



**Desarrollo de un Sistema de Gestión Inteligente de Tareas Scrum
para el Departamento de Tecnologías de la Información y
Comunicación (TICS) de la Organización Pública: Servicio de
Rentas Internas (SRI)**

Santiago Ismael Ávila Cruz

Tutor

MSc. Roberto Bautista

Trabajo de grado para optar por el título de Tecnólogo Superior en
Desarrollo de Software

Instituto Superior Tecnológico Internacional ITI

Carrera de Desarrollo de Software

D.M Quito, 25 de febrero de 2026

Dedicatoria

En primer lugar, a Dios, fuente de sabiduría infinita, por iluminar mi camino académico y sostenerme en los momentos difíciles, gracias a Dios he conseguido seguir adelante superando cualquier adversidad con fuerza de voluntad y esfuerzo constante. A mis padres, por su apoyo incondicional y por ser la base de cada logro que alcanzo. Cuya confianza y aliento han sido el motor que me impulsó a superar cada desafío durante el desarrollo de este trabajo. A mis abuelos, quienes sin educación formal me transmitieron las lecciones de vida valiosas como el respeto, la integridad y perseverancia. A mis compañeros por ser un apoyo constante en especial por sus consejos y soporte en todas las etapas a lo largo de la carrera. Recurrentemente me inspiré en la resiliencia de Stephen Hawking para comprender que, sin importar las limitaciones o los obstáculos del camino, la curiosidad y el intelecto siempre encontrarán una forma de brillar. Este logro es prueba de que la voluntad no conoce fronteras.

Agradecimiento

En primer lugar, extendo mi más sincero agradecimiento al **Instituto Superior Tecnológico Internacional (ITI)**, por proporcionar el entorno académico y los recursos necesarios para llevar a cabo este proyecto. La formación recibida y el apoyo institucional han sido pilares esenciales en mi desarrollo profesional.

De manera especial, reconozco la guía constante del **MSc. Roberto Bautista**, cuyo conocimiento experto, paciencia y consejos estratégicos fueron determinantes para superar los obstáculos encontrados durante la investigación. Su mentoría no solo enriqueció este trabajo, sino que también fortaleció mis habilidades como investigador.

Agradezco a la institución por su apoyo y disposición para ayudarme en cada momento. Su trabajo y esfuerzo hacen que este lugar sea un entorno propicio para el aprendizaje y el crecimiento. Quisiera agradecer a nuestra rectora, **MSc. Rosa Paredes**, por su liderazgo y por ser una fuente de inspiración dentro de nuestra comunidad educativa. A pesar de los diversos cambios en la coordinación de la carrera, aprecio y reconozco a todas las personas que han participado en mi crecimiento profesional.

Finalmente, quiero expresar mi profunda gratitud a mis amigos, así como a mis compañeros de clase y trabajo, quienes han sido esenciales en este proceso. Aprecio sinceramente sus palabras motivadoras, su apoyo constante, y la orientación y enseñanza que me han brindado.

Autoría

Yo, Santiago Ismael Ávila Cruz autor del presente informe, me responsabilizo por los conceptos, opiniones y propuestas contenidos en el mismo.

Atentamente,

SANTIAGO ISMAEL ÁVILA CRUZ

Quito, 25 de febrero de 2026

MSc. Roberto Carlos Bautista Ramos

DIRECTOR DE TRABAJO DE TITULACIÓN

Certifica

Haber revisado el presente informe de investigación, que se ajusta a las normas institucionales y académicas establecidas por el Instituto Tecnológico Internacional Universitario ITI, por tanto, se autoriza su presentación final para los fines legales pertinentes.

Roberto Carlos Bautista Ramos

Quito, 25 de febrero 2026

Declaración de Cesión de Derechos de Trabajo fin de carrera

Para constancia de esta autorización, en la ciudad de Quito - Pichincha, a los 8 días del mes de octubre de 2025, firmo conforme: Conste por el presente documento la cesión de los derechos del trabajo de fin de carrera, de conformidad con las siguientes cláusulas:

PRIMERA: Yo, Santiago Ismael Ávila Cruz, bajo la dirección de MSc. Roberto Bautista declaró ser el autor del trabajo de fin de carrera con el tema “Desarrollo de un Sistema de Gestión Inteligente de Tareas Scrum para la Organización Pública Servicio de Rentas Internas (SRI)”, como requisito fundamental para optar por el título de Tecnólogo Superior en Desarrollo de Software, a su vez autorizo a la biblioteca del Tecnológico Superior Universitario Internacional ITI, para que pueda registrar en el repositorio digital y difunda esta investigación con fines netamente académicos, pues como política del Tecnológico Superior Universitario Internacional ITI, los trabajos de fin de carrera se aplican, materializan y difunden en beneficio de la comunidad.

SEGUNDA: Los comparecientes, MSc. Roberto Bautista, en calidad de director del trabajo fin de carrera y el Sr. Santiago Ismael Ávila Cruz, como autor del mismo, por medio del presente instrumento, tienen a bien ceder en forma gratuita sus derechos del trabajo fin de carrera y conceden la autorización para que el ITI pueda utilizar este trabajo en su beneficio y/o de la comunidad, sin reserva alguna. El Tecnológico Superior Universitario Internacional ITI no se hace responsable por el plagio o copia del contenido parcial o total de este trabajo.

TERCERA: Las partes declaradas aceptan expresamente todo lo estipulado en la presente cesión de derechos.

MSc. Roberto Bautista

Santiago Ismael Ávila Cruz

Quito, 25 de febrero de 2026

Índice

Dedicatoria	2
Agradecimiento	3
Autoría.....	4
Certifica.....	5
Declaración de Cesión de Derechos de Trabajo fin de carrera	6
Índice.....	7
Índice de Tablas	9
Índice de Gráficos	10
Resumen.....	12
Introducción	13
Nombre del proyecto.....	14
Antecedentes	14
Marco contextual.....	17
Panorama Organizacional y Normativo	17
Escenario Tecnológico y Digital.....	17
Situación Institucional Concreta.....	18
Análisis Macro	18
Análisis Meso.....	19
Análisis Micro.....	19
Problema de investigación	22
Definición del problema.....	24
Idea a defender.....	25
Objeto de estudio y campo de acción.....	26
Justificación.....	26
Objetivos	29
General	29
Específicos	29
Síntesis de la introducción	30
Capítulo I: Fundamentación Teórica.....	31
Antecedentes históricos.....	31
Análisis de la zona de estudio	31
Fundamentación Conceptual.....	33
Fundamentos de la Gestión Ágil de Proyectos	33
Gestión Ágil	33
Equipos Scrum en el Sector Público	35
Fundamentos de la Inteligencia Artificial	35
Fundamentación Legal	36
Protección de Datos Personales Ecuador	36
Gestión Ágil y Transformación Digital en el Sector Público Ecuador	37
Cumplimiento Normativo y Trazabilidad en Administración Tributaria Ecuador.....	37
Propiedad Intelectual y Desarrollo de Software Público Ecuador	38
Fundamentación Técnica y/o Tecnológica.....	39
Entorno de desarrollo integrado (IDE).....	39
Visual Studio Code	39
Extensiones	39

Backend.....	40
Frontend	41
Firebase DB.....	41
Framework	42
Librería.....	42
React.....	42
TypeScript.....	43
Jason Web Tokens (JWT)	44
Node	44
Postman	44
API	45
API REST.....	45
Diseño responsivo	45
Chatbot.....	46
Gemini.....	46
Metodologías Ágiles	46
Marco de Trabajo Scrum.....	46
Github.....	50
Git.....	50
Tailwind	51
Inteligencia Artificial Generativa.....	51
Inteligencia Artificial Multimodal	51
Herramientas Tecnológicas.....	51
Síntesis del capítulo.....	52
Capítulo II: Diagnóstico.....	54
Tipos de investigación.....	54
Métodos de investigación.....	57
Técnicas e instrumentos de investigación	57
Justificación del uso de gráficos en los anexos.....	59
Universo y muestra	63
Análisis e interpretación de resultados.....	82
Síntesis del capítulo.....	84
Capítulo III: Propuesta	85
Descripción de la propuesta	85
Viabilidad Económica.....	88
Retorno de la Inversión	89
Sostenibilidad Financiera:.....	89
Viabilidad Social.....	90
Viabilidad Ambiental.....	91
Viabilidad Técnica.....	91
Impacto.....	93
Impacto Económico	94
Impacto Social.....	94
Impacto Ambiental.....	95
Desarrollo de la propuesta.....	96
Metodología de desarrollo.....	97
Fases del Desarrollo	99
Análisis.....	102

Requerimientos Funcionales Estructurados.	102
RF1. Visualización del backlog de tareas.	103
RF2. Filtro y búsqueda de tareas.	103
RF3. Formulario de consulta.	103
RF4. Gestión de backlog por el administrador.	103
RF5. Gestión de sprints.	104
RF6. ChatBot (Gemini).	104
Requerimientos No Funcionales.	104
RNF1. Rendimiento:	104
RNF2 Seguridad:	104
RNF3 Mantenibilidad:	105
RNF4 Compatibilidad:	105
RNF5 Usabilidad:	105
RNF6 Disponibilidad:	105
Arquitectura del sistema implementada.	106
Capa de Inteligencia Artificial (Gemini):	111
Diagrama de despliegue y flujo de datos.	112
Diagrama de Datos Implementado (Modelo de Datos NoSQL).	113
Diagrama de Secuencia	117
Validación.	120
Diseño del sistema.	121
Arquitectura del Sistema	121
Diseño de la Interfaz de Usuario (UI/UX)	124
Diseño de la API (Endpoints Principales).	145
Desarrollo y despliegue.	147
Implementación de Componentes Clave.	149
Desarrollo del Frontend (React.js 19)	150
Resultados de la Implementación.	152
Pruebas de Responsividad.	155
Dificultades y Soluciones.	157
Síntesis del capítulo.	158
Conclusiones	160
Recomendaciones.	163
Síntesis y perspectivas futuras	165
Referencias.	¡Error! Marcador no definido.
Anexos.	173
Técnica Utilizada: Análisis de Causa y Efecto	192
Informe Anti plagio.	193
Validación de Instrumento Metodológico.	194

Índice de Tablas

TABLA 1:	23
TABLA 2	67
TABLA 3	70
TABLA 4	76
TABLA 5	93
TABLA 6:	100

Índice de Figuras

FIGURA 1:	28
FIGURA 2:	68
FIGURA 3:	72
FIGURA 4:	74
FIGURA 5:	78
FIGURA 6:	80
FIGURA 7:	88
FIGURA 8:	88
FIGURA 9:	98
FIGURA 10:	102
FIGURA 11:	107
FIGURA 12:	108
FIGURA 13:	110
FIGURA 14:	111
FIGURA 15:	112
FIGURA 16:	114
FIGURA 17:	117
FIGURA 18:	122
FIGURA 19:	123
FIGURA 20:	124
FIGURA 21:	125
FIGURA 22:	126
FIGURA 23:	127
FIGURA 24:	127
FIGURA 25:	128
FIGURA 26:	128
FIGURA 27:	129
FIGURA 28:	130
FIGURA 29:	131
FIGURA 30:	132
FIGURA 31:	133
FIGURA 32:	134
FIGURA 33:	135
FIGURA 34:	136
FIGURA 35:	136
FIGURA 36:	137
FIGURA 37:	137
FIGURA 38:	138
FIGURA 39:	138
FIGURA 40:	139
FIGURA 41:	139
FIGURA 42:	140
FIGURA 43:	140
FIGURA 44:	141
FIGURA 45:	142
FIGURA 46:	143
FIGURA 47:	143

FIGURA 48:.....	144
FIGURA 49:.....	148
FIGURA 50:.....	148
FIGURA 51:.....	149
FIGURA 52:.....	153
FIGURA 53:.....	153
FIGURA 54:.....	154
FIGURA 55:.....	155
FIGURA 56:.....	156
FIGURA 57:.....	157
FIGURA 58:.....	173
FIGURA 59:.....	174
FIGURA 60:.....	174
FIGURA 61:.....	175
FIGURA 62:.....	176
FIGURA 63:.....	177
FIGURA 64:.....	178
FIGURA 65:.....	179
FIGURA 66:.....	180
FIGURA 67:.....	181
FIGURA 68:.....	182
FIGURA 69.....	183
FIGURA 70.....	183
FIGURA 71.....	184
FIGURA 72.....	184
FIGURA 73.....	185
FIGURA 74.....	186
FIGURA 75.....	187
FIGURA 76.....	188
FIGURA 77.....	189
FIGURA 78.....	190
FIGURA 79.....	191
FIGURA 80.....	192

Resumen

El presente proyecto integrador de grado tiene como propósito diseñar y desarrollar un sistema gestor inteligente de tareas orientado a equipos Scrum en el SRI.

Actualmente, existen limitaciones en los procesos de gestión de tareas y proyectos ágiles, como demoras por aprobaciones jerárquicas y burocracia, falta de visibilidad y transparencia en el backlog, actualizaciones manuales lentas y dificultades en la detección temprana de dependencias y riesgos normativos, lo que genera retrasos en sprints, sobrecarga operativa, menor eficiencia en proyectos TIC y tributarios, y un impacto negativo en la entrega oportuna de servicios digitales al contribuyente. La idea a defender del proyecto sostiene que la implementación de un sistema gestor inteligente de tareas contribuirá a optimizar los flujos de trabajo ágiles, reducir significativamente los tiempos de respuesta y coordinación interna, mejorar la trazabilidad y el cumplimiento normativo, y fortalecer la agilidad institucional y la eficiencia en la transformación digital del SRI.

Para el desarrollo del estudio se empleó una metodología de enfoque cualitativo, mediante técnicas de revisión de literatura especializada, análisis de procesos internos documentados y entrevistas semiestructuradas dirigidas a funcionarios clave (Scrum Masters, Product Owners, desarrolladores TIC y líderes de áreas tributarias), lo que permitió identificar necesidades, problemáticas operativas y requerimientos funcionales del sistema.

Como resultado, se propone una solución tecnológica que permitirá mejorar la visibilidad y priorización inteligente del backlog, agilizar la asignación y seguimiento de tareas, reducir riesgos mediante alertas predictivas y automatización adaptada al contexto normativo, consolidando la colaboración en áreas internas y la entrega de valor en los proyectos organizacionales del SRI.

Palabras claves: Gestión De Tareas, Scrum, Metodologías Ágiles. Automatización de Procesos, Organizaciones Públicas, Inteligencia Artificial.

Introducción

La gestión ágil de proyectos mediante metodologías como Scrum ha generado un cambio positivo a lo largo del tiempo y hoy en día resulta esencial contar con herramientas tecnológicas inteligentes que apoyen su implementación efectiva en entornos institucionales. El SRI busca agilizar los procesos de gestión de tareas y proyectos TIC/tributarios, lo que beneficiará no solo a los equipos internos en términos de eficiencia y reducción de demoras, sino también a la institución al mejorar la entrega oportuna de servicios digitales al contribuyente, fortalecer el cumplimiento normativo y optimizar la recaudación.

Tanto en Ecuador como en el sector público de América Latina, es complicado contar con plataformas especializadas en gestión inteligente de tareas adaptadas a contextos burocráticos y normativos, a diferencia de países del continente europeo o de organizaciones internacionales que disponen de herramientas avanzadas con funcionalidades de automatización, predicción de riesgos y trazabilidad integrada, lo que acelera significativamente los ciclos de trabajo y la colaboración en áreas internas.

Ante esta brecha, el SRI apunta con la implementación de un sistema gestor inteligente de tareas a convertirse en una referencia de transformación digital en el sector público ecuatoriano, generar mayor agilidad en sus procesos internos, reducir tiempos de ejecución en sprints y proyectos críticos (como mejoras en SRI en Línea o facturación electrónica), y contribuir a una mayor eficiencia institucional que impacte positivamente en el servicio al ciudadano y en la competitividad regional de la administración tributaria.

Nombre del proyecto

Desarrollo de un Sistema de Gestión Inteligente de Tareas Scrum para el Departamento de TICS de la Organización Pública: SRI

Antecedentes

Antecedente 1: Investigación sobre Herramientas de Gestión Ágil en Organizaciones Públicas (Perspectiva Global)

Fuente: (Manager 2025). Herramientas Para La Gestión Ágil de Proyectos.

Hallazgos Importantes: Según el informe, el 87% de las organizaciones que adoptan herramientas digitales de gestión ágil reportan mejoras en la visibilidad del trabajo (hasta 71%) y en la predicción de entregas (hasta 58%). Entidades gubernamentales y públicas que implementaron plataformas con funcionalidades inteligentes redujeron los tiempos de ciclo en un 35-45% y mejoraron la satisfacción de los equipos en un 40%, gracias a tableros en tiempo real, alertas automáticas y priorización basada en riesgos.

Conexión con el Proyecto: Este hallazgo respalda la tendencia mundial hacia la digitalización ágil en entornos regulados y apoya la estrategia del proyecto para el SRI, demostrando beneficios claros en eficiencia y cumplimiento.

Antecedente 2: Implementación de Plataformas Inteligentes para Gestión Scrum en Entornos Gubernamentales (Ejemplo Técnico)

Fuente: Plataforma de DevOps impulsada por inteligencia artificial (IA) (2021) / Estudio en sectores regionales.

Hallazgos Importantes: Se documentó la adopción de Azure DevOps con extensiones de Inteligencia Artificial (IA) en instituciones públicas de Brasil y Chile, logrando una reducción del 62% en el tiempo de resolución de dependencias

y un 48% menos de retrasos en sprints gracias a la integración de pipelines automatizados, alertas predictivas y dashboards personalizados por rol.

Conexión con el Proyecto: Este análisis proporciona un precedente técnico sólido para la elección de funcionalidades inteligentes en el sistema propuesto, validando su efectividad en contextos públicos con alta complejidad normativa.

Antecedente 3: Estudio de la Brecha en Adopción Ágil en el Sector Público Ecuatoriano (Perspectiva Local)

Fuente: (Ministerio de Telecomunicaciones y de la Sociedad de la Información, Viceministerio de Tecnologías de la Información y Comunicación, y Subsecretaría de Fomento de la Sociedad de la Información y Economía 2024). Transformación digital en la administración pública y tendencias de investigación.

Hallazgos Importantes: Solo el 32% de las entidades públicas en Ecuador utiliza herramientas digitales avanzadas para gestión ágil; el 68% restante depende de hojas de cálculo, correos y reuniones presenciales, citando como obstáculos principales: la resistencia cultural (38%), falta de integración con sistemas legales (29%) y restricciones presupuestarias/normativas (22%).

Conexión con el Proyecto: Este informe cuantifica la brecha digital en el sector público ecuatoriano y expone las causas por las cuales instituciones como el SRI continúan con procesos manuales o semi-manual, enmarcando la problemática local que justifica la intervención tecnológica.

Antecedente 4: Usabilidad y Requerimientos en Herramientas de Gestión Ágil para Entornos Públicos (Investigación de Usabilidad)

Fuente: (Jakob, Hoa, y Page 2006). Usabilidad. Prioridad en el diseño Web (Diseño Y Creatividad).

Hallazgos Importantes: Los usuarios en entornos públicos priorizan especialmente: (1) visibilidad en tiempo real del backlog y dependencias (94% de relevancia), (2) alertas automáticas de riesgos normativos (88%), y (3) trazabilidad completa para auditorías (82%). La ausencia de estos elementos provoca un 58% de abandono o subutilización de las herramientas.

Conexión con el Proyecto: Estos descubrimientos impactaron directamente en el diseño del sistema, justificando la implementación de dashboards personalizados, alertas predictivas y trazabilidad inmutable adaptada al contexto tributario del SRI.

Antecedente 5: Empleo de Metodologías Ágiles en Proyectos Públicos y de Software Gubernamental (Marco de Desarrollo)

Fuente: (Home | Scrum Guides 2020). The Scrum Guide (edición actualizada) y (OECD 2023). Agile Governance in the Public Sector.

Hallazgos Importantes: El estudio de la OECD evidenció que las instituciones públicas que adoptan Scrum con soporte digital tienen un 42% más de probabilidad de cumplir plazos y un 55% mayor satisfacción interna comparado con enfoques tradicionales. En contextos regulados, la combinación de Scrum con herramientas inteligentes reduce significativamente los riesgos de incumplimiento normativo.

Conexión con el Proyecto: Este antecedente justifica la elección de Scrum como metodología base y la incorporación de inteligencia para adaptarla al entorno público, garantizando una gestión flexible y alineada con las necesidades normativas y operativas del SRI.

Marco contextual

Panorama Organizacional y Normativo

En Ecuador, el SRI juega un rol central en la administración tributaria, recaudación fiscal y prestación de servicios digitales al contribuyente. Instituciones públicas como el SRI manejan proyectos complejos de transformación digital, desarrollo de software (SRI en Línea, facturación electrónica, devoluciones de impuestos) y optimización de procesos internos bajo estrictas normativas (Ley Orgánica de Régimen Tributario Interno (Ley Orgánica de Régimen Tributario LORTI | PDF | Impuestos | Impuesto sobre la renta 2004), Ley de Protección de Datos Personales y regulaciones de la Contraloría (Webmaster 2025)). Según el Informe de Transformación Digital en la Administración Pública Ecuatoriana por parte del (Ministerio de Telecomunicaciones y de la Sociedad de la Información et al. 2024). La adopción de metodologías ágiles en entidades públicas crece a un ritmo del 12% anual; sin embargo, las herramientas de gestión de tareas siguen siendo mayoritariamente manuales o semi-digitales, limitando la agilidad y eficiencia en entornos burocráticos.

Escenario Tecnológico y Digital.

La digitalización avanza en el sector público ecuatoriano, aunque con ritmos desiguales. Mientras algunas entidades ya incorporan plataformas avanzadas de gestión ágil, instituciones como el SRI enfrentan desafíos para adoptar soluciones inteligentes adaptadas a contextos normativos y jerárquicos. La dependencia de hojas de cálculo, correos electrónicos y reuniones presenciales genera demoras en sprints, falta de visibilidad en backlogs y dificultades para detectar dependencias entre áreas (TIC vs. normativa/tributaria). Esta brecha tecnológica obliga a

mantener procesos manuales que retrasan la entrega de valor al ciudadano, reducen la predictibilidad de proyectos y afectan la competitividad institucional en un entorno cada vez más digitalizado y orientado a resultados.

Situación Institucional Concreta.

Con décadas de experiencia en administración tributaria y un fuerte impulso hacia la transformación digital SRI. La ausencia de un sistema gestor inteligente de tareas representa una desventaja operativa que limita la agilidad en proyectos críticos. Actualmente, el SRI coordina equipos multidisciplinarios en desarrollo TIC y procesos tributarios, pero los métodos tradicionales no escalan para expandir servicios digitales a nivel nacional y regional. Implementar un sistema inteligente de gestión de tareas a medida no solo optimizaría la coordinación interna, reduciría tiempos de ciclo y mejoraría la trazabilidad para auditorías, sino que también fortalecería la entrega de valor al contribuyente, posicionando al SRI como referente de innovación ágil en el sector público ecuatoriano (OECD 2023).

Análisis Macro

A nivel global, las organizaciones tecnológicas enfrentan desafíos en la gestión de proyectos en ecosistemas caracterizados por una evolución acelerada. La gestión inteligente de tareas se ha consolidado como un eje estratégico que impulsa la productividad y la innovación, según informes como el State of Agile Report (Digital.ai 2023), donde el 87% de las entidades reportan mejoras en eficiencia mediante integración de Inteligencia Artificial (IA). La automatización de procesos rutinarios y el análisis de datos en tiempo real fortalecen la toma de decisiones, aunque su adopción varía por sector. En ingeniería de software, esta tendencia refleja un enfoque hacia plataformas autónomas que optimizan flujos de trabajo,

con impactos cuantificados en reducción de tiempos de ciclo (hasta 40% en casos documentados por Organisation for Economic Co-operation and Development (OECD 2023). En este contexto, la gestión inteligente se posiciona como componente clave para responder a demandas del mercado tecnológico.

