), 45–60.