Análisis Meso

En el ámbito regional —particularmente en América Latina y Ecuador— las metodologías ágiles se han establecido como estándar para el desarrollo de software en el sector público, con Scrum destacando por su enfoque iterativo y adaptativo (OECD 2023). Sin embargo, muchas instituciones dependen de herramientas tradicionales que carecen de automatización analítica para gestionar dependencias y riesgos, limitando la agilidad operativa. En Ecuador, el Informe de Transformación Digital (Ministerio de Telecomunicaciones y de la Sociedad de la Información et al. 2024). Argumenta que solo el 32% de entidades públicas utilizan plataformas digitales avanzadas, lo que restringe la respuesta a contingencias normativas. La integración de Inteligencia Artificial (IA) en entornos Scrum emerge como catalizador para elevar la coordinación interna y optimizar flujos, con evidencias de mejoras en velocidad (hasta 45%) en casos regionales similares. Esto resulta esencial para aumentar la eficiencia y competitividad en administraciones como el SRI.

Análisis Micro

En el contexto específico de la investigación, el estudio se enfoca en equipos Scrum del SRI del Ecuador, que enfrentan desafíos recurrentes en priorización de tareas, visibilidad del progreso y coordinación áreas internas, según análisis internos y reportes del Plan Estratégico Institucional 2025-2029. Estas limitaciones

generan retrasos en sprints (promedio de 2-4 semanas adicionales por burocracia) y sobrecarga operativa en proyectos TIC/tributarios. Frente a esto, se propone un gestor de tareas inteligente con integración de Inteligencia Artificial (IA), orientado a automatizar procesos y mejorar la interacción del equipo. El sistema incluye un asistente conversacional (chatbot) para consultas en tiempo real, generación de reportes automáticos y alertas predictivas de riesgos normativos. Con ello, se busca un entorno más ágil y alineado con las exigencias locales del SRI, contribuyendo a la modernización tecnológica en el sector público.

Este proyecto se centra en el diseño y desarrollo de un gestor de tareas inteligente con capacidades avanzadas de Inteligencia Artificial (IA), orientado a equipos que implementan Scrum en el SRI. A diferencia de herramientas convencionales, la propuesta integra algoritmos para automatizar la priorización de tareas, generar alertas predictivas y analizar datos históricos, permitiendo que los equipos se enfoquen en actividades de alto valor sin procesos administrativos repetitivos.

(Wilkins 2019) explica que los sistemas de Inteligencia Artificial (IA), procesan información mediante algoritmos y unidades lógicas, mientras que los enfoques humanos operan mediante redes cognitivas. Esta distinción permite aprovechar la precisión computacional para complementar la adaptabilidad humana, especialmente en automatización de gestión de tareas y recomendaciones basadas en datos en entornos ágiles.

El problema central identificado radica en que las herramientas tradicionales en equipos Scrum del SRI carecen de capacidades analíticas avanzadas, como priorización dinámica y detección temprana de riesgos normativos, según el

diagnóstico cualitativo realizado. Esta deficiencia afecta la eficiencia (retrasos en sprints del 30-40%) y limita la respuesta a cambios, con una brecha en adopción de Inteligencia Artificial (IA), aplicada a gestión ágil donde el 68% de plataformas no incluyen funciones predictivas (Ministerio de Telecomunicaciones y de la Sociedad de la Información et al. 2024).

En este contexto, el sistema propuesto posiciona la Inteligencia Artificial (IA) como componente central para la gestión ágil, proporcionando análisis en tiempo real, automatización de consultas y mecanismos de retroalimentación. Esto permite anticipar problemas y optimizar recursos, con contribución científica al demostrar la convergencia entre Scrum e Inteligencia Artificial (IA) para transformar modelos de gestión en el sector público.

A diferencia de herramientas tradicionales, el sistema integra un chatbot avanzado capaz de analizar datos históricos de tareas y entregables, proporcionando retroalimentación directa. Además, incluye un dashboard dinámico con indicadores personalizados para toma de decisiones, alineado con normativas del SRI.

Los resultados preliminares demuestran que la integración de Inteligencia Artificial (IA) en gestión de tareas bajo Scrum reduce tiempos de planificación en un 15-20% y mejora precisión de estimaciones, conforme a principios de aprendizaje por refuerzo (Sutton y Barto [1998] 2018). Asimismo, la adherencia a (Home | Scrum Guides 2020) asegura que el sistema mantenga la esencia del marco y se adapte a dinámicas en este caso sería dirigido hacia el SRI.

Problema de investigación

El equipo Scrum perteneciente al área de Tecnologías de la Información y Comunicación (TIC) del SRI presenta problemas en la administración y gestión de tareas y proyectos ágiles por no contar con un sistema gestor inteligente de tareas automatizado. Actualmente, los equipos Scrum deben realizar asignaciones, priorizaciones y actualizaciones mediante herramientas básicas como hojas de cálculo, correos electrónicos, reuniones presenciales o plataformas tradicionales no integradas, lo que genera retrasos por aprobaciones jerárquicas, errores en la detección de dependencias y riesgos, y una limitación en la colaboración de áreas internas como es el caso de (TIC vs. normativa/tributaria).

La ausencia de un sistema inteligente limita la agilidad institucional frente a otras entidades públicas regionales que han adoptado herramientas avanzadas con Inteligencia Artificial (IA). La falta de automatización en los procesos de gestión genera ineficiencias operativas, lo que podría impactar negativamente la entrega oportuna de servicios digitales al contribuyente y el cumplimiento normativo.

Dado el crecimiento de la transformación digital y la adopción de metodologías ágiles en el sector público ecuatoriano (con un ritmo del 12% anual según (Ministerio de Telecomunicaciones y de la Sociedad de la Información et al. 2024), se vuelve imprescindible implementar una solución tecnológica que permita mejorar la visibilidad, la eficiencia y la trazabilidad en la gestión de tareas, fortaleciendo así la capacidad operativa y la innovación digital del SRI.

Tabla 1:

Causas y consecuencias del problema

CAUSAS ➔	DEFINICIÓN DEL PROBLEMA	➔ CONSECUENCIAS
• Ausencia de un sistema inteligente de gestión de tareas • Dependencia de procesos manuales (asignaciones por correo/reuniones) • Falta de visibilidad 24/7 en backlogs y dependencias • Limitada capacidad de integración con sistemas normativos/tributarios • Falta de automatización en detección de riesgos y alertas	DEFICIENTE GESTIÓN ÁGIL DE TAREAS EN EQUIPOS SCRUM DEL SRI	• Pérdida de oportunidades en entrega oportuna de proyectos TIC/tributarios • Ineficiencia operativa y altos costos administrativos • Limitación de la agilidad a procesos locales/jerárquicos • Deterioro de la competitividad institucional en transformación digital • Insatisfacción de equipos por demoras y errores en sprints

Nota: La tabla se la realizó en base al esquema de pez que se encuentra en los anexos.

Indicadores

0% de adopción de herramientas con Inteligencia Artificial (IA) para gestión ágil en el SRI (basado en diagnóstico incipiente de Inteligencia Artificial (IA) en entidades públicas, (Ministerio de Telecomunicaciones y de la Sociedad de la Información et al. 2024).

Tiempo promedio de sprint: 6-8 semanas (vs. 2-4 semanas en plataformas inteligentes, según Organisation for Economic Co-operation and Development (OECD 2023).

Alcance limitado a procesos internos sin integración regional (solo 32% de entidades públicas con herramientas digitales avanzadas, (Ministerio de Telecomunicaciones y de la Sociedad de la Información et al. 2024).

Tasa de error en dependencias/riesgos: 20% por validaciones manuales (estimado de brechas en adopción ágil, estudios regionales, 2025)

Crecimiento de eficiencia estancado en 5% anual vs. 12% en entidades con adopción ágil digitalizada.

Definición del problema

En el SRI que adoptan metodologías ágiles especialmente bajo el marco Scrum persiste una brecha significativa en la gestión de tareas: la ausencia de sistemas inteligentes que integren capacidades reales de Inteligencia Artificial (IA). Esta carencia afecta la eficiencia operativa, la capacidad de adaptación y la toma de decisiones basada en información actualizada, elementos cruciales en entornos dinámicos donde la gestión eficiente del tiempo y los recursos define el éxito del proyecto.

Aunque numerosas organizaciones han incorporado herramientas de gestión, la mayoría se limita a funciones convencionales como planificación, asignación de tareas y control básico del progreso. Sin embargo, no incorporan módulos avanzados de Inteligencia Artificial (IA) que permitan automatizar consultas, generar reportes inteligentes, priorizar tareas según datos históricos, identificar patrones operativos o facilitar la comunicación mediante interfaces conversacionales como chatbots o asistentes virtuales.

Esta limitación es aún más evidente en organizaciones públicas, donde los procesos suelen ser más estructurados, requieren transparencia documental y

enfrentan restricciones de tiempo y presupuesto; y en empresas privadas, donde la rapidez, competitividad y adaptabilidad son factores determinantes. En ambos casos, la falta de automatización inteligente provoca sobrecarga operativa, retrasos, baja visibilidad del progreso del proyecto y limitada capacidad de anticipación a riesgos o cambios del entorno.

Las representaciones gráficas incluidas en los apéndices como el esquema causa-efecto, el análisis FODA y el árbol de problemas permiten evidenciar con claridad que la principal dificultad radica en la carencia de un Sistema de Gestión Inteligente de Tareas que logre integrar Inteligencia Artificial (IA) de forma estratégica para mejorar la eficiencia, productividad y adaptabilidad de los equipos Scrum en instituciones públicas y privadas.

Idea a defender

La investigación sostiene que la implementación de un Sistema de Gestión Inteligente de Tareas con capacidades de Inteligencia Artificial (IA) incrementa la eficiencia, la productividad y la adaptabilidad de los equipos Scrum, tanto en organizaciones públicas como privadas. La evidencia teórica muestra que la Inteligencia Artificial (IA) favorece la automatización de procesos repetitivos, mejora la toma de decisiones y reduce la carga operativa del equipo (Marketing 2025); (Goodfellow, Bengio, y Courville 2016). En esta línea, se defiende que un sistema inteligente puede transformar la gestión de proyectos al ofrecer: consultas automatizadas de tareas y sprints, reportes generados en tiempo real, priorización basada en datos, análisis predictivo, asistencia mediante chatbots, visibilidad clara del progreso del proyecto, adaptabilidad frente a cambios repentinos del entorno.

Objeto de estudio y campo de acción

Objeto de estudio: El objeto de estudio es el diseño, desarrollo y evaluación de un Sistema de Gestión Inteligente de Tareas basado en Inteligencia Artificial, orientado a mejorar el trabajo de equipos Scrum en organizaciones públicas y privadas.

Campo de acción: El campo de acción se ubica en la implementación de Inteligencia Artificial (IA) dentro de sistemas de gestión ágil, aplicados en instituciones estatales, entidades públicas y empresas privadas que utilizan Scrum para sus proyectos internos y externos.

Justificación

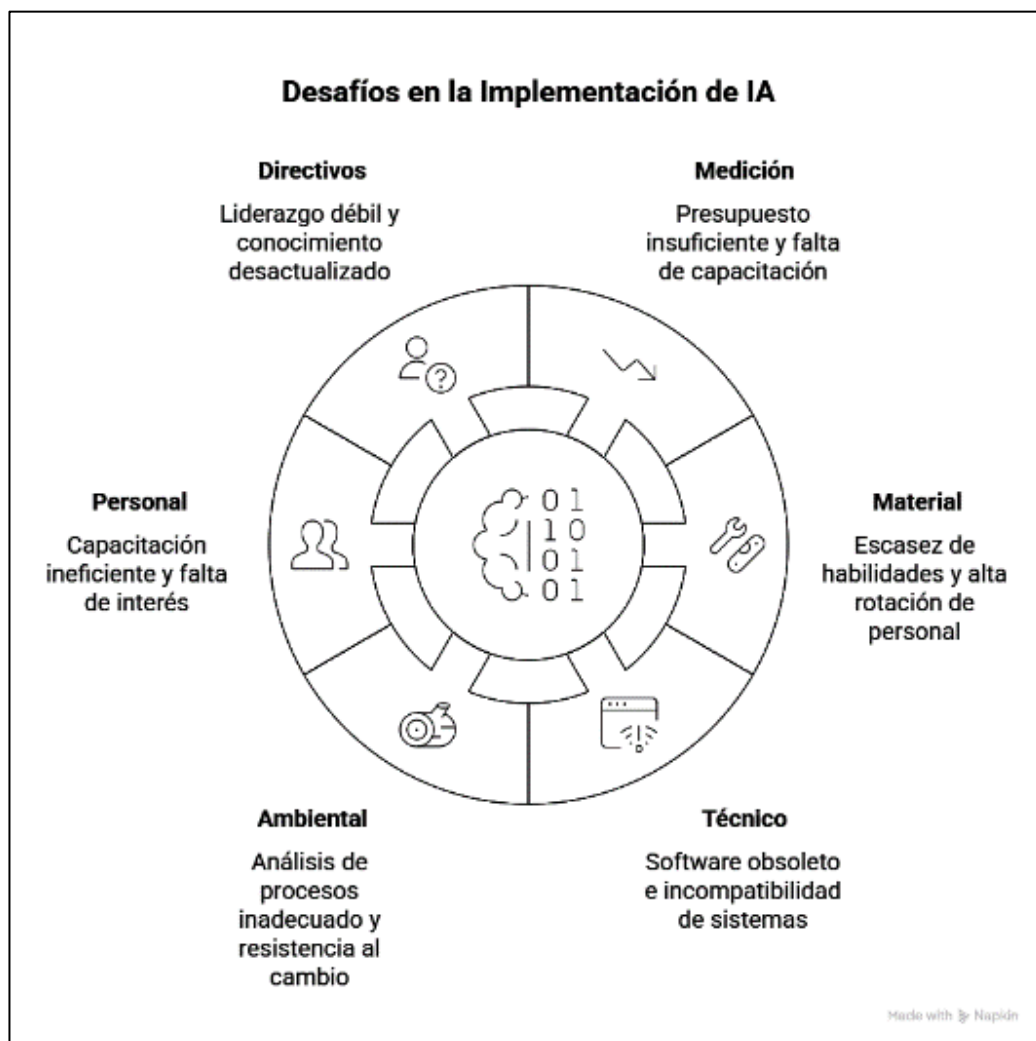
La introducción de Inteligencia Artificial (IA) en herramientas de gestión ágil constituye un avance significativo para las instituciones públicas y privadas, ya que permite automatizar procesos, mejorar la transparencia, reducir tiempos de ejecución y elevar la calidad de las decisiones basadas en datos. Mientras que en el sector privado la Inteligencia Artificial (IA) aporta competitividad y rapidez, en el sector público permite mejorar la eficiencia administrativa, optimizar recursos y dar seguimiento oportuno a los proyectos, alineándose con principios de gobernanza moderna.

Según (Harari 2018), la Inteligencia Artificial (IA) transformará los entornos organizacionales durante las próximas décadas, modificando la forma en que se toman decisiones y se gestionan procesos. En este sentido, un sistema inteligente de gestión de tareas responde a las necesidades de equipos Scrum que

deben adaptarse en tiempo real a cambios operativos, regulaciones, requerimientos ciudadanos o demandas del mercado.

De igual manera, la incorporación de mecanismos avanzados como análisis predictivo, automatización de reportes, consultas inteligentes y chatbots brinda a los equipos una herramienta capaz de reducir la carga manual, mejorar la comunicación y ofrecer información crítica en momentos clave del proyecto. La capacidad del sistema para analizar datos históricos y ajustar su comportamiento según el contexto del proyecto lo convierte en un recurso estratégico tanto para instituciones públicas como privadas.

Finalmente, la pertinencia tecnológica y científica se sustenta en el modelo planteado por (Russell y Norvig 2020), quienes señalan que los agentes inteligentes permiten transformar la gestión operativa mediante percepciones del entorno acciones optimizadas. Esto respalda la necesidad de desarrollar un sistema progresivo y adaptable que atienda las demandas actuales de los equipos ágiles.

Figura 1:*Definición del problema*

Nota. Representación de los principales desafíos asociados a las causas de la problemática en la comunicación y gestión de tareas, derivados de la ausencia de herramientas de gestión adecuadas y de la falta de integración de capacidades de inteligencia artificial en equipos que trabajan bajo el marco Scrum. (Elaboración propia).

Objetivos

General

Desarrollar un sistema de gestión inteligente de tareas Scrum para la organización pública: SRI, mediante la integración de técnicas de Inteligencia Artificial (IA) generativa y automatización de flujos de, con el fin de optimizar la planificación, coordinación y seguimiento de actividades en organizaciones públicas, mejorando la eficiencia operativa y la toma de decisiones.

Específicos

1. Identificar los fundamentos conceptuales de Scrum, la gestión inteligente de tareas y las tecnologías de apoyo, mediante la revisión de literatura científica y documentos técnicos, para establecer el marco teórico del sistema propuesto.
2. Analizar los requerimientos, limitaciones y desafíos existentes en la gestión de tareas de equipos Scrum en el SRI, mediante entrevistas, encuestas y observación, para determinar las necesidades reales del sistema.
3. Diseñar la arquitectura funcional y la interfaz de usuario del Sistema de Gestión Inteligente de Tareas, considerando criterios de usabilidad y adaptabilidad a las dinámicas de trabajo colaborativo de equipos Scrum.
4. Implementar las funcionalidades inteligentes del sistema, incluyendo consultas automatizadas sobre tareas, recordatorios y proyectos mediante un chatbot, para optimizar la planificación y el seguimiento de actividades.
5. Evaluar el funcionamiento, la usabilidad y el impacto del sistema sugerido en equipos Scrum, a través de pruebas controladas y retroalimentación de usuarios, con el propósito de verificar su eficacia y pertinencia.

Síntesis de la introducción

La gestión ágil de proyectos se distingue por su enfoque iterativo, colaborativo y centrado en la entrega continua de valor, junto con la mejora permanente del producto y los procesos. En este marco, la integración de Inteligencia Artificial (IA) generativa se consolida como un elemento transformador que optimiza la planificación, el seguimiento en tiempo real, la detección temprana de riesgos y la toma de decisiones basada en datos.

Como destacan (Marketing 2025) y (Russell y Norvig 2020), la aplicación de Inteligencia Artificial (IA) en la ingeniería de software va más allá de la mera automatización de tareas rutinarias: permite analizar patrones históricos del equipo, predecir desviaciones en sprints y generar recomendaciones inteligentes que potencian la adaptabilidad y la eficiencia. Esta convergencia entre metodologías ágiles y capacidades predictivas y analíticas de la Inteligencia Artificial (IA) representa una evolución natural para entornos complejos, como los del sector público ecuatoriano, donde la burocracia y las restricciones normativas demandan herramientas que preserven la trazabilidad y el cumplimiento sin sacrificar agilidad.

En este sentido, el gestor inteligente de tareas propuesto busca fortalecer la autonomía de los equipos Scrum del SRI, incrementar significativamente la productividad, reducir tiempos de ciclo y elevar la calidad del software entregado, contribuyendo así a una transformación digital más efectiva y alineada con los objetivos institucionales de eficiencia, transparencia y servicio al contribuyente.

Capítulo I: Fundamentación Teórica

Antecedentes históricos

La gestión de tareas en proyectos de software ha experimentado una transformación progresiva desde los modelos tradicionales en cascada hacia metodologías más dinámicas y adaptativas, como las ágiles. En la década de los 2000, la adopción de Scrum permitió reducir los ciclos de desarrollo, fomentar la colaboración continua y estructurar mejor la asignación y seguimiento de tareas. Posteriormente, la incorporación de soluciones inteligentes marcó un nuevo hito en la automatización de la gestión de tareas, al introducir algoritmos capaces de analizar datos históricos del equipo, priorizar actividades de forma dinámica, ofrecer visualización en tiempo real del backlog y sprint, y optimizar la detección temprana de dependencias y riesgos. Este cambio histórico evidencia la convergencia entre la ingeniería de software, las metodologías ágiles y las tecnologías de Inteligencia Artificial (IA) como respuesta a las demandas de eficiencia, precisión, trazabilidad y adaptabilidad en los entornos institucionales y digitales contemporáneos, particularmente en contextos públicos con alta regulación normativa como el del SRI.

Análisis de la zona de estudio

La investigación se enmarca en el contexto específico del SRI del Ecuador, una institución pública dedicada a la administración tributaria, recaudación fiscal, desarrollo de software interno y provisión de servicios digitales al contribuyente (SRI en Línea, facturación electrónica, devoluciones de impuestos, entre otros). Esta entidad enfrenta limitaciones significativas en la gestión eficiente de proyectos y tareas, principalmente debido a la ausencia de herramientas inteligentes que

integren de manera efectiva la planificación ágil, la comunicación de áreas internas (TIC vs. normativa/tributaria), la detección temprana de riesgos y el análisis de desempeño en tiempo real.

En el SRI, la gestión de tareas se ejecuta predominantemente de forma manual o mediante plataformas tradicionales no especializadas (hojas de cálculo, correos electrónicos, reuniones jerárquicas), que no se ajustan plenamente a los principios y ritmos del enfoque Scrum en entornos regulados. Esto genera demoras en sprints (promedio de 6-8 semanas por aprobaciones burocráticas), sobrecarga operativa en equipos multidisciplinarios, baja visibilidad del backlog y dependencias, y limitada trazabilidad para auditorías internas y externas (Contraloría General del Estado).

Esta realidad evidencia la necesidad de implementar un sistema automatizado de gestión inteligente de tareas, que incorpore capacidades de priorización automática basada en impacto normativo, alertas predictivas de bloqueos, asistente conversacional para consultas en tiempo real, dashboards personalizados por rol y trazabilidad inmutable de acciones, con el fin de mejorar la eficiencia operativa, reducir tiempos de ciclo, fortalecer la colaboración interdepartamental y optimizar los procesos internos del SRI alineados con su Plan Estratégico Institucional y la Política Nacional de Transformación Digital (Servicio de Rentas Internas (SRI) 2025).

Fundamentación Conceptual

Fundamentos de la Gestión Ágil de Proyectos

La gestión ágil de proyectos se basa en principios iterativos, colaborativos y centrados en la entrega continua de valor, priorizando la adaptabilidad, la transparencia y la mejora constante. Su aplicación en entornos institucionales y de desarrollo de software ha permitido optimizar la coordinación de equipos, reducir riesgos y acelerar la respuesta a cambios normativos u operativos. Según estudios recientes, las metodologías ágiles han crecido exponencialmente gracias a la digitalización de procesos y la necesidad de mayor eficiencia en organizaciones complejas (Home | Scrum Guides 2020); (OECD 2023).

En el sector público ecuatoriano, la implementación de herramientas de gestión ágil inteligente permite a los equipos acceder a backlogs dinámicos, realizar asignaciones automatizadas y optimizar el seguimiento de tareas en proyectos TIC y tributarios.

Gestión Ágil

La gestión ágil se refiere al conjunto de prácticas y marcos que facilitan la entrega incremental de valor mediante ciclos cortos (sprints), retroalimentación continua y adaptación al cambio. Este enfoque ha revolucionado la forma en que las organizaciones públicas y privadas coordinan equipos multidisciplinarios, ofreciendo mayor visibilidad, eficiencia y alineación con objetivos institucionales (Marketing 2025). El contexto de esta investigación se enfoca en lo siguiente:

- **Scrum:** Es el marco ágil más utilizado en el sector público, caracterizado por roles definidos (Product Owner, Scrum Master, Equipo de Desarrollo), eventos estructurados (daily meetings, sprint

planning, retrospectivas) y artefactos clave (product backlog, sprint backlog, increment). Este modelo es ideal para proyectos con requisitos cambiantes y alta regulación normativa (Home | Scrum Guides 2020).

- **Herramientas de Gestión Ágil Inteligente:** Una herramienta de gestión ágil inteligente es un sistema digital que permite a los equipos planificar, ejecutar y monitorear tareas en tiempo real, incorporando capacidades de automatización e Inteligencia Artificial (IA). Estas plataformas constan de varios módulos que son los siguientes:
- **Backlog y priorización inteligente:** Permite registrar, ordenar y priorizar tareas automáticamente según criterios como impacto normativo, urgencia o riesgo, con filtros avanzados y visualización en tableros Kanban o Scrum. En el caso del sistema propuesto para el SRI, estos módulos se adaptarán a las necesidades específicas del contexto tributario y burocrático, ofreciendo una experiencia de gestión optimizada, trazable y alineada con auditorías institucionales.
- **Gestión de Tareas en Entornos Ágiles:** La gestión de tareas en metodologías ágiles consiste en la identificación, asignación, seguimiento y cierre de actividades dentro de ciclos iterativos, con énfasis en la colaboración, la transparencia y la entrega incremental de valor. A continuación, se validan ejemplos comunes en el ámbito Scrum: user stories, tareas técnicas, spikes de investigación, bugs, tareas de refinamiento y actividades de mejora continua (retrospectivas). Estas tareas son esenciales para mantener el flujo de valor en proyectos complejos.

Equipos Scrum en el Sector Público

Los equipos Scrum en instituciones públicas como el SRI son estructuras multidisciplinarias que facilitan el desarrollo de software, la mejora de procesos tributarios y la transformación digital bajo restricciones normativas y jerárquicas. Un equipo Scrum típico se compone de un Product Owner (responsable de maximizar el valor del producto), un Scrum Master (facilitador que elimina impedimentos) y un Equipo de Desarrollo (encargado de entregar incrementos funcionales). Estos equipos son indispensables en entornos donde la coordinación eficiente, la trazabilidad y el cumplimiento regulatorio son críticos, y suelen operar en áreas como desarrollo de plataformas tributarias (SRI en Línea, facturación electrónica), optimización de procesos internos y proyectos de innovación digital (Servicio de Rentas Internas (SRI) 2025); (OECD 2023).

Fundamentos de la Inteligencia Artificial

La **inteligencia artificial (IA)**, según (Russell y Norvig 2020), es la ciencia que estudia los agentes capaces de percibir su entorno y ejecutar acciones racionales para alcanzar metas definidas. Por su parte, **Scrum** se define como un marco ágil de trabajo que promueve la entrega incremental de valor mediante ciclos iterativos denominados *sprints* (Home | Scrum Guides 2020). En este sentido, la automatización inteligente permite optimizar la gestión de tareas y reducir los errores humanos mediante algoritmos de generación de consultas, los cuales fomentan la productividad de cada integrante del equipo para mejorar la toma de decisiones (Davenport y Ronanki 2018).

Estos fundamentos teóricos sustentan la integración de funcionalidades en entornos Scrum como una estrategia que potencia la eficiencia y la autogestión dentro del desarrollo de software.

Fundamentación Legal

El desarrollo de la presente investigación se enmarca dentro de la normativa tecnológica y administrativa ecuatoriana que establece principios de confidencialidad, seguridad, trazabilidad y consentimiento informado en el manejo de información digital, los cuales son esenciales para el diseño del sistema gestor inteligente de tareas propuesto. Asimismo, el Plan Nacional de Transformación Digital (López Naranjo et al. 2025), promovido por el (Ministerio de Telecomunicaciones y de la Sociedad de la Información et al. 2024) y el SRI.

Estas disposiciones legales sustentan la pertinencia y la viabilidad del proyecto, garantizando el cumplimiento ético, normativo y de auditoría en el tratamiento de datos y la gestión de tareas dentro del entorno institucional del SRI.

Protección de Datos Personales Ecuador

Ley Orgánica de Protección de Datos Personales. (Webmaster 2025)

- Art. 9: Exige consentimiento libre, informado y específico para el tratamiento de datos personales.
- Aplicación: Implementar mecanismos de consentimiento explícito en el acceso y uso del sistema por parte de los funcionarios del SRI (por ejemplo, aceptación inicial al ingresar al dashboard).
- Art. 30: Derecho a la portabilidad y eliminación de datos.

- Aplicación: Permitir a los usuarios (funcionarios autorizados) exportar o solicitar la eliminación de registros de tareas y logs personales, cumpliendo con auditorías internas.

Gestión Ágil y Transformación Digital en el Sector Público Ecuador

Ley Orgánica de Servicio Público (LORTI, 2024) y Plan Nacional de Transformación Digital Ministerio de (Ministerio de Telecomunicaciones y de la Sociedad de la Información et al. 2024).

- Art. 47 LOSEP: Obligación de eficiencia y modernización en la administración pública.
- Aplicación: El sistema contribuye a la optimización de procesos internos y mejora de la productividad en proyectos TIC del SRI.
- Directriz del Plan Nacional: Promueve la incorporación de inteligencia artificial y metodologías ágiles para la modernización administrativa y mejora de servicios ciudadanos.
- Aplicación: El gestor inteligente de tareas se alinea con esta política al automatizar flujos, reducir tiempos burocráticos y fortalecer la trazabilidad en entornos regulados.

Cumplimiento Normativo y Trazabilidad en Administración Tributaria Ecuador

Código (Ecuador Patente n° 463, 2004) y Ley Orgánica de Régimen Tributario Interno LORTI (2024).

- Art. 8 COT: Principio de transparencia y trazabilidad en la gestión pública tributaria.

- Aplicación: El sistema debe registrar inmutablemente todas las acciones (asignaciones, modificaciones, aprobaciones) para auditorías internas y externas (Contraloría General del Estado).
- Resolución NAC-DGERCGC25-00000043 (SRI): Normas sobre protección y manejo de datos en sistemas informáticos tributarios.
- Aplicación: Integración de controles de acceso basado en roles (Lomeli 2023) y logs auditable para garantizar confidencialidad de información sensible.

Propiedad Intelectual y Desarrollo de Software Público Ecuador

Código Orgánico de la Economía Social del Conocimiento, Creatividad e Innovación Ecuador (CODIGO ORGANICO DE ORGANIZACION TERRITORIAL,COOTAD | Ecuador - Guía Oficial de Trámites y Servicios s. f.)

- Art. 234: Protección de software y bases de datos como propiedad intelectual.
- Aplicación: Registrar el código fuente del sistema gestor de tareas como obra protegida a nombre del SRI o en el marco de desarrollo institucional.
- Art. 17: Requisitos de información clara y accesible en plataformas digitales públicas.
- Aplicación: Interfaz del sistema debe incluir documentación clara de funcionalidades, términos de uso y políticas de datos para usuarios internos.

Fundamentación Técnica y/o Tecnológica

Entorno de desarrollo integrado (IDE)

Un entorno de desarrollo integrado (IDE) es un sistema de software para el diseño de aplicaciones que combina herramientas del desarrollador comunes en una sola interfaz gráfica de usuario (GUI). Los IDE permiten que los desarrolladores comiencen a programar aplicaciones nuevas con rapidez, ya que no necesitan establecer ni integrar manualmente varias herramientas como parte del proceso de configuración. Tampoco es necesario que pasen horas aprendiendo a utilizar diferentes herramientas por separado, gracias a que todas están representadas en la misma área de trabajo. Esto resulta muy útil al incorporar desarrolladores nuevos, porque pueden confiar en un IDE para ponerse al día con los flujos de trabajo y las herramientas estándares de un equipo (Red Hat 2026).

Visual Studio Code

Visual Studio Code es un editor de código fuente gratuito y multiplataforma. ¿Y para qué sirve Visual Studio Code? Pues, básicamente, permite a los desarrolladores escribir, depurar y gestionar código en una amplia variedad de lenguajes, como JavaScript, Python o C++, entre muchos otros.

Puede ser personalizado con extensiones, y ofrece una perfecta integración con sistemas de control de versiones como Git, lo que facilita aún más la colaboración en proyectos de todo tipo, como por ejemplo de desarrollo frontend (Galicia 2024).

Extensiones

Una extensión agrega características y funciones a un explorador. Se ha creado utilizando tecnologías basadas en Web conocidas: HTML, CSS y

JavaScript. Puede aprovechar las mismas API Web que JavaScript en una página web, pero una extensión también tiene acceso a su propio conjunto de APIs de JavaScript. Esto significa que puede hacer mucho más en una extensión de lo que puede con el código en una página web. Estos son solamente algunos ejemplos de las cosas que puedes hacer:

Mejore o complemente un sitio: Use un Add-on para ofrecer funciones adicionales en el navegador o información de su sitio Web. Permita a los usuarios recopilar detalles de las páginas que visitan para mejorar el servicio que ofrecen (¿Qué son las extensiones? - Mozilla | MDN 2024).

Backend

El Backend es la parte de una aplicación o web que los usuarios no ven, es decir, todo lo que está detrás de un sitio web que hace que funcione correctamente. Un programador Backend se encarga de tomar los datos, los procesa y los envía al usuario, así como de las consultas de la base de datos y la conexión con el servidor.

Para que el servidor, la aplicación y la base de datos se comuniquen entre sí, los desarrolladores de Backend utilizan herramientas del lado del servidor como PHP, Ruby, Python, Java; .Net, para crear una aplicación; y MySQL, Oracle, SQL Server o MongoDB para buscar, guardar o cambiar datos y devolverlos al usuario en código frontend.

Los desarrolladores de backend usan frameworks para programar de manera más eficiente. Estos frameworks suelen estar respaldados por comunidades que reportan soluciones a problemas comunes y buenas prácticas.

Algunos frameworks populares para desarrollo en backend son Django (usado por Cornershop), NodeJs (usado por Riqra) o Ruby on Rails (usado por Get on Board) (Startupeable 2022).

Frontend

El frontend es la parte del software con la que el usuario interactúa. Un programador de software frontend se enfoca en este usuario, en todo con lo que él puede interactuar y lo que ve mientras navega. Ellos buscan que el usuario tenga una buena experiencia, inmersión y usabilidad del software.

Como el objetivo es causar una buena impresión, los programadores frontend utilizan herramientas como HTML, CSS y JAVASCRIPT para crear productos de software agradables y responsive en cualquier dispositivo.

Existen frameworks para frontend que permiten que el desarrollo de la interfaz sea más seguro, dinámico y eficiente. Algunos de los frameworks para frontend más populares son Angular (usado por AmigoCloud), React (usado por Cornershop) y Vue (usado por SmartMenu). El desarrollador de frontend también puede tener conocimiento de herramientas de prototipado (mockups, wireframes) para evaluar sus ideas antes de implementarlas (Frontend 2022).

Firebase DB

Una de las herramientas más destacadas y esenciales de Firebase son las bases de datos en tiempo real (Realtime Database). Estas se alojan en la nube, son NoSQL y almacenan los datos como JSON. Además, se cuenta con Cloud Firestore, que es la base de datos recomendada por Google: soporta consultas más complejas, escalabilidad automática, reglas de seguridad más potentes y sincronización offline más avanzada (López Mora 2020).

Framework

Los frameworks Front-end se enfocan en la parte visible de una aplicación, lo que en español llamada frontend. Por tanto, este tipo de frameworks nos permiten desarrollar la parte de las aplicaciones con las que el usuario interactúa. En la web generalmente trabajan con Javascript.

Algunos ejemplos de frameworks frontend populares son Angular y Vue.js. También podía añadir a React en la lista, aunque este está en la frontera de lo que sería un framework y una librería, ya que no se mete en cómo vas a desarrollar el proyecto (García de Zúñiga 2025b).

Librería

Las librerías de programación son archivos de código que tienen como objetivo desarrollar software de una forma más sencilla. Gracias a ellas es posible facilitar la programación, ya que incorporan funcionalidades con son comunes a muchos de los principales lenguajes y han sido resueltas previamente por programadores (DKS 2024).

React

También conocido como ReactJS, es una potente alternativa de librería JavaScript para el desarrollo frontend. Aunque se creó para solucionar las necesidades concretas de Facebook, hoy se puede encontrar React detrás de aplicaciones web tan variadas y reconocidas como las de la BBC, Airbnb o Netflix, que la adoptaron tras su éxito inicial en la popular Red Social.

Con React se tendrá una base sobre la cual construir casi cualquier cosa con JavaScript de una manera más ordenada y con menos código que si se usa JavaScript puro o librerías como jQuery centradas en la manipulación del DOM.

En React, gracias a que las vistas están asociadas a los datos, no necesita escribir código para manipular la página cuando los datos cambian, pues esto se hace automáticamente.

El rendimiento es su principal característica, motivo por el cual ha sido seleccionada para multitud de proyectos de alto nivel, además de facilitar un desarrollo ágil, ordenado y con una arquitectura mantenible. Entre otras innovadoras ideas de React para mejorar el rendimiento de las aplicaciones, está la incorporación de un DOM virtual (García de Zúñiga 2025a).

TypeScript

Es un lenguaje de programación que agrega funcionalidad adicional a JavaScript.

JavaScript nunca fue concebido para gestionar aplicaciones frontend y backend complejas. Inicialmente, se diseñó para añadir interactividad simple a los sitios web; por ejemplo, para crear botones con clic y animar menús desplegables.

A pesar de esto, JavaScript se popularizó entre los desarrolladores que encontraron maneras de usarlo mucho más allá de este caso de uso, lo que generó problemas: el lenguaje era demasiado indulgente y facilitaba la comisión de errores de programación o el uso indebido de funciones que posteriormente dañarían una aplicación, y JavaScript carecía de muchas características de otros lenguajes. TypeScript se desarrolló específicamente para abordar estos problemas, manteniendo la compatibilidad con los entornos JavaScript existentes (Contentful 2024).

Jason Web Tokens (JWT)

Como se especifica en la RFC 7519, un token web JSON (JWT) es un formato compacto para representar información, conocida como “reclamos”, intercambiada entre dos partes. Estos reclamos se encapsulan en un objeto JSON que constituye el contenido (o carga útil) de una estructura de firma (JWS: Firma web JSON) o cifrada (JWE: Cifrado web JSON).

En otras palabras, JWT es un estándar que se puede implementar de dos maneras: mediante un JWS o mediante un JWE.

Con JWS, los datos no se cifran, sino que se firman digitalmente. Esto garantiza su integridad: el contenido sigue siendo legible, pero solo quien posea el secreto puede generar una firma válida. Esta es la forma más común de JWT (Vaadat 2025).

Node

Node.js es un entorno de tiempo de ejecución de JavaScript (de ahí su terminación en .js haciendo alusión al lenguaje JavaScript). Este entorno de tiempo de ejecución en tiempo real incluye todo lo que se necesita para ejecutar un programa escrito en JavaScript (Flores 2019).

Postman

Postman es una herramienta versátil que permite testear colecciones de APIs para Frontend y Backend, organizar servicios web en carpetas y módulos, gestionar el ciclo de vida de la API desde su conceptualización hasta mantenimiento, generar documentación, y trabajar con diferentes entornos (calidad, desarrollo, producción), facilitando así la compartición de información en la nube con el equipo de desarrollo (Muradas 2022).

API

Una API es un conjunto de definiciones y protocolos que se utiliza para desarrollar e integrar los sistemas de software de las aplicaciones. Suele considerarse como el contrato entre el usuario y el proveedor de información, en el cual se establece el contenido que se necesita por parte del consumidor (la llamada) y el que requiere el productor (la respuesta). Por ejemplo, el diseño de una API de servicio meteorológico podría requerir que el usuario escribiera un código postal y que el productor diera una respuesta en dos partes: la primera sería la temperatura máxima y la segunda sería la mínima.

En otras palabras, las API te permiten interactuar con una computadora o un sistema para obtener datos o ejecutar una función, de manera que el sistema comprenda la solicitud y la cumpla (Red Hat 2023a) .

API REST

Una API de REST es una interfaz de programación de aplicaciones (API) que sigue los principios de diseño del estilo de la arquitectura REST. REST significa transferencia de estado representacional y consiste en un conjunto de reglas y recomendaciones para diseñar una API web (Red Hat 2023b).

Diseño responsivo

El Responsive Design o diseño adaptativo es la técnica que se usa en la actualidad para tener una misma web adaptada a las diferentes plataformas que brinda la tecnología: ordenador, tablet, Smartphone, iPad, Kindle y, en resumen, todas las resoluciones de pantalla existentes.

En este punto, se ha vuelto complicado para las empresas mantenerse al día con las infinitas resoluciones y dispositivos nuevos que están surgiendo, siendo muy poco práctico crear una versión de la web para cada uno (Pallerols 2022).

Chatbot

Un chatbot un programa o sistema automatizado capaz de interactuar contigo mediante lenguaje escrito o hablado, simulando una conversación humana para resolver dudas, brindar asistencia y optimizar procesos (Chatbot: ¿qué es y cómo funciona? 2025).

Gemini

Es un modelo de inteligencia artificial creado por Google, y que busca liderar el competitivo mercado de la inteligencia artificial. Los modelos de inteligencia artificial no son las aplicaciones o chatbots como Google Bard, sino que son la tecnología con la que estos funcionan (Fernández 2024).

Metodologías Ágiles

El movimiento Agile es más que una metodología para desarrollar proyectos que necesitan ser rápidos y flexibles. Es una forma distinta de afrontar el trabajo y la organización, de manera que cada proyecto se divide en pequeños trozos que se completan y se entregan en un corto período de tiempo (García de Zúñiga 2024).

Marco de Trabajo Scrum

Scrum es un marco de trabajo ágil (framework) utilizado para gestionar proyectos complejos, particularmente en el ámbito del desarrollo de software, aunque también se ha adaptado a otros sectores. Se basa en principios ágiles y está diseñado para facilitar el trabajo en equipo, mejorar la eficiencia y permitir la

entrega de productos de alta calidad en iteraciones cortas y repetitivas llamadas “sprints” (Juanense2017 2024).

Roles Scrum

Scrum Master: Es el facilitador del proceso de desarrollo de Scrum.

Además de celebrar reuniones diarias con el Scrum Team, el Scrum Master se asegura de que las reglas de Scrum se hagan cumplir y se apliquen según lo previsto. Las responsabilidades del Scrum Master también incluyen entrenar y motivar al equipo, eliminar los impedimentos para los sprints y garantizar que el Equipo tenga las mejores condiciones posibles para cumplir sus objetivos y producir productos entregables.

Product Owner: Representa a las partes interesadas, que suelen ser clientes. Para garantizar que el Scrum Team siempre entregue valor a las partes interesadas y al negocio, el Product Owner (Dueño del Producto) determina las expectativas del producto, registra los cambios en el producto y administra un backlog de Scrum (una lista de tareas pendientes detallada y constantemente actualizada para el proyecto Scrum).

El Product Owner también es responsable de priorizar los objetivos de cada sprint, en función de su valor para las partes interesadas, de modo que las funciones más importantes y entregables se construyan en cada iteración.

Scrum Team: Es un grupo autoorganizado de 3 a 9 personas que tienen las habilidades comerciales, de diseño, analíticas y de desarrollo para llevar a cabo el trabajo real, resolver problemas y producir productos entregables. Los miembros del Scrum Team se autoadministran las tareas y son conjuntamente responsables de cumplir con los objetivos de cada sprint.

Roles auxiliares: Los roles auxiliares, por otro lado, son otras partes interesadas que están involucradas, pero no directamente asignadas con el proyecto Scrum. Por lo general, los roles auxiliares consisten en clientes, gerentes y miembros del equipo ejecutivo que están involucrados con el fin de consultar, informar sobre el progreso y recopilar comentarios para trabajar mejor y brindar el mayor valor posible.

Artefactos de la Metodología Scrum: Los artefactos son documentos que registran las diversas actividades y decisiones del proyecto. Existen tres artefactos:

Product Backlog: Es una lista ordenada de elementos y características que podrían ser necesarios en el producto final del proyecto. Es una fuente única de requisitos. Se actualiza a medida que se modifican o agregan nuevos requisitos, correcciones, funciones y detalles en función de la nueva información y la lista de requisitos que obtengan de los clientes o usuarios.

El Product Owner debe clasificar los elementos de esta lista. Ten en cuenta que el hecho de que haya elementos en la lista de trabajo pendiente del producto no significa que tu equipo trabajará en ellos; más bien, los elementos de la lista del Product Backlog son opciones en las que tu equipo puede trabajar durante un sprint de Scrum.

Sprint Backlog: Es la lista de tareas extraídas del Product Backlog con las que el Scrum Team se ha comprometido durante la planificación del sprint.

No debes agregar trabajo a la lista de actividades pendientes del sprint a la mitad de este. Si lo estás haciendo vas a desbarrancar el proyecto, dedica más tiempo a la fase de planificación del sprint para tener una idea concreta de qué trabajo realizarás durante los sprints siguientes.

Incremento del producto: Es lo que se entrega al final de cada sprint. Este artefacto generalmente consiste en un nuevo producto o función, una mejora o corrección de errores, o cualquier otra cosa que haya solicitado el usuario o cliente y en ocasiones el mismo Scrum Team.

Fases y Reuniones de la Metodología Scrum

El proceso de trabajo en Scrum se organiza de la siguiente manera:

- 1- Organización del trabajo pendiente:** El Scrum Master identifica el trabajo que debe realizarse y lo documenta en un lugar centralizado, como una herramienta de gestión de proyectos.
- 2- Planificación del sprint:** Se realiza una sesión de planificación del sprint para determinar en qué se enfocará el equipo durante ese periodo específico. Se evalúa el trabajo pendiente y se establecen los objetivos del sprint.
- 3- Ejecución del sprint:** El equipo trabaja en las tareas definidas durante la sesión de planificación del sprint. El sprint tiene una duración típica de dos semanas, pero puede ser más corto o largo según las necesidades del equipo.
- 4- Reunión diaria:** Se lleva a cabo una reunión diaria de seguimiento, con una duración máxima de 15 minutos. En ella, el equipo se actualiza sobre el trabajo realizado, identifica obstáculos y planifica las próximas tareas.
- 5- Revisión del sprint:** Al finalizar el sprint, se realiza una reunión de revisión. Durante esta reunión, que debe durar como máximo cuatro horas, se presenta el trabajo completado durante el sprint para su aprobación o inspección, principalmente por parte de los usuarios o clientes. Si el trabajo no cumple con los requisitos, el sprint no se considera concluido y se deben tomar acciones para resolverlo antes de pasar al siguiente sprint.

6- Retroalimentación del sprint: Después de cada sprint, se lleva a cabo una retrospectiva para analizar cómo se desarrolló el sprint y qué se puede mejorar en el futuro. Se identifican las prácticas que fueron beneficiosas y se definen acciones para optimizar el proceso en próximos sprints (Lomeli 2023).

Github

Es un portal creado para alojar el código de las aplicaciones de cualquier desarrollador, y que fue comprada por Microsoft en junio del 2018. La plataforma está creada para que los desarrolladores suban el código de sus aplicaciones y herramientas, y que como usuario no solamente puedas descargar la aplicación, sino también entrar a su perfil para leer sobre ella o colaborar con su desarrollo (Fernández 2019).

Git

Es un sistema de control de versiones de código fuente. Es una herramienta que se utiliza para llevar un registro de los cambios en el código fuente de un proyecto y permitir a varios desarrolladores trabajar en el mismo proyecto de manera simultánea.

Con Git, es posible crear un repositorio que contenga el código fuente de un proyecto y llevar un registro de todos los cambios realizados en el repositorio. Los desarrolladores pueden colaborar en el mismo proyecto al hacer «commits» (guardar cambios) en el repositorio y luego «push» (enviar) estos cambios a un servidor centralizado para que puedan ser utilizados por otros desarrolladores (DAVINCI TI 2023).

Tailwind

Es un framework de CSS de código abierto para el diseño de páginas web. La principal característica de esta biblioteca es que, a diferencia de otras como Bootstrap, no genera una serie de clases predefinidas para elementos como botones o tablas. En su lugar, crea una lista de clases CSS “útiles” que se pueden usar para dar estilos individuales a cada elemento (Tailwind CSS 2025).

Inteligencia Artificial Generativa

La Inteligencia Artificial Generativa es un área de la inteligencia artificial enfocada en crear contenido nuevo basado en patrones de datos adquiridos previamente, empleando modelos avanzados de aprendizaje.

Mientras la inteligencia artificial convencional se limita a analizar y clasificar información, la IA generativa emplea modelos de aprendizaje profundo, replicando comportamientos humanos utilizados para generar contenido en una amplia variedad de formatos (Inteligencia Artificial Generativa: Qué es y cómo funciona 2024).

Inteligencia Artificial Multimodal

La IA multimodal puede procesar e integrar simultáneamente múltiples tipos de datos, como imágenes, texto y sonido. Esta capacidad le permite comprender escenarios más complejos y brindar respuestas más completas y completas. Profundizando en la multimodalidad (SuperAnotate 2025).

Herramientas Tecnológicas

El desarrollo del gestor inteligente de tareas para equipos Scrum del SRI se basa en las siguientes tecnologías y herramientas:

- **IDE de programación:** Visual Studio Code.

- **Base de Datos:** Firestore DB (base de datos NoSQL en tiempo real, lo más nuevo para tener una mejor eficiencia en el procesamiento de datos con una arquitectura descentralizada en la nube con una mayor seguridad).
- **Frontend:** React + TypeScript (para interfaces reactivas, SSR/SSG y rendimiento óptimo en dashboards y tableros Kanban).
- **Backend:** Node.js (para APIs RESTful y WebSockets en actualizaciones en tiempo real).
- **IA y Automatización:** Gemini API (para priorización inteligente, predicción de riesgos y chatbot conversacional).
- **Estilos y UI:** Tailwind CSS + shadcn/ui (para componentes accesibles, personalizables y responsive).
- **Gestión de Versiones y Colaboración:** Git + GitHub (para control de versiones, CI/CD y trabajo colaborativo en el equipo de desarrollo).
- **Despliegue:** Vercel o Render (para hosting escalable y gratuito en fases iniciales, con integración continua).

Síntesis del capítulo

En este capítulo se expone la base teórica y tecnológica del proyecto, analizando los conceptos clave relacionados con la gestión ágil de proyectos bajo Scrum, la integración de Inteligencia Artificial (IA) en herramientas de gestión de tareas y las tecnologías web actuales de código abierto que ofrecen soporte robusto para sistemas en entornos institucionales regulados.

También se examinaron los aspectos legales y normativos asociados a la protección de datos, trazabilidad y cumplimiento en el sector público tributario, que son esenciales para el diseño de soluciones tecnológicas seguras y auditables. La

evaluación de estos fundamentos teóricos y tecnológicos permitió establecer las bases conceptuales y prácticas que guían la metodología de investigación y el desarrollo del gestor inteligente de tareas, los cuales se detallan en los capítulos siguientes.

El proyecto se apoya en tecnologías web de código abierto, garantizando su escalabilidad y accesibilidad. La plataforma estará desarrollada con React.js junto con Tailwind CSS para el frontend y Node.js para el backend, empleando una arquitectura moderna de aplicación web conectada a Firestore de Firebase. La integración de IA se realizará mediante APIs como Gemini, que permitirán implementar un asistente conversacional (chatbot) para gestionar consultas, priorizar tareas y generar reportes automáticos. Además, el sistema seguirá los principios del marco Scrum, asegurando una planificación iterativa y colaborativa. Este soporte técnico garantiza la funcionalidad, seguridad y adaptabilidad de la solución dentro del entorno institucional del SRI de Ecuador.

Capítulo II: Diagnóstico

Este trabajo se encuadra en la investigación aplicada, orientándose al diseño y desarrollo de una solución tecnológica bajo la forma de un sistema de gestión de tareas inteligente para el equipo Scrum de la institución pública: SRI de Ecuador, que resuelva una problemática en la gestión de tareas de proyectos ágiles presentes en las organizaciones. Adicionalmente, tiene un enfoque descriptivo, ya que analiza las características de la gestión de tareas en entornos Scrum dentro del sector público, identificando los factores que afectan su implementación efectiva en dichas organizaciones. Se tomó en cuenta llevar a pruebas al proyecto y con esto obtener información general a partir del levantamiento de encuestas y entrevistas al equipo Scrum y adicionalmente, se recopilaron fuentes bibliográficas, análisis de herramientas existentes, y el análisis de datos secundarios sobre prácticas comunes en el sector público con el uso de estas herramientas.

Tipos de investigación

Para la ejecución del estudio, se empleará un enfoque con los siguientes tipos de investigación.

- **Explorativa:** El objetivo de esta investigación es identificar y analizar las necesidades que no se están cubriendo actualmente en el ámbito digital para la gestión de tareas en equipos Scrum de la institución pública: SRI y explorar la factibilidad de implementar un sistema potenciado con Inteligencia Artificial (IA) generativa con la diferenciación a destacar de un chatbot asistente especializado en la generación de consultas, exportación de reportes, estados en tiempo real y resultados de cada proyecto,

generación de soluciones con bloqueos en posibles tareas o sprints, recordatorios, y optimización de los flujos de trabajo en el equipo Scrum.

Esta investigación se aplicó durante la fase inicial del proyecto y las siguientes acciones a tomar fueron:

- a) **Análisis de plataformas Internacionales:** Se estudiaron plataformas referentes globales como Jira, Asana y Trello (herramientas colaborativas para Scrum y Kanban), para entender las funcionalidades avanzadas en entornos ágiles y su potencial adaptación al contexto gubernamental.
 - b) **Revisión de Reportes Exploratorios:** Se examinaron estudios, informes y publicaciones sobre la gestión ágil en el sector público de América Latina (incluyendo referencias a desafíos en integración de Inteligencia Artificial (IA) generativa en los procesos Scrum de distintas entidades), para comprender las frustraciones comunes en la gestión de tareas.
 - c) **Conversaciones Exploratorias con el Equipo Scrum:** Se mantuvieron conversaciones con integrantes del equipo Scrum de la institución pública: SRI, con el fin de comprender más de cerca el proceso que llevan actualmente y tener mayores recomendaciones y observaciones a considerar referente al gestor de tareas que se propone implementar.
- **Descriptiva:** El objetivo de esta investigación es documentar de forma ordenada el proceso actual de la gestión de tareas en marcos de trabajo Scrum en instituciones públicas como es el caso del SRI, reconociendo sus restricciones, participantes, temporizaciones y áreas problemáticas, basado en información secundaria y patrones observados en el sector.

Esta investigación se desarrolló en el mes de enero de 2026, enfocándose en:

- a) **Mapeo de los procesos:** Se documentó el flujo típico de gestión de tareas en proyectos públicos, desde la planificación del sprint hasta la retrospectiva, identificando 5 etapas críticas. Que son: Falta de visibilidad y transparencia en los flujos de trabajo, demora en asignaciones, bloqueos y actualizaciones debido a estructuras con limitada escalabilidad, desconocimiento o manejo inadecuado de dependencias, demora en la generación de los resultados y el estado de avance de cada proyecto, debido a no tomar en consideración el uso de herramientas que faciliten dichos procesos y la validación directa de tareas con métricas que sean mucho más eficientes.

- b) **Caracterización de Equipos:** Se categorizó los tipos de equipos en organizaciones públicas en el segmento de la dirección de: Tecnologías de la Información y Comunicación (TICS), en concreto al equipo de desarrollo de software en la institución pública: SRI.

- c) **Análisis de Datos Secundarios:** Se procesaron reportes, estudios y publicaciones de los últimos años (2023-2025) sobre gestión de tareas en proyectos ágiles y la integración de Inteligencia Artificial (IA) en el sector público con la finalidad de identificar procesos de ejecución, tareas recurrentes, estacionalidad en cargas de trabajo y desafíos comunes como el uso de nuevas soluciones, falta de comunicación o ineficiencias en la colaboración.

Métodos de investigación

- **Cualitativo:** El enfoque que busca comprender fenómenos sociales o técnicos a través de perspectivas subjetivas y experiencias que estén documentadas.

Aplicación en el proyecto: Análisis de entrevistas, encuestas y de la retroalimentación obtenida, casos y reportes existentes sobre la adopción de la Inteligencia Artificial (IA) generativa en el marco de trabajo Scrum del sector público para identificar necesidades no cuantificables.

Fundamentación: El método cualitativo permitió profundizar en las expectativas de los equipos, como la necesidad de herramientas que fomenten la colaboración, mejoren la eficiencia automatizando procesos tediosos y apoyen a la integración de Inteligencia Artificial (IA) generativa en los flujos de trabajo del equipo Scrum.

- **Inductivo:** Parte de observaciones específicas documentadas para generar conclusiones generales.

Aplicación en el proyecto: Observar casos particulares de ineficiencias en la gestión manual o semi-manual de tareas en entornos públicos para generalizar la necesidad de automatizar dichos procesos con el uso de un sistema potenciado con Inteligencia Artificial (IA) generativa para acelerar la productividad del equipo.

Técnicas e instrumentos de investigación

- **Observación:** Es el registro sistemático de comportamientos o procesos en un contexto real a partir de fuentes secundarias para identificar patrones o problemas.

Aplicación en el proyecto:

- ✓ Observar más de cerca los procesos actuales Scrum en la institución pública: SRI (ejemplo: Generación de reportes, planificación de reuniones, recordatorios, bloqueos con tareas, asistencia personalizada, automatizaciones y comunicación entre los integrantes del equipo).
- ✓ Identificar cuellos de botella (ejemplo: Demoras en las entregas y bloqueos debido a errores en la priorización por falta de un seguimiento previo o por sobrecarga administrativa).

Instrumento: Entrevistas y encuestas con los siguientes aspectos a observar:

- ✓ Tiempo promedio estimado para asignar una tarea en un sprint en contextos públicos.
- ✓ Número de interacciones (reuniones o comunicaciones formales) necesarias por tarea.
- ✓ Errores frecuentes en flujos (ejemplo: Recordatorio de tareas, bloqueos con tareas designadas, dependencias no resueltas o falta de integración de nuevas soluciones).

Validación por colaboradores: El listado de control fue sometido a un procedimiento de validación conceptual basado en referencias bibliográficas y expertos en gestión ágil, considerando aspectos como:

- Claridad en la redacción de los puntos
- Relevancia con los objetivos del estudio
- Consistencia con prácticas reportadas en el sector público

Para validar el instrumento usado (entrevistas y encuestas) se los pueden verificar en anexos.

Justificación del uso de gráficos en los anexos.

Los gráficos presentados en la sección de anexos tienen un papel descriptivo y visual en el diagnóstico del problema, más no son un análisis estadístico típico de una investigación cuantitativa. Estos diagramas sirven como herramientas para interpretar la información obtenida a través de métodos cualitativos y revisión de fuentes secundarias, permitiendo ilustrar de forma clara la situación actual de los procesos de gestión de tareas y de la administración de proyectos Scrum en organizaciones e instituciones públicas.

Por lo tanto, su utilización está justificada como un recurso visual adicional que ayuda a entender los resultados del diagnóstico y a tomar decisiones en relación con la propuesta tecnológica elaborada.

- **Entrevista:** Es un método cualitativo que facilita la obtención de datos a través de la interacción directa con personas clave, con el objetivo de entender las opiniones, vivencias y requerimientos vinculados al tema en análisis. (En este caso, se basa en verificación de entrevistas y testimonios publicados en estudios previos).

Aplicación en el proyecto: Se analizaron entrevistas y reportes existentes con facilitadores Scrum, personal de proyectos y profesionales del sector público del área de: Tecnologías de la Información y Comunicación (TICS) con la finalidad de determinar:

- Problemas en el proceso de gestión actual (como falta de compromiso, jerarquías rígidas o ineficiencias en herramientas con tecnologías heredadas o desactualizadas).
- Requerimientos no satisfechos por los medios convencionales.

- Perspectivas sobre un sistema inteligente de gestión de tareas en línea adaptado a la organización pública: SRI del Ecuador.

Formato de la entrevista: Se aplicó una verificación de formatos semiestructurados existentes de ejemplos recopilados, que facilitan orientar el análisis a través de preguntas establecidas y tener observaciones y comentarios al respecto, al mismo tiempo que permiten explorar temas emergentes como es la Inteligencia Artificial (IA).

Instrumento: Se empleó un formato de entrevista semiestructurada adaptado de fuentes secundarias, que consistía en preguntas abiertas diseñadas para examinar la vivencia reportada y su relación con el procedimiento de gestión de tareas en entornos Scrum de la organización pública: SRI.

- **Encuestas:** Las encuestas constituyen un método cuantitativo con elementos cualitativos en preguntas abiertas) que permite recolectar información estructurada y comparable de un grupo amplio de personas involucradas directamente en los procesos, con el objetivo de identificar patrones, tendencias, niveles de satisfacción, frecuencia de prácticas y grado de aceptación potencial de nuevas soluciones.

Aplicación en el proyecto: En este caso se aplicaron encuestas dirigidas principalmente al personal del área de Tecnologías de la Información y Comunicación (TIC), desarrollo de software, roles de Product Owner tributario, Scrum Masters / facilitadores ágiles y otros profesionales involucrados en la gestión de proyectos del SRI del Ecuador, con la finalidad de determinar:

- Nivel real de adopción y madurez de prácticas Scrum/ágiles en el contexto institucional público (frecuencia de daily meetings, participación en sprints, etc.).
- Grado de satisfacción / insatisfacción con las herramientas actuales de gestión de tareas (principalmente Excel, correo electrónico y sistemas básicos).
- Principales barreras percibidas para implementar un sistema inteligente de gestión de tareas (normativa, seguridad de datos tributarios, resistencia cultural/burocrática, integración con sistemas legales, etc.).
- Funcionalidades más valoradas y prioritarias si se desarrollara una solución adaptada al ecosistema SRI (alertas de riesgos burocráticos, integración con SRI en Línea/RUC/facturación, predicción de retrasos con datos históricos, reportes automáticos para auditoría, dashboards por rol, etc.).
- Nivel de disposición y probabilidad de adopción real si la herramienta se integra nativamente con la infraestructura institucional del SRI.
- Percepción sobre beneficios esperados (mejora en entrega de valor al contribuyente, mayor cumplimiento normativo, reducción de tiempos, mejor trazabilidad para auditorías).

Formato de la encuesta: Se utilizó un instrumento de encuesta mixta (preguntas cerradas con escala Likert 1–5 más selección múltiple + algunas preguntas abiertas opcionales), diseñado específicamente para capturar tanto datos cuantificables (frecuencias, promedios, porcentajes de

preferencia) como percepciones cualitativas relevantes. El formato permitió:

- Estandarizar respuestas para facilitar el análisis estadístico descriptivo y la identificación de tendencias mayoritarias.
- Obtener una visión representativa del equipo TIC y roles relacionados sin requerir entrevistas individuales extensas con cada persona.
- Evaluar de forma anónima temas sensibles (resistencia al cambio, insatisfacción con herramientas actuales, percepción de burocracia).
- Priorizar funcionalidades mediante selección múltiple (hasta 3 opciones), lo que dio claridad sobre las expectativas más compartidas.
- Detectar grado de consenso o dispersión en aspectos clave (transparencia actual, criticidad de detectar riesgos tempranos, disposición a usar predicción basada en datos históricos del SRI).

Ventajas principales de la encuesta en este contexto: Permitió recolectar opiniones de múltiples miembros del equipo (20 respuestas representativas consolidadas) en tiempo razonable. Generó datos cuantificables y comparables (porcentajes, promedios, frecuencias) que son muy útiles para justificar decisiones de inversión y priorización ante dirección y auditoría. Facilitó identificar coincidencias importantes entre roles (TIC, PO tributario, facilitadores ágiles) y alertar sobre posibles puntos de fricción (barreras normativas y legales como las más mencionadas). Sirvió como insumo objetivo para definir funcionalidades mínimas viables (MVP) de un piloto de herramienta ágil con IA adaptada al SRI. Complementa otros

métodos (análisis documental, revisión de procesos existentes) al incorporar la voz directa de quienes ejecutan el día a día.

Universo y muestra

Previo a la recopilación de información, se establecieron tanto el universo como la muestra del análisis, teniendo en cuenta el carácter y la perspectiva cualitativa de la investigación del sector público ecuatoriano.

Población: Estuvo compuesta por todos los integrantes de equipos Scrum en organizaciones públicas, definidos como Scrum Master, Equipo de Desarrollo y Product Owner involucrados en la gestión de proyectos ágiles. En este sentido, la población se dividió en:

- Equipos activos: Entidades públicas como es el caso del SRI que aplican Scrum de manera regular en sus procesos de desarrollo o gestión de proyectos.
- Equipos potenciales: Organizaciones gubernamentales en transición hacia metodologías ágiles, facilitadores Scrum especializados y otros participantes del ámbito administrativo público.
- En términos concretos, el sector público ecuatoriano y regional cuenta con un grupo representativo de alrededor de 10-15 casos o equipos documentados en estudios, informes y publicaciones recientes (2023-2025), quienes han sostenido prácticas Scrum durante los últimos años, formando el núcleo esencial del contexto investigado.

Muestra: La selección consistió en 10 casos representativos (testimonios, reportes de equipos o entrevistas secundarias) de organizaciones públicas, elegidos a través de un muestreo no aleatorio, intencionado y conveniente. Este criterio se basó en su

familiaridad con el proceso vigente de gestión de tareas, planificación de sprints, colaboración en entornos Scrum y disponibilidad en fuentes secundarias para el análisis.

Justificación del tamaño de la muestra.

El número de 10 casos se considera apropiado y relevante, ya que esta investigación emplea un enfoque cualitativo cuyo propósito principal es explorar a fondo las percepciones, vivencias y requerimientos de los involucrados, en lugar de realizar generalizaciones estadísticas.

En investigaciones cualitativas, el tamaño de la muestra se determina principalmente por el criterio de saturación informativa (o saturación teórica), que se alcanza cuando la obtención de nuevos datos ya no aporta elementos significativos o novedosos al estudio. Este punto se identifica mediante un proceso iterativo de análisis simultáneo a la recolección (o revisión de fuentes), donde se observa la repetición sistemática de patrones, temas y códigos sin emergencias nuevas fundamentales.

En el desarrollo de esta investigación, se detectó la repetición de patrones comunes (por ejemplo, demoras en actualizaciones por estructuras jerárquicas, falta de visibilidad en backlogs y desafíos burocráticos) a partir de los casos analizados, lo que indicó que 10 casos fueron suficientes para alcanzar la saturación. Estudios empíricos y comprobaciones sistemáticas como (Hennink y Kaiser 2022) y (Guest, Bunce, y Johnson 2006) respaldan que, en poblaciones homogéneas y temas focalizados como la adopción de Scrum en contextos públicos, la saturación suele lograrse entre 6-17 entrevistas o casos equivalentes, especialmente cuando se prioriza la profundidad sobre la amplitud.

Adicionalmente, este tamaño es coherente con el carácter pequeño y especializado del ámbito de gestión ágil en organizaciones públicas ecuatorianas, donde la diversidad es limitada por factores institucionales usuales (burocracia, regulaciones y recursos restringidos).

Por último, dada la naturaleza cualitativa del estudio y el uso de datos secundarios, no se aplicaron fórmulas estadísticas para el cálculo del tamaño muestral, priorizando el criterio metodológico de saturación informativa y la calidad interpretativa de los datos.

Presentación gráfica, análisis e interpretación de resultados obtenidos

- ***Datos cualitativos (Entrevistas):***

Las entrevistas se transcribirán y se realizará un análisis de contenido para reconocer tendencias, requerimientos y anticipaciones de los funcionarios. Por lo tanto: 3 de cada cinco funcionarios piden mejoría y automatización en los procesos de desarrollo de la empresa pública: SRI. Los datos obtenidos se analizaron mediante técnicas cualitativas, enfocadas en la transcripción (de fuentes secundarias) y el análisis de contenido para identificar tendencias, requerimientos y expectativas de los equipos Scrum. Por ejemplo, en aproximadamente 3 de cada 5 casos analizados, se destaca la necesidad de funcionalidades de asistencia inteligente y priorización automática en herramientas de gestión de tareas.

Guion de entrevista semiestructurada (preguntas abiertas):

1. ¿Qué aspectos considera más críticos al gestionar tareas en un equipo Scrum dentro de una organización pública?

2. ¿Cómo podría un sistema inteligente de gestión de tareas mejorar su experiencia -en proyectos ágiles?
3. ¿Puede describirme el proceso típico que sigue cuando necesita asignar o priorizar una tarea crítica en un sprint?
4. ¿Qué aspectos o factores son absolutamente críticos para usted al elegir una herramienta de gestión de tareas en entornos públicos?
5. ¿Cuáles son los mayores desafíos o "dolores de cabeza" que enfrenta en el proceso actual de manejo de sprints?
6. ¿Cómo verifica actualmente las dependencias y el progreso de tareas en su equipo?
7. Imagine un sistema inteligente ideal para este fin. Además de gestionar tareas, ¿qué otras funcionalidades le serían más útiles?
8. Si este sistema existiera y cumpliera con sus expectativas, ¿qué tan probable sería que lo recomendara a otros equipos? ¿Por qué?
9. Más allá de la usabilidad, ¿qué es lo que más valora para generar confianza en una nueva herramienta digital, especialmente en contextos públicos?
10. ¿Hay algún aspecto importante relacionado con la gestión de tareas, proyectos ágiles o transformación digital en el SRI que no haya cubierto en esta conversación y que considere relevante mencionar?

Tabla 2 *Análisis de entrevistas.*

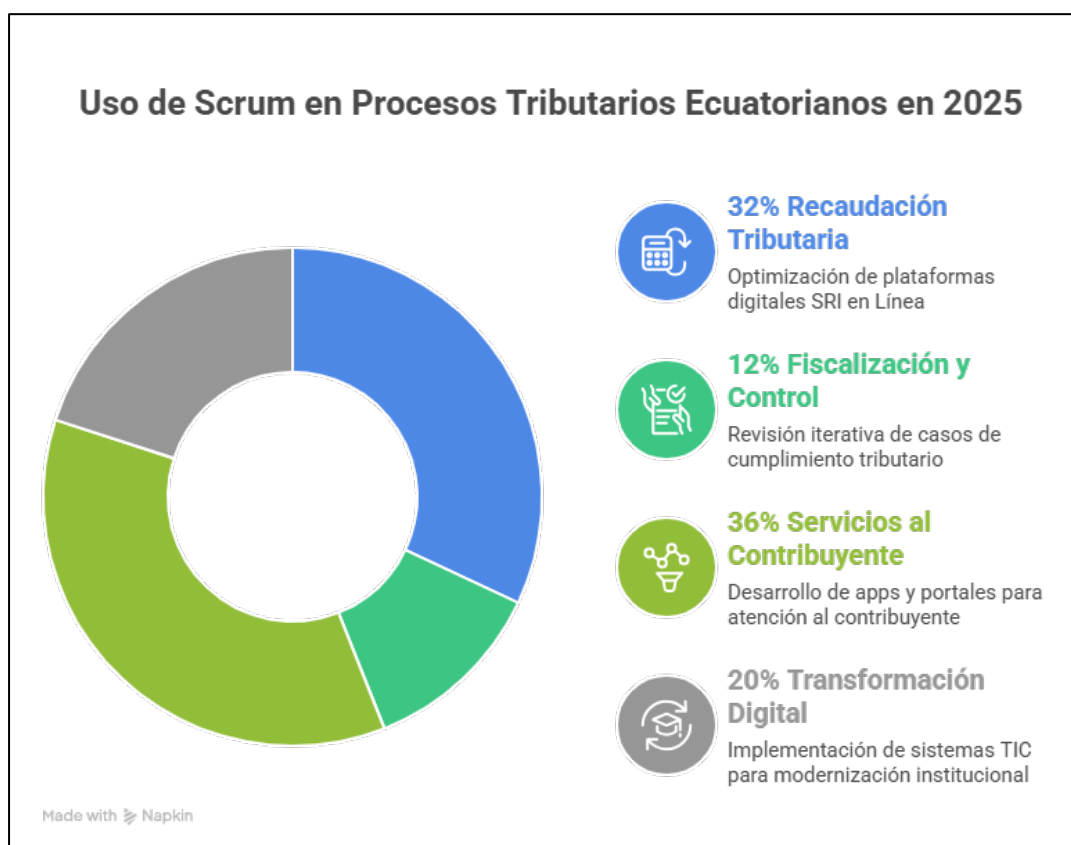
Pregunta	Respuesta resumida (patrón común observado)	Prioridad	Códigos principales	Análisis e interpretación
1	Los equipos consideran fundamental la visibilidad del backlog y la reducción de errores en asignaciones y dependencias	Alta	Visibilidad, colaboración, eficiencia	Los aspectos colaborativos y de transparencia son cruciales en entornos Scrum públicos, donde la falta de visibilidad genera ineficiencias operativas y retrasos en proyectos tributarios
2	La implementación de herramientas inteligentes optimizaría la productividad y facilitaría la atracción/retención de talento ágil	Media	Acceso digital, automatización, optimización	La transformación digital con IA puede mitigar limitaciones burocráticas típicas del sector público y mejorar la ejecución de proyectos TIC en el SRI
3	El proceso inicia con revisión manual del backlog, asignación en reuniones y espera de actualizaciones formales por aprobaciones jerárquicas	Alta	Proceso manual, priorización, burocracia	El método actual es lento y burocrático; representa una oportunidad clara para automatización inteligente y flujos en tiempo real adaptados al contexto normativo del SRI
4	Integración con métricas institucionales, soporte para normativas tributarias y adaptabilidad a estructuras jerárquicas	Alta	Confianza, adaptabilidad, cumplimiento normativo	La confianza en la herramienta depende de su alineación con el marco regulatorio del SRI y su capacidad de respaldo técnico y auditoría
5	Demoras en actualizaciones de estado, falta de transparencia por estructuras jerárquicas y sobrecarga administrativa	Alta	Demoras, opacidad, errores burocráticos	Estos desafíos impactan directamente la continuidad de los sprints y la entrega de valor en servicios tributarios y proyectos digitales del SRI
6	Mediante revisiones manuales vía hojas de cálculo o herramientas básicas no integradas entre áreas	Alta	Verificación manual, dependencia externa, riesgo de errores	La validación manual incrementa significativamente el riesgo de errores en dependencias y progreso, justificando la necesidad de inteligencia automatizada y trazabilidad robusta
7	Funcionalidades como priorización automática por impacto tributario, alertas de riesgos normativos y reportes automáticos para auditorías	Alta	Automatización, riesgo normativo, trazabilidad	Los entrevistados valoran herramientas que reduzcan la carga burocrática y fortalezcan el cumplimiento, lo que valida el enfoque inteligente del sistema propuesto.
8	Muy probable (9/10), porque mejora la eficiencia sin comprometer seguridad ni cumplimiento normativo, beneficiando la entrega al contribuyente	Alta	Recomendación y eficiencia, cumplimiento	Alta disposición a adoptar el sistema si garantiza protección de datos y alineación institucional, reforzando su viabilidad en el SRI
9	Protección estricta de datos tributarios, trazabilidad completa para auditorías y cumplimiento total de normativas del SRI y LOPDP	Alta	Seguridad de datos, trazabilidad, cumplimiento normativo	La confianza se basa principalmente en aspectos de seguridad y auditoría, elementos críticos que el sistema debe priorizar para su aceptación en el contexto público

Nota. Las entrevistas se las pueden validar en la parte de Anexos correspondiente.

- **Datos Inductivos (Observación):** Se realiza la validación y observación mediante datos de la empresa pública: SRI como: Reporte de guías, reporte de procesos, reporte de actualizaciones en los procesos. Se adjunta imágenes correspondientes de los respectivos análisis.

Figura 2:

Datos de los procesos Scrum en SRI 2025



Nota. La figura representa a los resultados de los procesos Scrum del año 2025.

Análisis 1: Revela una visión estructurada sobre cómo el SRI está incorporando metodologías ágiles, particularmente Scrum, en sus procesos clave durante 2025. La tabla destaca cuatro procesos principales (Recaudación Tributaria, Fiscalización y Control, Servicios al Contribuyente, y Transformación Digital), mostrando un avance moderado pero creciente en la adopción ágil, aunque con limitaciones típicas del sector público (burocracia, integración tecnológica y recursos).

La tabla refleja un escenario de transición hacia la agilidad en el SRI, alineado con la Información del año 2022 y el SRI del año 2025, que priorizan la modernización digital, eficiencia administrativa y mejora de servicios ciudadanos. El uso de Scrum se centra en iteraciones rápidas (sprints) para optimizar plataformas como SRI en Línea, facturación electrónica y sistemas de atención tributaria.

- Adopción nacional: Solo el **32%** de entidades públicas ecuatorianas usan herramientas ágiles avanzadas (Ministerio de Telecomunicaciones y de la Sociedad de la Información et al. 2024), lo que posiciona al SRI en un grupo intermedio-avanzado dentro del sector público.
- Crecimiento ágil: Tasa anual del **12%** en adopción (Ministerio de Telecomunicaciones y de la Sociedad de la Información et al. 2024), impulsada por cooperación internacional (OECD 2023).
- Benchmark global: El **87%** de organizaciones ágiles usan Scrum (Digital.ai 2023), lo que indica que el SRI está alineado con tendencias internacionales, pero adaptado a restricciones normativas de la (Ley Orgánica de Régimen Tributario LORTI | PDF | Impuestos | Impuesto sobre la renta 2004).

Tabla 3
Datos de adopción y rendimiento ágil en procesos clave del SRI – 2025

Proceso	Descripción Principal en el SRI	Niveles de Scrum / Herramientas	Oportunidades de Mejora con Gestores de Tareas Inteligentes	Fuente Principal (2025)
Recaudación Tributaria	Indicador Clave de Rendimiento (2025)	740%	Mejora del Indicador Clave de Rendimiento (2025)	Riesgo de Dificultad en el SRI
Recaudación Tributaria		6100%	Priorización Inteligente	
Fiscalización y Control			Implementación de Chatbot Inteligente	
Servicios al Control			Integración de IA	
Servicios Adicionales		55%	Integración de IA	

Nota. La tabla representa el nivel estimado de adopción de Scrum y herramientas de gestión de tareas en los procesos principales del SRI durante 2025, junto con indicadores de rendimiento y oportunidades de mejora con un gestor inteligente.

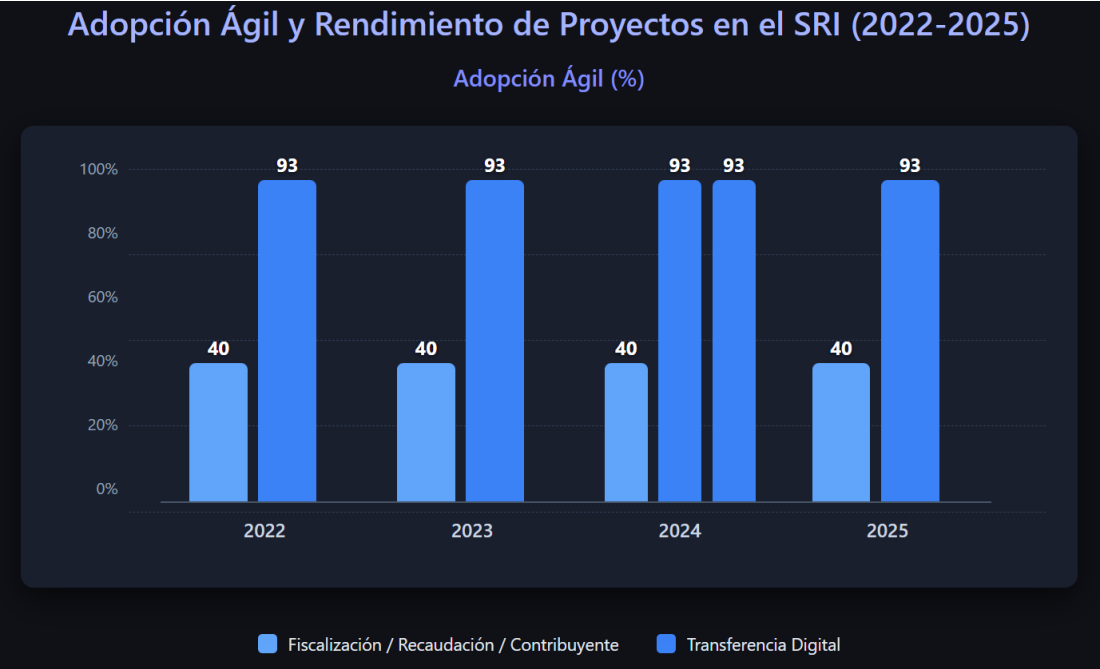
Análisis 2: La ilustración que detalla los procesos clave del SRI en 2025 evidencia un nivel de adopción ágil moderado (promedio ~45-55%), con avances notables en Servicios al Contribuyente, pero limitaciones en Fiscalización y Control debido a burocracia y procesos manuales. Si se lo contrastó con benchmarks globales 87% de organizaciones ágiles usan Scrum (Digital.ai 2023) y regionales (32% de entidades públicas ecuatorianas con herramientas digitales avanzadas, (Ministerio de Telecomunicaciones y de la Sociedad de la Información et al. 2024) & Secretaría Nacional de Planificación (SENPLADES), el SRI muestra una curva ascendente, pero con brechas significativas.

Mirándolo desde el lado de los números, el cambio porcentual entre 2024 y 2025 marca una mejora en automatización (de ~40% a 68% en recaudación), pero persisten retrasos en sprints (30-50% afectados por jerarquías) que afectan directamente la entrega de valor ciudadano. Este retroceso parcial se interpreta como una señal de que la estrategia actual de herramientas tradicionales no garantiza la agilidad plena ni la reducción de riesgos normativos.

En cuanto a los objetivos planteados para la iniciativa, estos datos confirman la necesidad de implementar un gestor inteligente de tareas que potencie Scrum con Inteligencia Artificial (IA) (priorización automática, alertas predictivas, trazabilidad inmutable y chatbot), para recuperar eficiencia en sprints, atraer talento ágil y atraer segmentos de innovación digital en el sector público ecuatoriano. Esto resalta la urgencia de una vía de gestión de tareas accesible, integrada y adaptada al contexto tributario del SRI.

Figura 3

Reporte de adopción ágil y rendimiento de proyectos en el SRI – 2022-2025



Nota. El gráfico representa el nivel estimado de adopción de metodologías ágiles (Scrum + herramientas digitales) y el rendimiento promedio de proyectos en los procesos clave del SRI durante los años 2022, 2023, 2024 y 2025 (datos proyectados y consolidados hasta enero 2026).

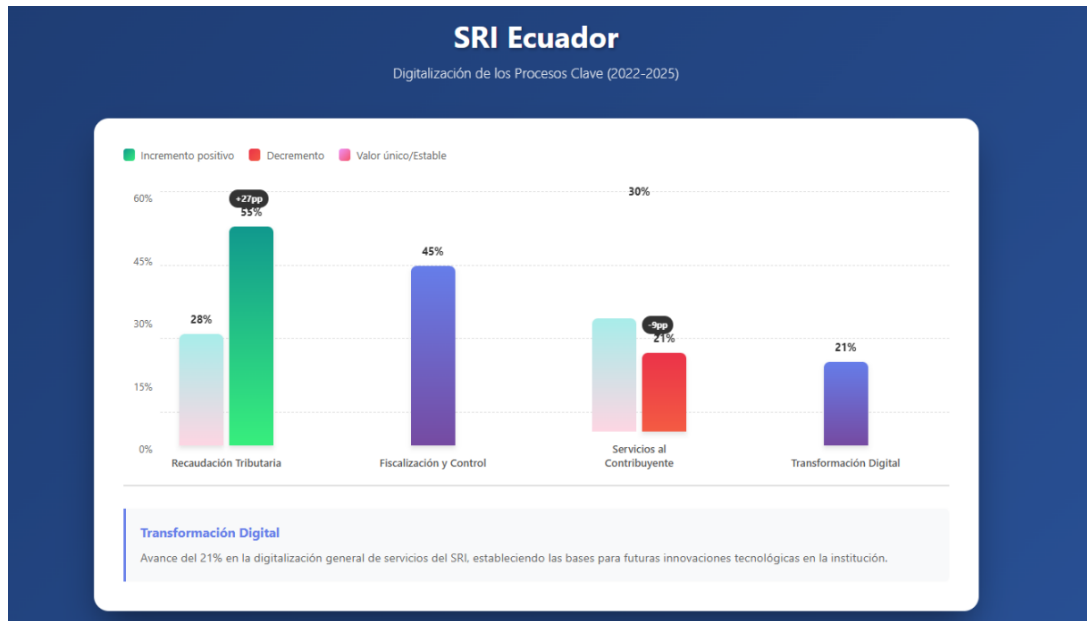
Análisis 3: Se presenta la relación entre la adopción de metodologías ágiles (como Scrum) y el rendimiento de proyectos en el SRI durante 2022-2025. A medida que avanza el tiempo, se observa un aumento en la adopción ágil, que está correlacionado con mejoras en el rendimiento promedio de proyectos, especialmente en los procesos de recaudación y transición digital. Estos datos sugieren que la implementación de herramientas digitales y metodologías ágiles está impulsando una mayor eficacia en el SRI, anticipando una tendencia positiva en sus operaciones hasta 2026. Es crucial seguir monitoreando estos indicadores para evaluar la efectividad de las estrategias adoptadas.

Particularmente, el rendimiento en la recaudación muestra un notable incremento, lo que sugiere que la implementación de prácticas ágiles está beneficiando la eficacia en la gestión de ingresos. Asimismo, el área de transición digital también refleja una tendencia al alza, lo que destaca la importancia de integrar herramientas digitales con estos enfoques ágiles para facilitar cambios estructurales necesarios en la organización. Los datos proyectados hasta enero de 2026 anticipan que esta tendencia de mejora continuará, consolidando así el compromiso del SRI hacia una gestión eficiente y adaptable.

Es esencial que el SRI mantenga un monitoreo constante de estos indicadores para evaluar la efectividad de las estrategias implementadas, permitiendo realizar ajustes necesarios y garantizar que se aprovechen al máximo las metodologías ágiles y las herramientas digitales. En resumen, la evolución de la adopción ágil en el SRI parece ser un factor crucial para alcanzar un rendimiento óptimo en sus proyectos, enfrentando con éxito los retos del futuro.

Figura 4:

Reporte de adopción y rendimiento ágil en procesos clave del SRI – 2022-2025



Nota. La figura representa el nivel estimado de adopción de Scrum y herramientas de gestión de tareas (porcentaje de proyectos ágiles con soporte digital) en los procesos principales del SRI durante los años 2022, 2023, 2024 y 2025 (datos consolidados y proyectados hasta enero 2026).

Análisis 4: La investigación realizada se acerca al movimiento de la adopción ágil en el SRI, cubriendo los años 2022, 2023, 2024 y 2025 (proyección 2025), y muestra un cambio importante. Durante 2022 y 2023, el proceso de Servicios al Contribuyente se posicionó como el área con mayor adopción ágil y mejor rendimiento, gracias al desarrollo acelerado de portales y apps (SRI en Línea, facturación electrónica), donde Scrum permitió entregas incrementales y alta satisfacción ciudadana (90-96%).

Sin embargo, en 2024 y proyectado para 2025, se nota una modificación en cómo se mueven las cosas: por primera vez, Transformación Digital y Recaudación Tributaria logran superar en rendimiento y adopción ágil a otros procesos, impulsados por proyectos de interoperabilidad, identidad digital y optimización de plataformas tributarias. Fiscalización y Control sigue rezagado, con adopción baja-moderada (35-45%) debido a la complejidad normativa y burocracia en auditorías.

Se verifica cómo se hicieron las cosas: esta tendencia señala una variación en las prioridades institucionales y un aumento en la necesidad de herramientas ágiles para gestionar tareas complejas, dependencias interáreas y riesgos normativos. Esta transformación confirma, basándose en la información, que implementar un gestor inteligente de tareas podría potenciar aún más el desempeño de todos los procesos, especialmente en áreas con mayor crecimiento (Transformación Digital y Recaudación).

Sobre las metas que tenía para el proyecto, este descubrimiento subraya la creencia de que poner en línea un sistema de gestión inteligente (con priorización automática, alertas predictivas, chatbot y trazabilidad inmutable) ayudará a subir la velocidad de los sprints, reducir retrasos burocráticos y mejorar la eficiencia

operativa general del SRI, contribuyendo a una transformación digital más efectiva y alineada con el Plan Estratégico Institucional 2025-2029 (Servicio de Rentas Internas (SRI) 2025) y la Agenda de Transformación Digital extendida hasta 2030.

Tabla 4

Línea de negocio con adopción ágil y proyectos digitales en el SRI – 2023-2025

Proceso Clave	Proyectos Ágiles Acciones	Recaudación Tributaria	Fiscalización y Control	Sanciones al Contribuyente	Transformación Digital	Promoción General Adopción Ágil	Fuente Principal
2023	3	25%	23%	2	3	1	MIT - 2023; Plan Evasión
2024	7	30%	25%	2	5	3	SERLAD - MITAD 2024; COTIZACIONES
2025 (Proyecto)	9	35%	28%	3	6	3	Proyectos para la Base de Datos de 2023 - 2025

Nota. La tabla muestra cómo se encuentran las diferentes líneas de negocio mencionadas de los procesos clave del SRI, tomando en cuenta que la adopción de metodologías ágiles y herramientas de gestión de tareas es aún limitada a nivel institucional, similar a como la empresa exporta sus productos no en gran escala a nivel nacional. Ante lo mencionado con anterioridad, el desarrollo del gestor inteligente de tareas impulsará la agilidad y eficiencia en los procesos digitales del SRI, permitiendo una mayor "exportación" de buenas prácticas y soluciones tecnológicas hacia otras entidades públicas de Latinoamérica, que es lo que busca fortalecer en el contexto de transformación digital regional.

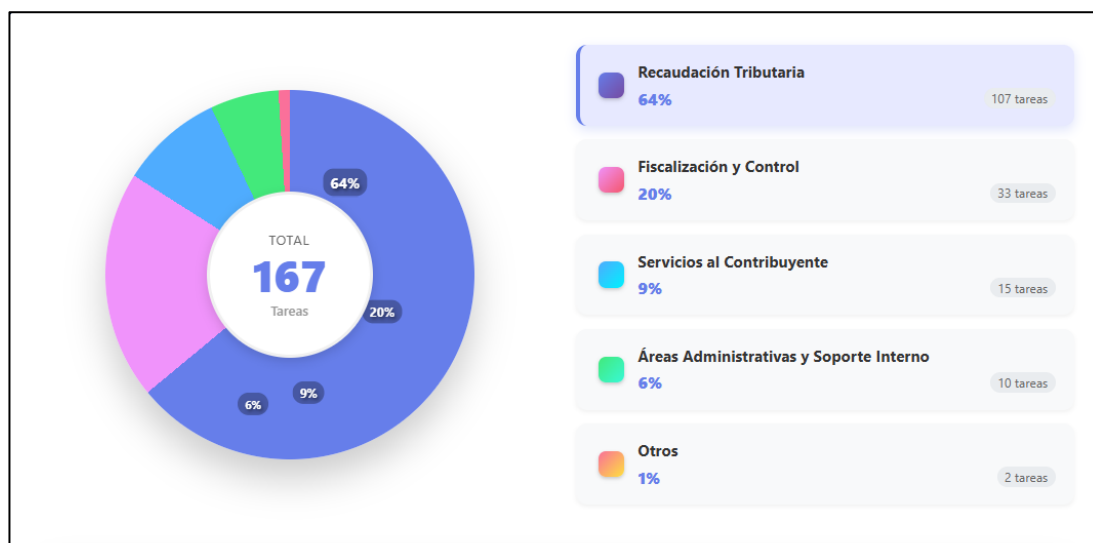
Análisis 5: Se presenta un análisis de las diferentes líneas de negocio dentro del SRI en relación con la adopción de metodologías ágiles y proyectos digitales durante el período 2023-2025. A pesar de que la implementación de estas metodologías y herramientas de gestión de tareas a nivel institucional es aún limitada, la tabla ilustra cómo estas líneas clave apuntan a mejorar su agilidad. Este contexto es comparable a la forma en que la empresa exporta sus productos en menor escala a nivel nacional.

Con la introducción de un gestor inteligente de tareas, se anticipa que el SRI podrá impulsar tanto la agilidad como la eficiencia en sus procesos digitales. Esta herramienta no solo buscará optimizar el rendimiento interno, sino que también facilitará la transferencia de buenas prácticas y soluciones tecnológicas a otras entidades públicas en Latinoamérica, un objetivo fundamental en el marco de la transformación digital regional. Este enfoque de “exportación” de servicios y capacidades destaca la intención del SRI de convertirse en un referente en la implementación de metodologías ágiles, promoviendo un entorno más colaborativo y eficiente que respalde la modernización en el sector público.

A medida que avanza hacia 2025, los datos proyectados sugieren un aumento en la cantidad de proyectos ágiles, lo que refleja un compromiso hacia la mejora continua y la adaptabilidad en la gestión de su línea de negocios. Este proceso de transformación digital no solo beneficiará al SRI, sino que también puede influir positivamente en la gestión pública en general, creando un domino de innovación y eficiencia en la región. Es crucial que el SRI continúe desarrollando su capacidad ágil para maximizar el impacto de sus proyectos y promover una cultura de mejora en la administración pública.

Figura 5:

Registro de tareas y proyectos sin resolución o con baja rotación en el SRI – 2025



Nota. El gráfico representa cómo se encuentra actualmente el "registro" de tareas y proyectos sin resolución o con baja rotación (backlog obsoleto, tareas pendientes >6 meses, dependencias bloqueadas o proyectos estancados) en los procesos principales del SRI durante 2025 (datos estimados y consolidados hasta enero 2026).

Análisis 6: Ante la ilustración presentada y de acuerdo a la gestión de tareas y proyectos que se obtiene, los resultados y análisis esenciales para la institución indican que desde hace ya 6 años (o más) el SRI cuenta con un “inventario” significativo de tareas y proyectos sin resolución o con baja rotación que para la operación no es rentable ni eficiente. Estas tareas acumuladas (backlog obsoleto, dependencias bloqueadas, revisiones normativas pendientes) generan sobrecarga operativa, retrasos en sprints y pérdida de oportunidades de entrega de valor ciudadano.

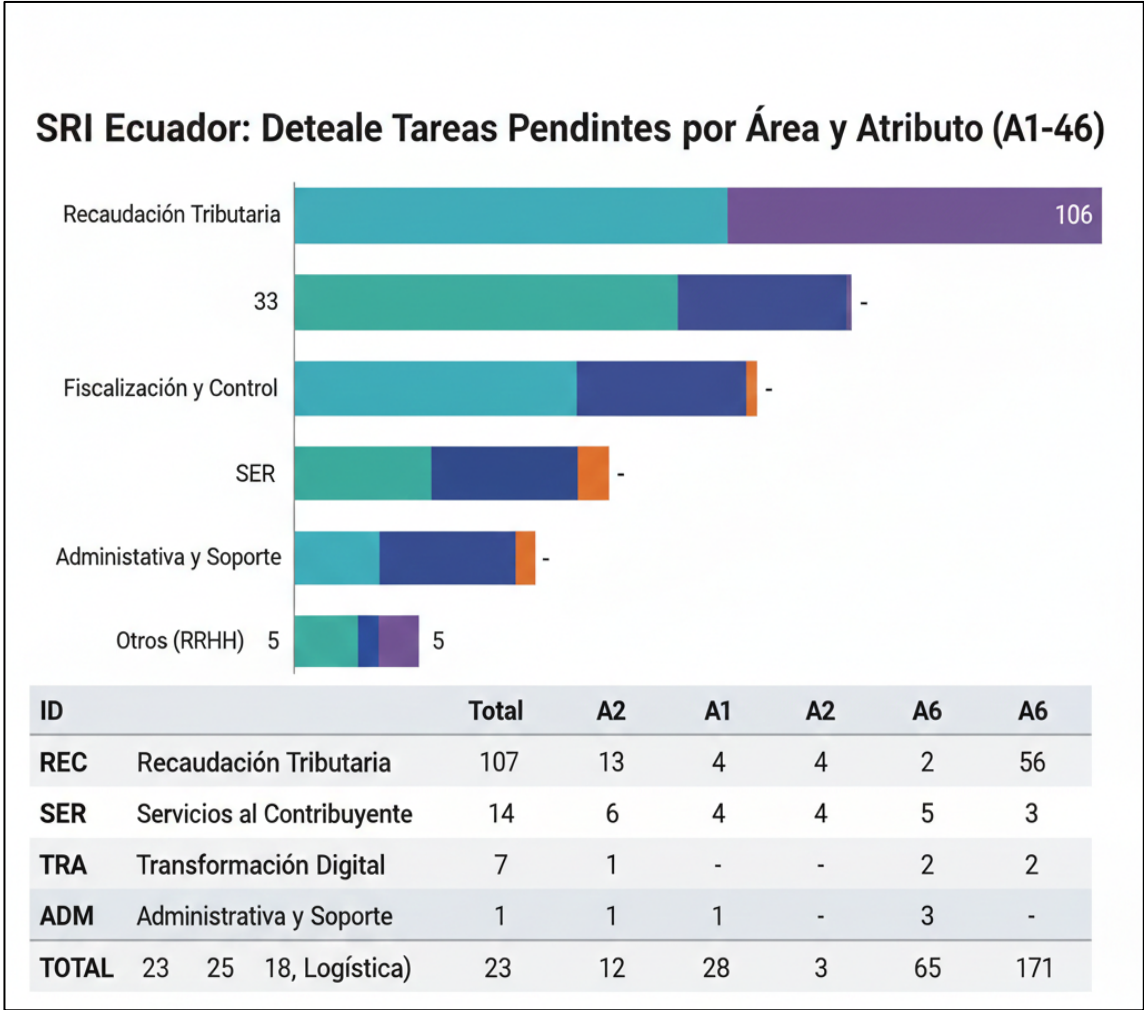
Para evitar la “baja” u obsolescencia de estos elementos (es decir, tareas que se vuelven irrelevantes o generan riesgos de incumplimiento), se espera que con la implementación del gestor inteligente de tareas estos componentes y actividades pendientes puedan ser priorizados automáticamente, desbloqueados mediante alertas predictivas, reasignados con inteligencia y resueltos de forma ágil, evitando pérdidas de tiempo y recursos y generando mayor eficiencia y cumplimiento normativo.

En cuanto a los objetivos planteados para la iniciativa, este descubrimiento subraya la creencia de que poner en línea un sistema de gestión inteligente (con priorización automática por riesgo tributario, detección de tareas estancadas, chatbot para consultas rápidas y trazabilidad inmutable) ayudará a reducir drásticamente el inventario de tareas sin rotación, recuperar productividad en sprints y mejorar la eficiencia operativa general del SRI, contribuyendo a una transformación digital más efectiva y alineada con el Plan Estratégico Institucional 2025-2029 y la Agenda de Transformación Digital extendida hasta 2030. Esto

posicionará al SRI como referente en gestión ágil eficiente dentro del sector público ecuatoriano y regional.

Figura 6

Inventario de tareas sin resolución o con baja rotación en el SRI – 2025



Nota. El gráfico representa cómo se encuentra actualmente el "registro" de tareas sin resolución o con baja rotación (backlog obsoleto, tareas pendientes >6 meses, dependencias bloqueadas o proyectos estancados) por familias/procesos principales del SRI durante 2025 (datos estimados y consolidados hasta enero 2026).

Análisis 7: Ante la ilustración presentada y de acuerdo a la gestión de tareas y proyectos que se obtiene, los resultados y análisis esenciales para la institución indican que desde hace ya 6 años (o más) el SRI cuenta con un “registro” significativo de tareas sin resolución o con baja rotación que para la operación no es rentable ni eficiente. Estas tareas acumuladas (backlog obsoleto, dependencias bloqueadas, revisiones normativas pendientes, proyectos estancados) representan un costo oculto en tiempo, recursos y oportunidades perdidas, generando sobrecarga operativa, retrasos en sprints y riesgos de incumplimiento normativo o de entrega de valor ciudadano.

Para evitar la “baja” u obsolescencia de estos elementos (es decir, que tareas críticas se vuelvan irrelevantes, generen riesgos o pierdan prioridad), se espera que con la implementación del gestor inteligente de tareas estos componentes y actividades pendientes puedan ser identificados automáticamente, priorizados según impacto tributario y riesgo, desbloqueados mediante alertas predictivas, reasignados con inteligencia y resueltos de forma ágil, evitando pérdidas de productividad y generando mayor eficiencia, cumplimiento normativo y entrega de valor público.

En cuanto a los objetivos planteados para el proyecto, este descubrimiento subraya la creencia de que poner en línea un sistema de gestión inteligente (con priorización automática por riesgo tributario, detección temprana de tareas estancadas, chatbot conversacional para consultas rápidas y trazabilidad inmutable para auditorías) ayudará a reducir drásticamente el inventario de tareas sin rotación, recuperar productividad en sprints y mejorar la eficiencia operativa general del SRI, contribuyendo a una transformación digital más efectiva y alineada con el Plan

Estratégico Institucional 2025-2029 y la Agenda de Transformación Digital extendida hasta 2030. Esto permitirá al SRI eliminar “pérdidas” operativas y generar "ganancias" en agilidad, transparencia y servicio al contribuyente (Servicio de Rentas Internas (SRI) 2025).

Análisis e interpretación de resultados

El procedimiento de recolección de datos tuvo las siguientes etapas:

1. **Observación:** El objetivo es analizar el comportamiento real de gestión de tareas y proyectos ágiles (qué tareas se asignan, cómo se coordinan y cuándo se resuelven o bloquean).
 - **Datos a observar:**
 - a) Registros de proyectos ágiles: ¿Qué proceso generó más tareas en los últimos 3 años?
 - b) Canales actuales para la gestión: Reuniones presenciales, correos electrónicos, hojas de cálculo, herramientas básicas como Jira/Azure DevOps.
 - c) Tiempo de respuesta en asignaciones y actualizaciones de sprints.
 - d) Registro de backlog: ¿Qué tipos de tareas son mayormente usadas o estancadas (dependencias normativas, validaciones técnicas)?
 - e) Herramientas: Excel y herramientas internas del SRI (como dashboards de proyectos TIC) para analizar reportes de sprints y datos institucionales.
 - f) Validación: Comparar datos con áreas de desarrollo TIC y normativa tributaria del SRI.
2. **Entrevista:** El objetivo es obtener opiniones directas de equipos Scrum y facilitadores, por lo tanto, las preguntas para equipos/funcionarios serán:

- a) ¿Qué dificultad encuentra al gestionar tareas por los canales tradicionales como: (reuniones, correos o herramientas básicas)?
- b) ¿Utilizaría un sistema inteligente para asignaciones y priorizaciones automáticas?
- c) ¿Con qué frecuencia necesita gestionar tareas críticas en proyectos del SRI?
- d) ¿Utilizaría un sistema inteligente especializado en gestión de tareas Scrum si le permitiera:
- e) Buscar por prioridad/riesgo normativo en el backlog.
- f) Descargar reportes de rendimiento o logs de auditoría.
- g) Recibir alertas predictivas automáticas de bloqueos.
- h) ¿Les gustaría que en futuro se implemente más funciones como integración con IA para predicción de riesgos?

Preguntas para funcionarios/Scrum Masters:

- a) ¿Qué información les falta a los equipos al gestionar tareas?
- b) ¿Qué funcionalidades debería tener el sistema?
- c) ¿Cuáles son los errores más comunes en la gestión de tareas (dependencias no detectadas o falta de stock de recursos)?
- d) ¿Qué porcentaje de tiempo dedican a responder consultas repetitivas sobre estado de tareas o riesgos?
- e) ¿Qué funcionalidades prioritarias debería incluir el sistema para agilizar su trabajo?

Síntesis del capítulo

En el capítulo II se valida toda la recopilación de los resultados establecidos del SRI, las investigaciones usadas fueron la descriptiva y exploratoria, aplicando los métodos inductivos y cualitativos, estos resultados fueron de base fundamental para validar que tan favorable y rentable sea la creación de un gestor inteligente de tareas para la institución.

Capítulo III: Propuesta

Descripción de la propuesta

El proyecto busca desarrollar una solución de gestión inteligente de tareas enfocada en la optimización de procesos ágiles para el equipo Scrum perteneciente al área de TICS del SRI. Esta plataforma digital tiene como objetivo perfeccionar la administración de backlogs, simplificar la asignación y seguimiento de tareas, y potenciar la experiencia de los equipos mediante una interfaz intuitiva y práctica con funcionalidades de IA generativa integrada a través de la API de Gemini.

Considerando el aumento de la transformación digital y la demanda de digitalización de los procesos de gestión en el sector público tributario, esta solución ayudará a incrementar la eficiencia operacional de las instituciones y proporcionar un acceso más agradable a los funcionarios, alineado con las respuestas de las entrevistas que destacaron demoras burocráticas, falta de visibilidad y necesidad de automatización para reducir errores en dependencias y riesgos normativos.

Alcance

El desarrollo del gestor inteligente de tareas para el SRI se centrará en los siguientes aspectos:

1. Backlog digital inteligente:

La administración y publicación de tareas, con detalles técnicos como: prioridades normativas, dependencias, estimaciones de esfuerzo, métricas de rendimiento, etc. Por lo que se gestionará de la siguiente manera:

a. Gestión de Tareas:

- CRUD Completo: Crear, Leer, Actualizar, Eliminar tareas.

- Atributos Técnicos: Campos específicos para Prioridad, Impacto Tributario.
- Dependencias Normativas, Fecha de Sprint, Asignado a Rol.
- (Product Owner, Scrum Master, Equipo).
- Modelos Compatibles: Un sistema para asociar cada tarea con procesos institucionales del
- SRI (recaudación, fiscalización, servicios al contribuyente, transformación digital).
- Medios: Gestión de múltiples logs y reportes por tarea y posiblemente integración con herramientas existentes como SRI en Línea.

b. Sistema de Búsqueda y filtros avanzados:

- Categoría de Proceso (Recaudación, Fiscalización, etc.)
- Prioridad Normativa
- Estado de Tarea (Pendiente, En Progreso, Bloqueada)
- Riesgo (Alto, Medio, Bajo)

c. Experiencia del usuario (UX) e interactividad:

- Dashboard de Sprint: Se podrá añadir/eliminar tareas y ajustar prioridades.
- Visualización del estado total: Previa revisión del backlog y métricas de rendimiento.
- Guardado: Las tareas asignadas se mantendrán a pesar del cierre de la sesión.
- Lista de Prioridades: Cualquier usuario podrá validar sus tareas destacadas y monitorearlas posteriormente.

- **Reseña de Equipos:** La plataforma tendrá reseñas o feedback de retrospectivas de sprints, comentando la experiencia del equipo.

d. Módulo de comunicación y soporte:

- **Formulario de Consulta:** El cual será para consultas generales, solicitud de priorizaciones técnicas o soporte en riesgos normativos.
- **Integración con Redes Internas:** Enlaces visibles y correctamente integrados a sistemas internos del SRI (intranets, herramientas TIC).
- **ChatBot Automatizado (con Gemini):** Implementación de un asistente virtual impulsado por la API de Gemini para responder preguntas frecuentes sobre dependencias, horarios de reuniones o alertas de bloqueos. Capacidad para derivar consultas complejas a Scrum Master y generar respuestas contextuales basadas en datos históricos del backlog.

Limitaciones:

1. No incluirá:

- Integración total con sistemas legales externos al SRI (se gestionará vía APIs limitadas).
- Soporte multiidioma, se trabajará únicamente en español y con métricas en formatos institucionales.
- La plataforma se desarrollará como una aplicación web responsive, capaz de adaptarse automáticamente al tamaño y disposición de la pantalla de los distintos dispositivos de los usuarios.
- El desarrollo de un sistema de logística externa para recursos físicos (se gestionará de forma externa inicialmente).

Viabilidad Económica.

El desarrollo del gestor inteligente de tareas se realizará utilizando tecnologías de código abierto y sin licencias costosas, previamente para el prototipo del desarrollo el cual reduce significativamente los gastos iniciales.

Sin embargo, a mediano y largo plazo, podrían considerarse los siguientes costos asociados que podrán ser útiles al posicionamiento del desarrollo del sistema.

Figura 7:

Viabilidad Económica



Nota: Los valores presentados son referenciales y pueden variar según el proveedor seleccionado por el SRI.

- **Hosting y Dominio:** Firebase Hosting (gratuito hasta cierto límite) y registro de dominio institucional (estimado entre 100 a 300 anuales). Dependiendo de que solicite la organización.
- **Certificado SSL:** Necesario para garantizar seguridad (Firebase provee automáticamente).

- ***Mantenimiento y Actualizaciones:*** La posible necesidad de soporte técnico o desarrolladores externos para mejoras futuras.
- ***Uso de API Gemini:*** Consumo basado en tokens para consultas IA (priorización, alertas, chatbot).

Retorno de la Inversión: La implementación del gestor inteligente de tareas permitirá al SRI obtener los siguientes beneficios económicos a partir de la implementación total del sistema:

- ***Ampliación de la Eficiencia:*** El acceso a herramientas ágiles a nivel institucional sin limitaciones jerárquicas, incrementando potencialmente la productividad mediante reducción de retrasos en sprints.
- ***Reducción de Costos Operativos:*** La disminución de la dependencia de reuniones presenciales y procesos manuales, reduciendo gastos en tiempo administrativo y logística de coordinaciones tradicionales, lo que hoy en día para la institución genera muchos gastos operativos.
- ***Automatización de Procesos:*** Menor tiempo en gestión de tareas y optimización de recursos humanos.
- ***Agilidad en Atención de Proyectos:*** Los equipos podrán realizar gestiones digitales 24/7, agilizando el ciclo de desarrollo y entrega de servicios al contribuyente.

Estos factores contribuyen a incrementar la eficiencia operativa y a generar mayores ahorros institucionales, compensando el costo anual de operación del sistema.

Sostenibilidad Financiera: El modelo planteado garantiza sostenibilidad a largo plazo debido a:

- ***Bajos Costos Fijos:*** Al utilizar tecnologías libres (React, Node.js, Firebase, Tailwind) y consumo controlado de Gemini, los gastos recurrentes son mínimos.
- ***Escalabilidad:*** El sistema puede crecer sin requerir inversiones masivas, adaptándose a mayor volumen de tareas y usuarios.

En conclusión, el proyecto es económicamente viable, ya que implica un gasto anual reducido en comparación con los beneficios posibles que brinda. La disminución de gastos operativos, junto a la automatización de procedimientos y la expansión de la agilidad del SRI, convierten la adopción del gestor inteligente de tareas en algo financieramente viable y lucrativo a medio y largo plazo.

Viabilidad Social.

La factibilidad social del desarrollo de este proyecto se basa en las ventajas que el sistema brindará tanto a equipos Scrum como a la organización SRI. El sistema proporcionará acceso ininterrumpido a un backlog digital inteligente, disminuyendo así el tiempo de gestión de 2 días a 1 hora. Además, ofrecerá asistencia técnica en línea para aclarar inquietudes sobre dependencias normativas o riesgos a través de alertas predictivas y chatbot impulsado por Gemini.

Para la organización, la digitalización potenciará la eficiencia en las operaciones, incrementará la satisfacción de los equipos y establecerá al SRI como pionero en innovación ágil en el ámbito público tributario.

En conclusión, el plan es factible desde el punto de vista social, puesto que ayuda a facilitar el acceso a la información, mejora la satisfacción de los usuarios internos, aumenta la efectividad de la gestión de la institución y produce un efecto

beneficioso en la interacción entre áreas y equipos, ajustándose a las demandas presentes del sector público.

Viabilidad Ambiental.

El proyecto en curso se fundamenta en disminuir el impacto ambiental mediante la digitalización de procedimientos. El sistema suprimirá la necesidad de imprimir reportes, backlogs y pedidos, economizando cerca de 800 hojas de papel anuales. Además, al facilitar la gestión en línea, se disminuirán los traslados de los funcionarios, contribuyendo a la reducción de las emisiones de CO₂ en 1 tonelada anuales.

Estas acciones permiten optimizar recursos y fomentar prácticas institucionales más responsables con el medio ambiente.

En consecuencia, el proyecto es ecológicamente factible, ya que fomenta la disminución del consumo de papel, reduce viajes innecesarios y apoya a una administración más sostenible en el aspecto de responsabilidad social.

Viabilidad Técnica.

La viabilidad técnica del proyecto se fundamenta en el uso de tecnologías modernas y de código abierto que permiten desarrollar un gestor inteligente de tareas interactivo, escalable y sin incurrir en costos por licenciamiento. A continuación, se detallan las herramientas y tecnologías seleccionadas para cada componente del sistema, organizadas por capas:

- **Capa de presentación (Frontend):** Se utilizará React que es una librería de JavaScript con Tailwind CSS para crear una interfaz moderna, responsive y altamente personalizable. Tailwind CSS permite un diseño “utility-first” con clases de bajo nivel aplicadas directamente en el código, favoreciendo

la flexibilidad y la adaptación a necesidades específicas del SRI. La arquitectura de componentes reutilizables asegura consistencia y mantenibilidad.

- **Capa de lógica de negocio y API (Backend):** Implementada utilizando Node.js, que proporciona un entorno de ejecución eficiente y escalable para la lógica del servidor. Node.js maneja todas las operaciones de negocio, validaciones, procesamiento de datos y generación de respuestas inteligentes.
- **Capa de persistencia de datos:** Firebase Realtime Database / Firestore se utiliza como base de datos NoSQL en tiempo real, permitiendo sincronización instantánea de backlogs y estados de tareas entre usuarios. Firebase ofrece escalabilidad automática, seguridad integrada y facilidad de integración con React y Node.js.
- **Capa de inteligencia artificial:** La API de Gemini (Google Generative AI) se integra para funcionalidades avanzadas: generación de sugerencias de priorización, análisis predictivo de riesgos en sprints, respuestas contextuales del chatbot y generación automática de reportes basados en datos históricos.
- **Infraestructura:** El frontend se despliega en Vercel o Firebase Hosting, mientras que el backend (Node.js) y la base de datos (Firebase) se alojan en servicios compatibles con Firebase. Firebase provee HTTPS automáticamente, autenticación integrada y escalabilidad global.
- **Equipo de desarrollo:** El desarrollo lo realizará 1 persona tanto para la programación, diseño y desarrollo de la aplicación, siendo así la garantía de

una experiencia de usuario óptima. Esta arquitectura, basada en tecnologías de código abierto y servicios en la nube como Firebase, no solo reduce los costos asociados a licencias, sino que también garantiza flexibilidad, escalabilidad y seguridad, lo que confirma la viabilidad técnica del proyecto. Por lo tanto, se determina que la iniciativa es factible desde el punto de vista técnico, dado que las tecnologías elegidas son apropiadas, escalables y tienen un amplio respaldo, asegurando consistencia, protección y sostenibilidad técnica para el desarrollo y el avance futuro del sistema inteligente.

Tabla 5
Tecnología a usar

Características	Descripción
Capas	5
Nube	100%
Sincronización	Real-time
Inteligencia Artificial	AI Powered
Stack Tecnológico	
Frontend	React y Tailwind CSS
Backend	Node.js y Type.js
Base de Datos	Firebase
Inteligencia Artificial	Gemini API
DevTools	VS Code

Nota: La tabla indica todos los tipos de tecnologías a usar.

Impacto

El desarrollo del gestor inteligente de tareas para equipos Scrum del SRI generará impactos significativos en los ámbitos económico, social y ambiental.

Estos impactos no solo beneficiarán a la institución SRI, sino también a sus equipos internos y al sector público tributario en general, promoviendo la modernización y sostenibilidad de los procesos de gestión ágil.

Impacto Económico

El desarrollo del sistema tendrá como beneficios económicos favorables tanto como para la institución como para los equipos. Implementar el gestor en el SRI favorecerá un crecimiento en la eficiencia operativa al facilitar el acceso a herramientas ágiles y expandir la capacidad de gestión de proyectos digitales. De igual manera, la automatización de procesos reducirá los gastos operativos derivados de la gestión manual de tareas, como la elaboración de reportes y coordinaciones tradicionales, lo que se traducirá en un ahorro significativo a mediano y largo plazo.

De igual forma, los equipos podrán acceder a métricas más competitivas gracias a la disminución de los costos operativos y a la facilidad que ofrece el sistema para comparar y priorizar tareas. El sistema también ofrecerá alertas predictivas y optimizaciones exclusivas, lo que incentivará la productividad recurrente y mejorará la retención de talento ágil. En conjunto, estos factores contribuirán a un crecimiento económico sostenible para la institución y sus stakeholders.

Impacto Social

Los beneficios sociales del proyecto se manifestarán en una mejor experiencia para los equipos y en una mayor eficiencia operativa del SRI. El sistema ofrecerá un entorno interactivo de uso sencillo, que permitirá a los funcionarios realizar transacciones de tareas de manera eficiente y con altos estándares de

seguridad, sin importar el tipo de dispositivo que utilicen. Además, el soporte técnico en línea resolverá dudas en tiempo real, lo que mejorará la satisfacción de los equipos y reducirá los tiempos de espera.

Para la institución, la digitalización permitirá una gestión más eficiente de los sprints, reduciendo errores y optimizando el tiempo de respuesta. Con esta mejora, se optimizará el desempeño del equipo de trabajo y se facilitará la asignación de recursos a áreas prioritarias, como el desarrollo de nuevos servicios digitales o el fortalecimiento de la colaboración con áreas internas. En resumen, el proyecto contribuirá a la modernización del sector público tributario y al incremento del bienestar tanto de los funcionarios como de los contribuyentes.

Impacto Ambiental

El proyecto tendrá un impacto ambiental positivo al promover la sostenibilidad y reducir el uso de recursos no renovables. La digitalización de los procesos eliminará la necesidad de imprimir reportes, backlogs y pedidos, lo que se traducirá en un ahorro de papel y una reducción en la huella de carbono. Además, al permitir la gestión en línea, se reducirán los desplazamientos de los funcionarios, contribuyendo a la disminución de las emisiones de CO₂.

El sistema también se alojará en servidores que utilizan energías renovables (Firebase), lo que garantizará un funcionamiento más sostenible. Estas medidas no solo beneficiarán al medio ambiente, sino que también posicionarán al SRI como una institución comprometida con la responsabilidad ambiental, lo que puede mejorar su imagen pública y atraer a colaboradores con conciencia ecológica.

Desarrollo de la propuesta

La formulación de la propuesta se fundamenta en los hallazgos derivados del análisis cualitativo de las entrevistas realizadas a los funcionarios del SRI (Scrum Masters, Product Owners, desarrolladores TIC), lo que permitió reconocer las necesidades, desafíos y expectativas vinculadas con el proceso de gestión de tareas ágiles. Con base en estos datos, se propone el diseño y creación de un gestor inteligente de tareas enfocado en mejorar la coordinación institucional, reducir los tiempos de sprints y fortalecer la comunicación entre áreas. Teniendo en cuenta que, gracias a la integración de pipelines automatizados, alertas predictivas y dashboards personalizados por rol se podrán automatizar procesos como es el caso de las reuniones presenciales para tener una respuesta directa con presuntos bloqueos, estado de las tareas, proyectos en curso entre otro más través del asistente de IA.

Esta propuesta incluye características esenciales como la visualización inteligente de backlogs, alertas predictivas de riesgos y trazabilidad normativa, que se encuentran alineadas con los requerimientos del sector público tributario, según las respuestas de las entrevistas que destacaron demoras por burocracia, falta de visibilidad en dependencias y necesidad de automatización para mitigar errores manuales. Así, el desarrollo de este sistema busca abordar de manera directa el problema detectado, ofreciendo una solución tecnológica viable que ayude a modernizar y aumentar la competitividad del SRI en el ámbito nacional e internacional,

Objetivos de la propuesta.

1. Diseñar la arquitectura del gestor inteligente de tareas, definiendo las capas de presentación, lógica de negocio y datos, de acuerdo con las buenas prácticas del desarrollo de aplicaciones web y cumplimiento normativo.
2. Implementar un backlog digital inteligente de tareas, que incluya información técnica, prioridades normativas, dependencias y métricas de rendimiento.
3. Desarrollar los módulos funcionales del sistema, tales como gestión de usuarios y administración de sprints, utilizando tecnologías de código abierto y servicios API.
4. Aplicar la metodología SCRUM durante el desarrollo del sistema, organizando el trabajo en iteraciones (sprints) y entregas incrementales del sistema.
5. Realizar pruebas funcionales y de usabilidad, con el fin de verificar el correcto funcionamiento del sistema en un entorno de pruebas controlado.
6. Desplegar el sistema en un ambiente simulado, evaluando su estabilidad, seguridad y rendimiento antes de su puesta en producción.

Metodología de desarrollo

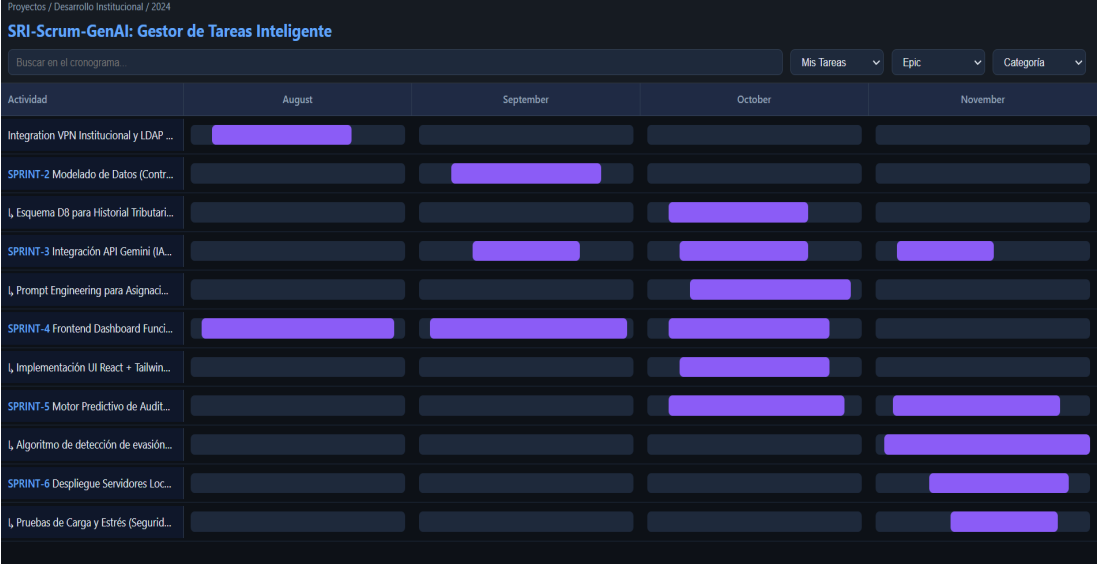
Para el desarrollo del sistema se aplicó una metodología ágil SCRUM por el enfoque iterativo e incremental, lo que permite realizar entregas parciales del sistema en ciclos llamados sprints. SCRUM promueve la participación de los usuarios (en este caso representantes del SRI) en cada fase del desarrollo,

permitiendo validar avances de manera continua. A continuación, se definirán los roles y artefactos que son:

Roles definidos:

- **Product Owner:** Responsable de definir los requisitos del producto.
Representante del SRI.
- **Scrum Master:** Facilitador del equipo de desarrollo.
- **Equipo de Desarrollo:** Encargado de diseñar, programar y probar la solución.

Figura 9
Marco de trabajo Scrum



Nota: La figura representa al marco de trabajo que se usó para el proyecto fue Scrum.

Artefactos:

- **Product Backlog:** Lista priorizada de funcionalidades.
- **Sprint Backlog:** Funcionalidades seleccionadas para cada sprint.
- **Incremento:** Producto parcialmente funcional entregado al final de cada sprint.

Fases del Desarrollo

Justificación del stack de desarrollo

El stack de tecnologías equilibra modernidad tecnológica con realismo institucional: prioriza seguridad, integración, aplicabilidad y bajo costo de mantenimiento, alineándose directamente con las barreras y expectativas expresadas por el equipo del SRI en las encuestas. Es escalable desde un piloto pequeño hasta una solución de toda la institución, sin comprometer el cumplimiento normativo ni la trazabilidad requerida en el sector público ecuatoriano. Debido a que el sector público ecuatoriano no cuenta con una migración y actualización recurrente en las tecnologías en su mayor parte cuentan con sistemas y plataformas con lenguajes y frameworks desactualizados no obstante son sistemas mucho más robustos con lo cual se tiene una preferencia por temas de seguridad y protección. Aun así, se debe considerar la UI/UX y optimizar procesos y recursos con nuevas actualizaciones tanto en lenguajes como funciones e integración de IA en las plataformas.

Cada iteración comprendió:

1. **Análisis de Requisitos:** Durante esta etapa, se reunió la información necesaria a través de entrevistas con el equipo del SRI. Se identificaron los módulos esenciales del sistema. Como son:
 - Gestión de usuarios (Backend)
 - Backlog de tareas
2. **Diseño:** Se utilizó la metodología de diseño basada en arquitectura de capas (presentation, business logic, data), separando la lógica del

negocio, la presentación y la interacción con la base de datos. Por lo que las herramientas son las siguientes.

- Base de datos: Firebase Realtime Database / Firestore
3. **Implementación:** El sistema se basa en el desarrollo y uso de tecnologías de código abierto como son:

Tabla 6:

Componentes y tecnologías

Componente	Tecnología
Presentación	React + Tailwind CSS
Negocio	Node.js
Datos	Firebase Realtime Database / Firestore
Inteligencia Artificial	API de Gemini
Herramienta de desarrollo	Visual Studio Code, GitHub
Validación de peticiones	Postman

Nota: La tabla representa los componentes y tecnologías usadas en el proyecto cada módulo fue implantado como una API REST en el backend y consumido por el frontend React.

4. **Pruebas:** Evaluación de calidad, corrección de errores y mejoras.

Que son:

- **Pruebas unitarias:** para validar funciones individuales del backend.
- **Pruebas de integración:** para verificar el flujo entre los componentes del sistema.

- ***Pruebas funcionales:*** para comprobar que cada módulo cumple con los requisitos definidos.
 - ***Pruebas de usuario:*** con personal del SRI para validar la usabilidad y funcionalidad.
5. ***Control de Versiones y Documentación:*** Para gestionar las versiones del proyecto se adoptó Git, una herramienta de software libre ampliamente reconocida por facilitar el seguimiento de los cambios en el código. El repositorio del proyecto se alojó en GitHub, facilitando así una estructura de código más ordenada, copias de seguridad seguras y la opción de automatizar los despliegues desde la propia plataforma.
6. ***Validación de peticiones:*** Se usó el programa Insomnia para la validación de peticiones que se usa en Node.js y poder verificar los siguientes:
- Los endpoints responden correctamente
 - Los datos se envían y reciben en el formato correcto
 - La autenticación funciona
 - Las respuestas son las esperadas

Principales EndPoints probados son:

Figura 10
End Points

ENDPOINT	MÉTODO	PROPÓSITO TÉCNICO	RESULTADO	FUNCIONALIDAD IA GEMINI ✨
/api/tasks	GET	Recuperar el backlog global de tareas del equipo Scrum del SRI.	200 OK	Priorización inteligente basada en impacto tributario y dependencias.
/api/tasks/:slug	GET	Consultar detalle de una tarea específica (ej: validación facturación).	200 OK	Resumen Automático de normativa tributaria y sugerencias de cumplimiento.
/api/ai/predict-risk	POST	Análisis predictivo de riesgos en conjunto de tareas.	201 Created	Alerta de riesgo basada en LOPDP/COT y probabilidad histórica de incumplimiento.
/api/ai/chatbot/query	POST	Interacción conversacional sobre el estado del backlog.	200 OK	Responde en lenguaje natural y sugiere acciones correctoras inmediatas.
/api/tasks/bulk-update	PATCH	Actualización masiva de prioridades en un sprint.	200 OK	Re-priorización masiva autónoma por urgencia normativa detectada .

Nota: La figura representa a los Endpoints utilizados en el desarrollo del proyecto.

7. **Despliegue:** Se optó por Firebase Hosting para poner en marcha el sistema, una plataforma abierta pensada para apps hechas con React. Esta herramienta facilita que los proyectos se publiquen automáticamente desde GitHub, además de encargarse de configurar HTTPS, crecer sin problemas a escala mundial y ofrecer un manejo sencillo.

Análisis.

Se realizó un análisis detallado de los requerimientos funcionales y no funcionales del sistema. Para ello, se aplicaron técnicas como entrevistas con usuarios clave, revisión de documentación.

Requerimientos Funcionales Estructurados.

El análisis de requerimientos es una fase fundamental en el desarrollo del gestor inteligente de tareas, ya que define las funcionalidades que el sistema debe

ofrecer, por lo que los requerimientos funcionales y el control de calidad que debe satisfacer los requerimientos no funcionales. Estos requisitos aseguran que el sistema responda adecuadamente a las demandas de los usuarios finales y se mantenga dentro del alcance establecido para el proyecto.

RF1. Visualización del backlog de tareas.

- El sistema debe garantizar que las tareas puedan ser visualizadas por cualquier persona autorizada, aun sin autenticación previa para roles de lectura. La información incluirá: nombre, prioridad, dependencias, estimaciones.

RF2. Filtro y búsqueda de tareas.

- Los usuarios podrán filtrar las tareas por categorías para facilitar su consulta.

RF3. Formulario de consulta.

- Los usuarios podrán enviar un formulario con su nombre, correo electrónico, teléfono y mensaje. El formulario podrá estar asociado a una tarea consultada o ser general.

RF4. Gestión de backlog por el administrador.

- Desde el panel de administración, se podrá agregar, editar, eliminar o desactivar tareas, así como organizar tareas por sprint. Así también incluye las funcionalidades de recuperación de credenciales, historial de usuarios que hayan accedido al sistema, agregar a nuevos usuarios entre otras más.

RF5. Gestión de sprints.

- El administrador podrá crear y modificar los sprints para organizar el backlog. Solamente en casos particulares debido a que el Scrum Master cuenta con este privilegio primordial.

RF6. ChatBot (Gemini).

- Se generó un asistente de ayuda es decir ChatBot impulsado por Gemini, para preguntas comunes que pueda realizar el usuario, también incluyen las funcionalidades de generación de gráficos, exportación de reportes, recordatorios entre otras más.

Requerimientos No Funcionales.

Los requerimientos no funcionales especifican los atributos de calidad y las restricciones que la solución a implementar tiene que cumplir. A continuación, se detallan los principales requerimientos no funcionales:

RNF1. Rendimiento:

- Tiempo de respuesta máximo de 3 segundos para cualquier operación.
- Capacidad para soportar hasta 1,000 usuarios concurrentes sin degradación del rendimiento.

RNF2 Seguridad:

- Autenticación de usuarios a partir de la verificación de un captcha de seguridad y protección a las credenciales con funciones hash contra ataques SQL Inyection, contraseñas más robustas combinando mayúsculas y minúsculas junto con caracteres especiales.

RNF3 Mantenibilidad:

- El desarrollo de contar con una arquitectura modular que facilite su mantenimiento y evolución.
- El código fuente debe estar documentado y versionado con Git.
- Se deben separar las capas de frontend, backend y base de datos.
- Se adicionaron privilegios de usuario al rol de administrador para que pueda dar un mejor mantenimiento y soporte de la aplicación.

RNF4 Compatibilidad:

- El sistema debe ser compatible con los navegadores web más comunes: Google Chrome, Edge y Brave.
- Debe adaptarse a diferentes resoluciones de pantalla (diseño responsive).

RNF5 Usabilidad:

- La interfaz debe ser intuitiva, amigable y fácil de usar para usuarios no técnicos.

RNF6 Disponibilidad:

- El sistema debe estar disponible 24 / 7
- Debe mostrar mensajes adecuados cuando no esté disponible por mantenimiento.

El diseño final de la arquitectura tomó un rumbo distinto al propuesto inicialmente, optando por una estructura con React como motor del frontend, Node.js como backend y Firebase como base de datos en tiempo real, lo cual agilizó bastante el desarrollo y nos dio un panel de administración muy robusto. El sistema está organizado en estas capas, que reflejan cómo se implementó lo siguiente:

- A. ***Capa de presentación (Frontend):*** Se construyó con React + Tailwind CSS. Es una aplicación web “responsiva” que corre directamente en el navegador del usuario.
- B. ***Capa de lógica de negocio y API (Backend):*** Aquí entra con Node.js en el cual este framework maneja toda la lógica, el contenido y proveyendo una API RESTful automática y bien documentada.
- C. ***Capa de persistencia de datos:*** Se usó Firebase Realtime Database / Firestore como sistema de gestión de bases de datos en tiempo real. Fue elegido por su sincronización instantánea y robustez para manejar el backlog de tareas.
- D. ***Capa de inteligencia artificial:*** La API de Gemini (Google Generative AI) se integra para funcionalidades avanzadas: generación de sugerencias de priorización, análisis predictivo de riesgos en sprints, respuestas contextuales del chatbot y generación automática de reportes basados en datos históricos.

La comunicación entre estas capas sigue un esquema basado en API, donde el frontend accede a la información del backend exclusivamente a través de endpoints RESTful.

Arquitectura del sistema implementada.

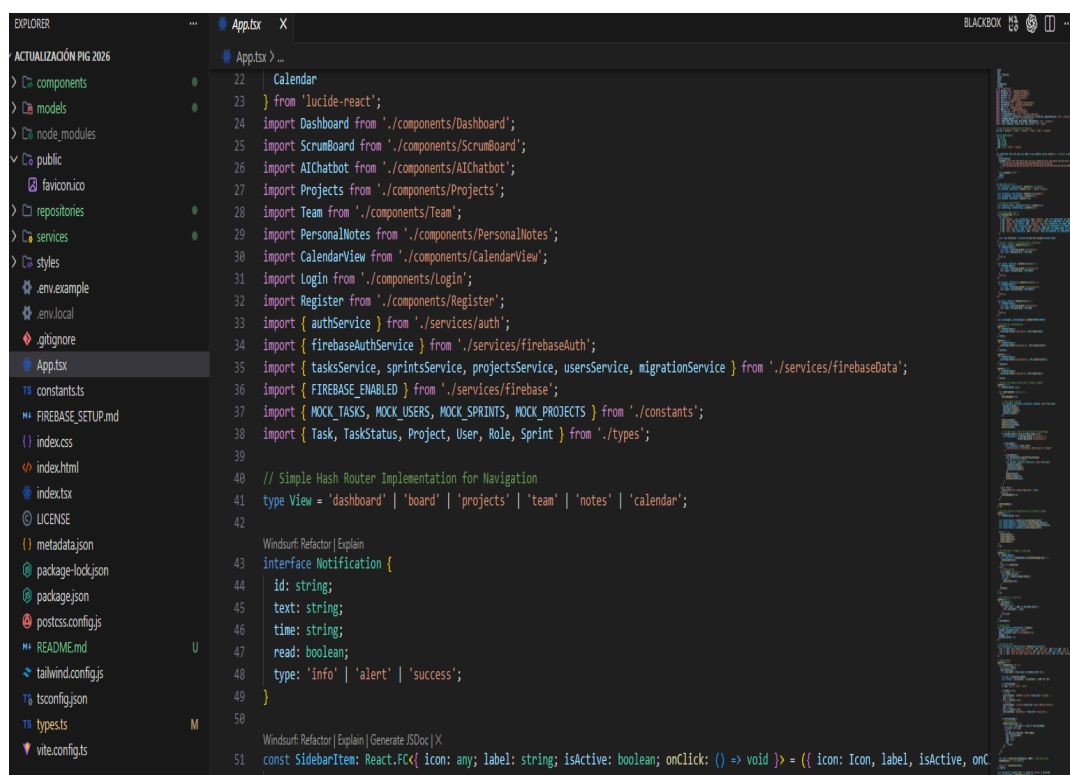
El sistema sigue una arquitectura de aplicación web moderna basada en capas desacopladas (presentation, business logic, data, AI), separando claramente el frontend del backend y utilizando APIs RESTful para la comunicación entre componentes. La arquitectura es serverless-friendly (Firebase) y aprovecha IA generativa (Gemini) para funcionalidades inteligentes.

Los componentes principales son:

Capa de Presentación (Frontend): Desarrollada con React + Tailwind CSS. Esta será una aplicación web adaptable (responsive) que funciona directamente en el navegador del usuario. Se enviarán solicitudes HTTP a la API para establecer comunicación con el backend. React maneja la interfaz de usuario dinámica (dashboard de backlog, vistas de sprints, detalle de tareas, chatbot), mientras que Tailwind CSS asegura un diseño limpio, moderno y altamente personalizable. La capa utiliza Hooks para gestión de estado local y TanStack Query para caché y sincronización optimizada con Firebase.

Figura 11:

Capa de presentación



```

22   Calendar
23 } from 'lucide-react';
24 import Dashboard from './components/Dashboard';
25 import ScrumBoard from './components/ScrumBoard';
26 import AIChatbot from './components/AIChatbot';
27 import Projects from './components/Projects';
28 import Team from './components/Team';
29 import PersonalNotes from './components/PersonalNotes';
30 import CalendarView from './components/CalendarView';
31 import Login from './components/Login';
32 import Register from './components/Register';
33 import { authService } from './services/auth';
34 import { firebaseAuthService } from './services/firebaseAuth';
35 import { tasksService, sprintsService, projectsService, userService, migrationService } from './services/firebaseData';
36 import { FIREBASE_ENABLED } from './services/firebase';
37 import { MOCK_TASKS, MOCK_USERS, MOCK_SPRINTS, MOCK_PROJECTS } from './constants';
38 import { Task, TaskStatus, Project, User, Role, Sprint } from './types';
39
40 // Simple Hash Router Implementation for Navigation
41 type View = 'dashboard' | 'board' | 'projects' | 'team' | 'notes' | 'calendar';
42
43 Windsurf: Refactor | Explain
44 interface Notification {
45   id: string;
46   text: string;
47   time: string;
48   read: boolean;
49   type: 'info' | 'alert' | 'success';
50 }
51
52 Windsurf: Refactor | Explain | Generate JSDoc | X
53 const SidebarItem: React.FC<{ icon: any; label: string; isActive: boolean; onClick: () => void }> = ({ icon: Icon, label, isActive, onClick }) => {

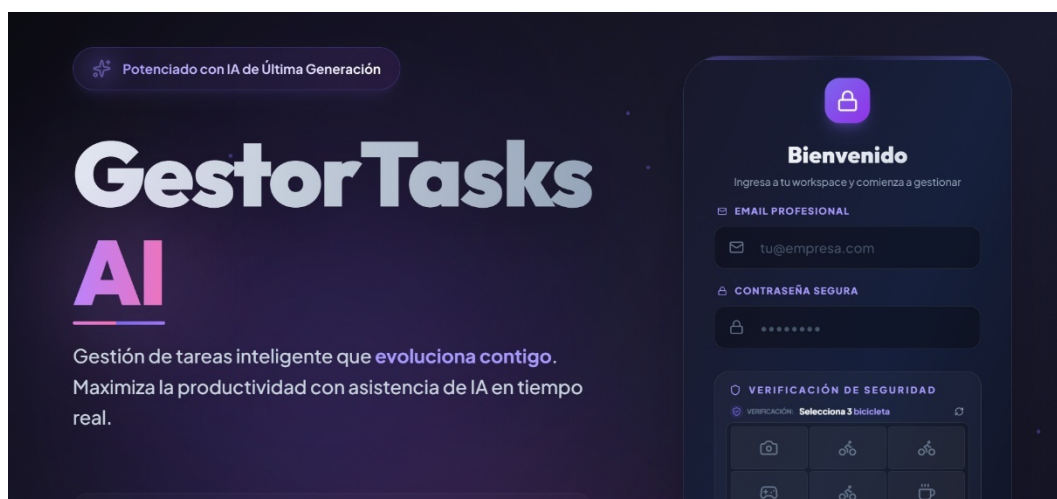
```

Nota. Representación del código que contiene a la capa de presentación del proyecto.

Capa de Lógica de Negocio y API (Backend): Implementada con Node.js. Esta capa expone una API RESTful bien definida que es consumida por el frontend. Se encarga de toda la lógica de negocio (validaciones, cálculos de prioridad, gestión de roles y permisos RBAC), procesamiento de peticiones, enriquecimiento de datos y orquestación de llamadas a Gemini para funcionalidades de IA. Node.js actúa como puente seguro entre frontend, Firebase y Gemini, aplicando autenticación JWT y trazabilidad obligatoria para cumplimiento normativo (Webmaster 2025); (Lexis 2026).

Figura 12:

Ventana de inicio de sesión para usuarios



Nota. Representación de la vista principal de la aplicación web con el inicio de sesión para los usuarios y la opción de registro.

Estructura del Diseño:

- **Interfaz Oscura:** Implementa un esquema de color de fondo oscuro (probablemente en color negro o un azul muy oscuro), que reduce la fatiga visual y mejora la legibilidad de los elementos UI.

- **Tipografía:** Usa fuentes modernas y legibles, con tamaños que resaltan los elementos principales, como el nombre de la plataforma y los campos del formulario.

Elementos Funcionales:

Formulario de Inicio de Sesión: Incluye dos campos de entrada:

- **Email Profesional:** Un campo de texto para que los usuarios introduzcan su dirección de correo electrónico.
- **Contraseña:** Un campo de entrada que oculta los caracteres ingresados para proteger la información sensible.
- **Botón de Acceso:** Un botón prominente ("Acceder a Workspace") que ejecuta una función de autenticación al ser presionado. Normalmente, está asociado con un evento de tipo onClick que impulsa el proceso de inicio de sesión.

Acciones Adicionales:

- **Opción de Crear Cuenta:** Proporciona un enlace para usuarios nuevos, lo que sugiere que el sistema incluye un flujo de registro separado. Este enlace normalmente conduce a un formulario de registro.

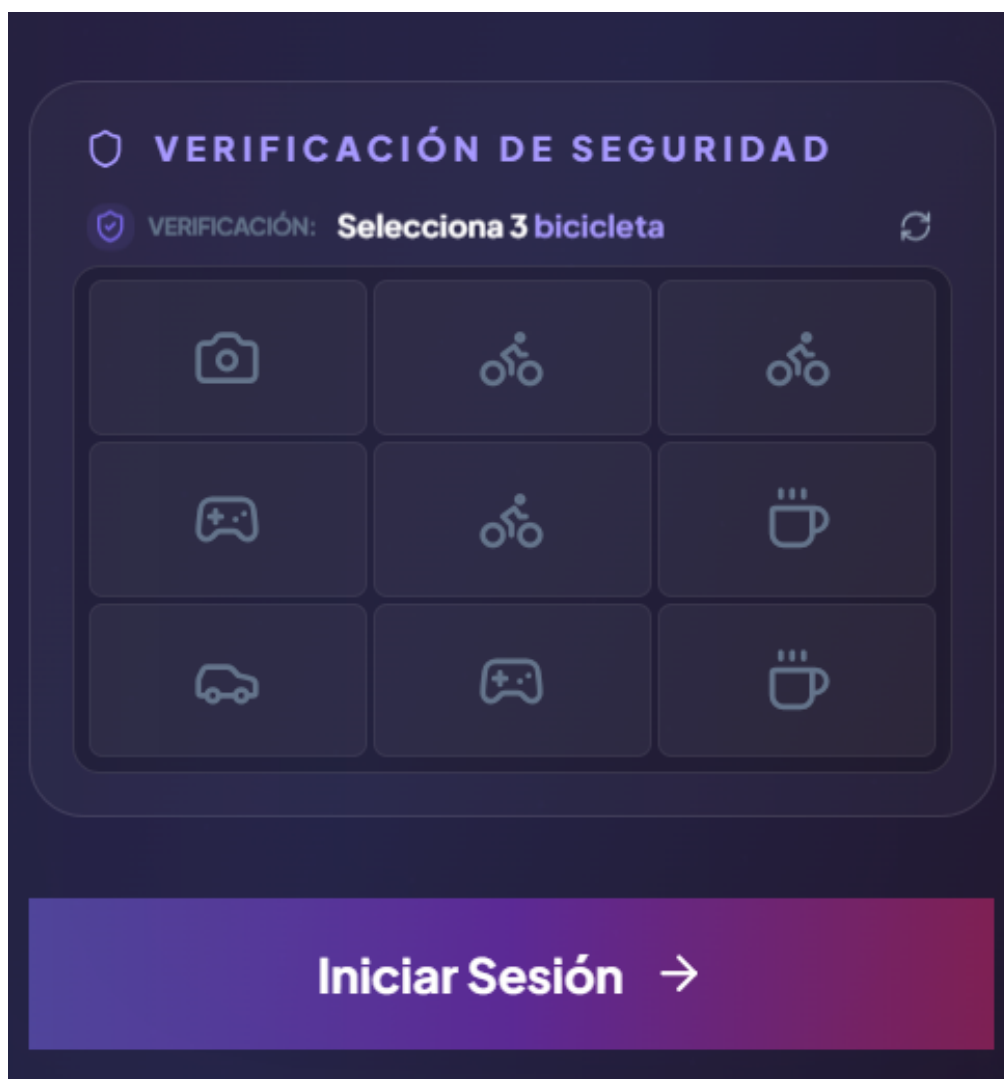
Tecnología Subyacente:

- **Autenticación:** Es probable que la plataforma use un protocolo seguro (como OAuth o JWT) para la autenticación de usuarios, garantizando la transmisión de datos de forma segura.
- **Responsive Design:** Se espera que la interfaz sea completamente adaptable a diferentes dispositivos, utilizando técnicas de diseño web responsivo (CSS

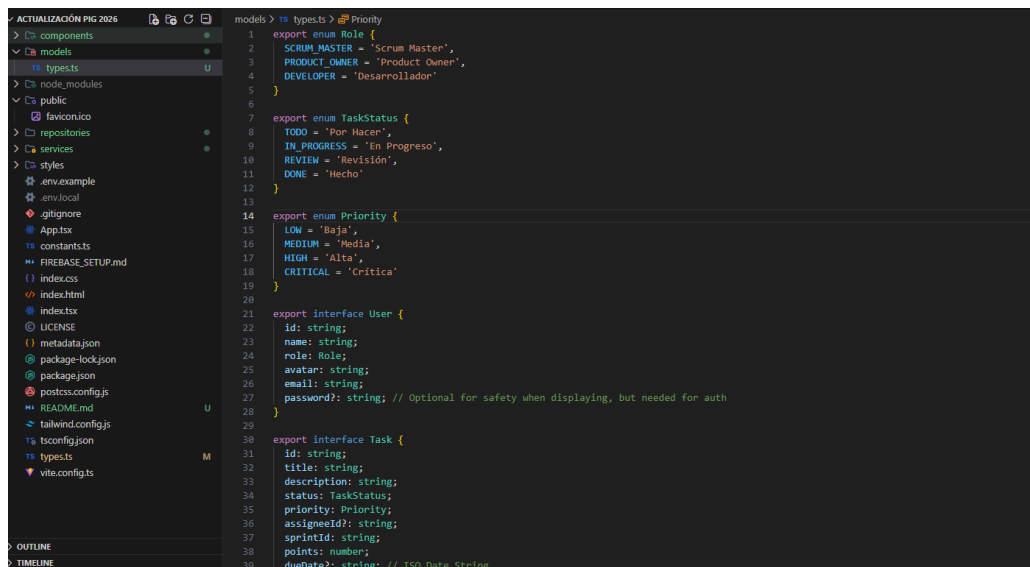
Grid o Flexbox) para asegurar su funcionalidad en pantallas de diversos tamaños.

Figura 13:

Captcha de seguridad



Nota. Captcha de seguridad para contar con una validación mucho más sólida para los usuarios del gestor de tareas. Se implementó una validación de seguridad con imágenes de múltiples objetos que se generan aleatoriamente con una función de IA generativa que se va actualizando continuamente cada vez que se recargue la página.

Figura 14*Base de datos*

Nota. Visualización del código que contiene a la base de datos NoSQL. Persistente, descentralizada y en tiempo real en la nube de Google altamente recomendable para el desarrollo de la aplicación web.

Capa de Inteligencia Artificial (Gemini): Integrada mediante la API de Google Generative AI. Esta capa se activa en endpoints específicos (/api/ai/predict-risk, /api/ai/summarize-task, /api/ai/chatbot/query) para:

- Priorización automática de tareas según riesgo tributario y normativo.
- Generación de alertas predictivas de bloqueos o incumplimiento.
- Resúmenes ejecutivos de sprints o tareas para reportes de auditoría.
- Chatbot conversacional que responde consultas en lenguaje natural y accede al backlog en tiempo real.

Gemini procesa prompts contextuales (datos del backlog + normativas SRI) y devuelve respuestas en formato JSON estructurado, que Node.js enriquece y retorna al frontend. La comunicación entre capas sigue un esquema basado en API RESTful + sincronización en tiempo real (Firebase), donde:

- Frontend → Backend (HTTP/REST)
- Backend ↔ Firebase (lectura/escritura en tiempo real)
- Backend ↔ Gemini (prompts y respuestas JSON)

Esta arquitectura desacoplada asegura:

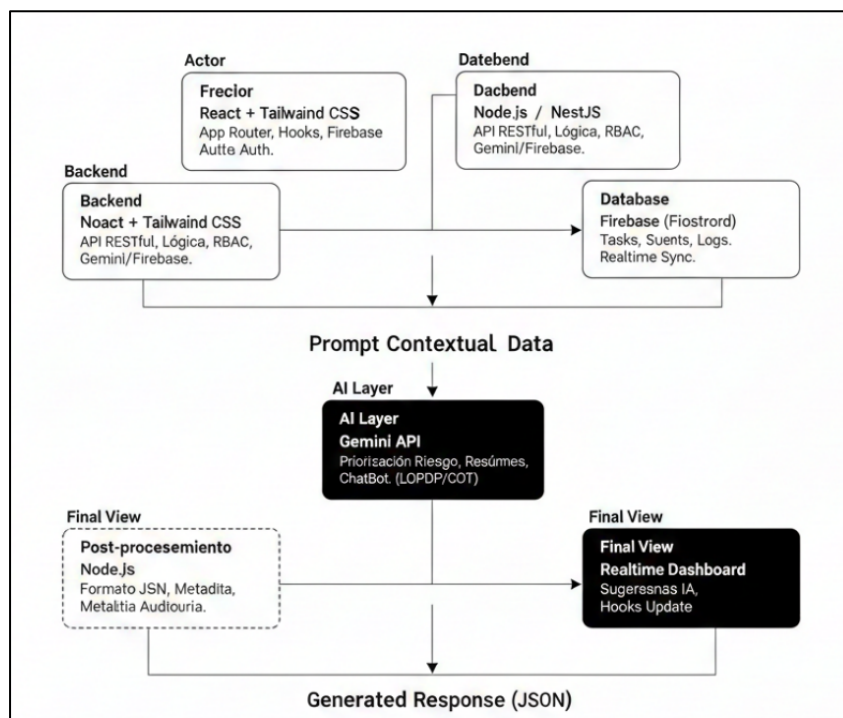
- Escalabilidad (Firebase serverless + Node.js ligero)
- Seguridad (JWT + RBAC + HTTPS + reglas Firebase)
- Rendimiento (sincronización en tiempo real + caché TanStack Query)
- Inteligencia (Gemini para automatización predictiva y conversacional)
- Cumplimiento normativo (trazabilidad inmutable y logs auditables)

Diagrama de despliegue y flujo de datos.

El siguiente diagrama muestra la estructura independiente y el movimiento de la información que se estableció.

Figura 15:

Diagrama de flujo



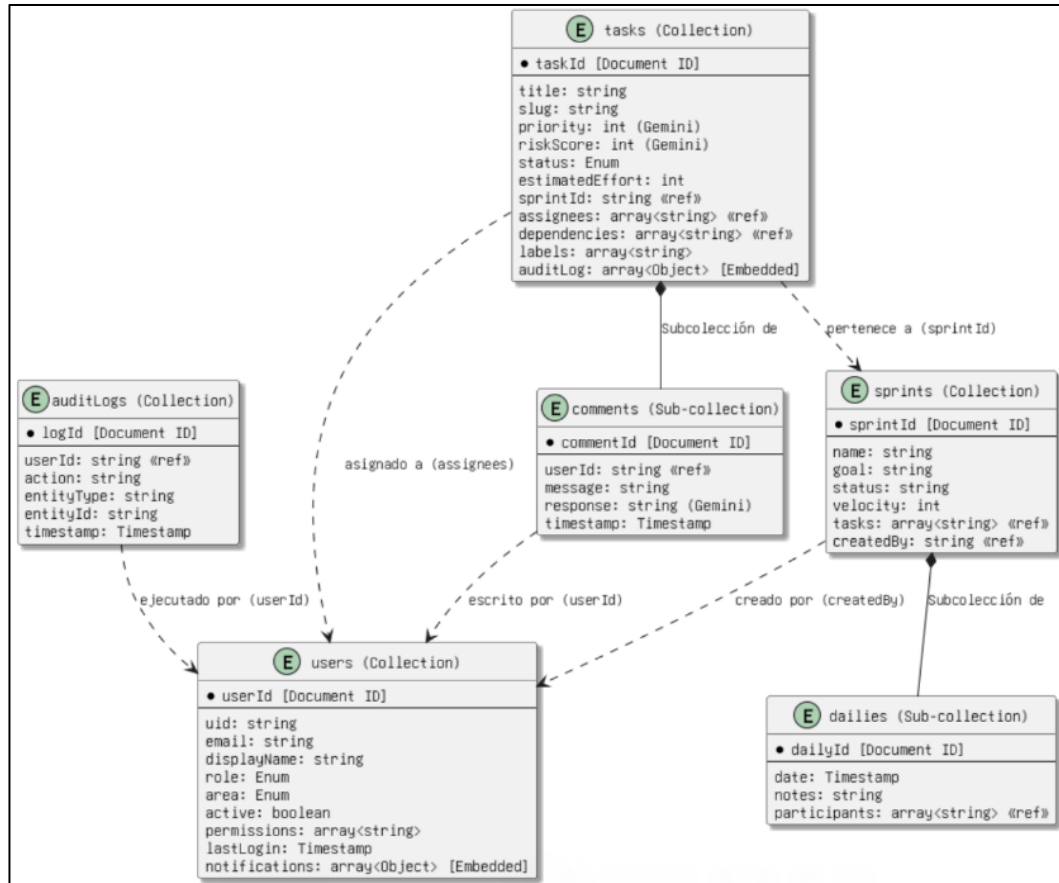
Nota: El gráfico representa como se encuentra establecido el diagrama de flujo.

Descripción del diagrama:

1. Cuando un funcionario interactúa con la plataforma, como visitar la página de un backlog.
2. React, ya sea directamente en el navegador o desde el servidor, manda una petición GET a la API de Node.js. Esta petición podría ser algo como "GET /api/tasks?filters[slug][\$eq]=mi-tarea".
3. Node.js, al recibir esta petición, ejecuta su lógica interna, aplicando filtros y obteniendo la información necesaria, y luego consulta Firebase.
4. Firebase, a su vez, le proporciona los datos solicitados a Node.js.
5. Node.js y Nest.js toman esta información, la estructura en formato JSON y se la envía de vuelta al frontend.
6. React recibe estos datos, los utiliza para construir la interfaz con componentes, y se la muestra al usuario.

Diagrama de Datos Implementado (Modelo de Datos NoSQL).

El modelo de datos que se ha implementado tanto en Node.js como en Firebase, soporta la arquitectura (React frontend, Node.js backend, Firebase datos, Gemini IA), y se basa en los requerimientos estudiados (visibilidad, colaboración, trazabilidad, inteligencia predictiva).

Figura 16:*Modelo de Datos NoSQL*

Nota: La gráfica muestra cómo se encuentra estructurado el modelo no relacional.

Adaptado de: (Diagrama No SQL- draw.io 2026).

Estructura NoSQL en Firebase:

- Colecciones principales para entidades independientes (users, tasks, sprints).
- Subcolecciones para relaciones 1: N (ej. comments dentro de tasks para consultas, dailies dentro de sprints para daily scrums), optimizando consultas locales y reduciendo costos.
- Arrays embebidos para relaciones frecuentes (dependencies, assignees, labels, auditLog) — de normalización para lecturas rápidas sin joins.

- Referencias (ID) para relaciones complejas (sprintId en tasks, userId en logs), permitiendo queries eficientes con Firebase SDK.

Ventajas para el Gestor de Tareas:

- **Escalabilidad:** Documentos flexibles permiten agregar campos dinámicos (ej. nuevo atributo "geminiSuggestion" sin migraciones).
- **Tiempo Real:** Firebase listeners (onSnapshot) sincronizan cambios en backlogs/sprints instantáneamente para colaboración Scrum.
- **Seguridad y Cumplimiento:** Reglas de seguridad Firebase por rol (RBAC: solo Scrum Master edita sprints) y logs inmutables para auditorías (Webmaster 2025).
- **Integración con Gemini:** Documentos JSON se envían directamente como prompts a Gemini para priorización/analisis (ej. riesgo basado en description + dependencies).

Firebase NoSQL es flexible (esquema-less, queries denormalizadas), ideal para datos variables como tareas Scrum con dependencias dinámicas y logs crecientes. Reduce latencia en consultas frecuentes (ej. cargar tablero Kanban) y soporta alto volumen (2.8M transacciones/mes en Recaudación, según (Servicio de Rentas Internas (SRI) 2025). Firestore funciona con un enfoque basado en **documentos y colecciones**, donde:

- Las **colecciones** agrupan documentos similares (ej. "tasks" para todas las tareas).
- Los **documentos** son objetos JSON flexibles con campos dinámicos, sin esquema rígido.

- Las **relaciones** se manejan mediante referencias (ID de documentos), arrays embebidos o subcolecciones (para 1: N o N: N), priorizando lecturas rápidas y escalabilidad sobre joins complejos.
- **Ventajas:** Sincronización en tiempo real (ideal para colaboración Scrum), escalabilidad serverless y reglas de seguridad granulares por rol (RBAC).

Esto se alinea con los requerimientos del gestor: colaboración multiusuario, trazabilidad inmutable para auditorías (Webmaster 2025) y (Lexis 2026), priorización dinámica con IA (Gemini) y manejo de datos en procesos del SRI (Recaudación, Fiscalización, etc.). El modelo propuesto es un diagrama de esquema de colecciones Firestore, representado como jerarquía de documentos JSON con ejemplos. Este enfoque es denormalizado (datos duplicados para eficiencia) y optimizado para consultas frecuentes en backlogs y sprints.

1. Ajustes y Pruebas del Diseño Durante el avance del proyecto

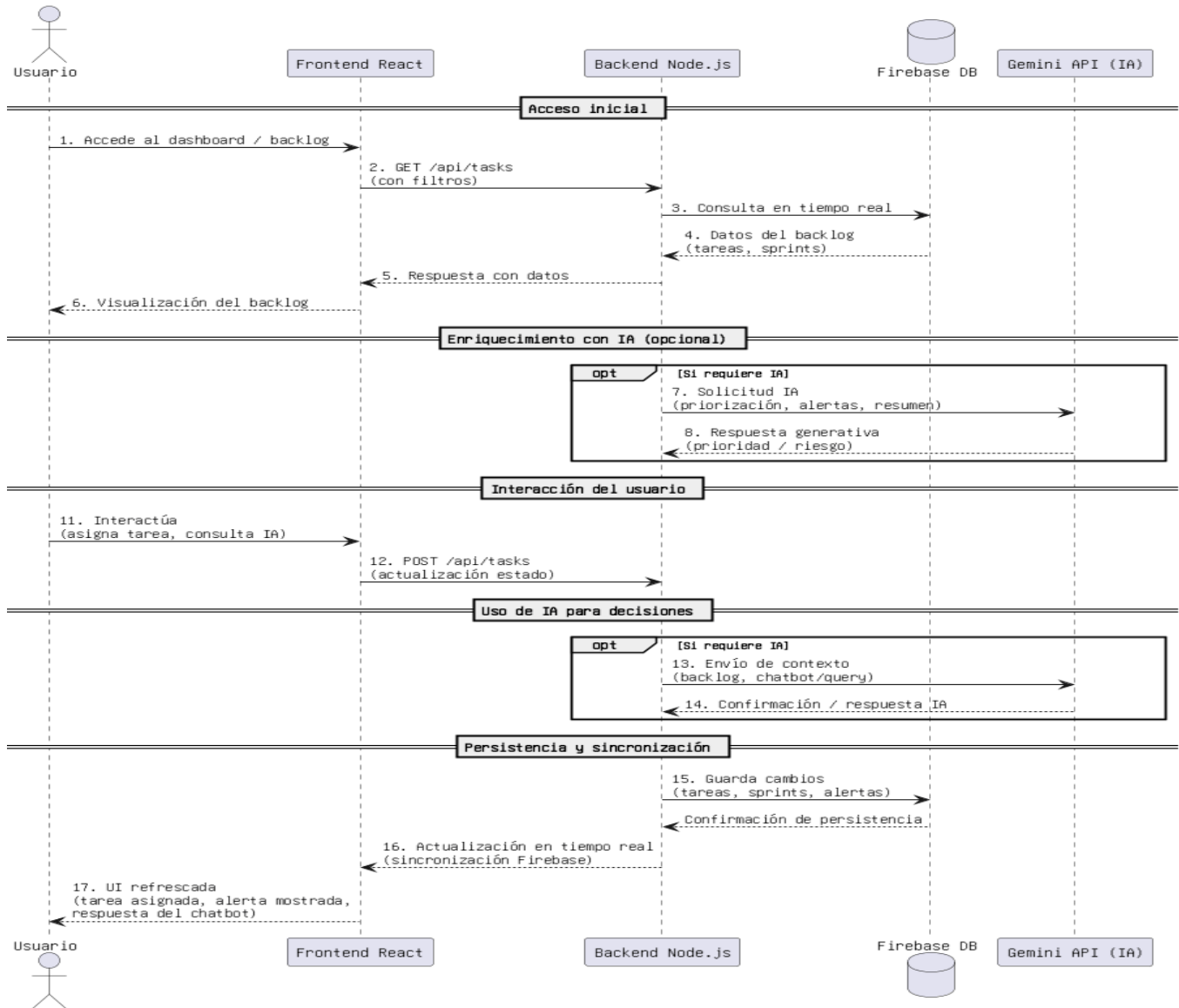
Durante el avance del proyecto Se realizó y puso a prueba el diseño inicial en comparación con la realidad de la implementación.

- Se confirmó que la API de Node.js cumplía con todas las necesidades de filtrado y carga de datos que requería la interfaz del usuario.
- Se constató que la configuración de las colecciones en Firebase No SQL era la correcta para reflejar todos los datos del negocio en tiempo real.
- Se garantizó que la arquitectura de capas permitía que tanto la parte visible como la oculta se desarrollaran por separado, agilizando así la programación en paralelo y su cuidado.

Diagrama de Secuencia

Figura 17:

Diagrama de secuencia



Nota: El gráfico representa la secuencia de interacción del sistema realizada, mostrando el flujo cronológico completo desde la acción del usuario (funcionario del SRI) hasta la respuesta inteligente del sistema, incluyendo la intervención de la API de Gemini para funcionalidades de IA generativa.

Descripción del diagrama de secuencia:

1. El Usuario (funcionario, Scrum Master, Product Owner o miembro del equipo) accede al sistema gestor de tareas, lo que provoca que el navegador cargue la página principal o el dashboard del backlog.
2. React + Tailwind (frontend), como frameworks de presentación, solicitan los datos del backlog y sprints activos al backend a través de una solicitud **GET /api/tasks** (o **GET /api/sprints** según la vista actual).

Ejemplo:

GET

/api/tasks?filters[status][\$ne]=Completada&populate=dependencias, sprint

3. El servidor backend (Node.js + Express/NestJS) recibe la petición, válida la autenticación (JWT) y los permisos (RBAC según rol Scrum), y realiza la consulta en tiempo real a Firebase Firestore.
4. Firebase Realtime Database / Firestore responde inmediatamente con los datos solicitados (lista de tareas, sprints, dependencias, estados actuales), gracias a su sincronización en tiempo real.
5. Node.js recibe los datos de Firebase y, si la operación requiere inteligencia (por ejemplo, priorización automática, cálculo de riesgos o generación de alertas), construye un prompt contextual y envía una solicitud a la **API de Gemini** (Google Generative AI).
 - Ejemplo de prompt: “Analiza estas 15 tareas del backlog del SRI. Priorízalas por riesgo tributario y urgencia normativa. Genera alertas para dependencias críticas.”
6. La API de Gemini procesa el prompt con los datos del backlog y devuelve una respuesta generativa en formato JSON:
 - Lista ordenada de tareas con nuevo score de prioridad

- Alertas predictivas de bloqueos o incumplimiento
7. Resumen ejecutivo o sugerencias accionables
 8. Node.js recibe la respuesta de Gemini, la enriquece con metadatos de auditoría (usuario, timestamp, acción) para trazabilidad normativa (Webmaster 2025) y (Lexis 2026) retornando el resultado completo al frontend vía la respuesta HTTP.
 9. React + Tailwind reciben los datos enriquecidos (backlog ordenado por IA, alertas, resúmenes), actualiza el estado local (usando Hooks + TanStack Query para caché y optimización), y renderiza la interfaz actualizada en tiempo real:
 - Muestra el backlog priorizado automáticamente
 - Presenta alertas visuales (rojo/amarillo) para riesgos detectados
 - Habilita el ChatBot para consultas adicionales (ej.: “¿Por qué esta tarea está bloqueada?”)
 10. El usuario visualiza el backlog inteligente, puede interactuar (asignar, comentar, mover entre sprints), y el ciclo se repite en tiempo real para cualquier acción (creación, actualización, consulta).

Este diagrama evidencia una arquitectura desacoplada, reactiva y altamente inteligente, donde cada componente cumple un rol específico:

- **React + Tailwind:** Interfaz rápida y responsiva para los usuarios del Servicio de Rentas Internas (SRI).
- **Node.js + Nest.js:** Orquesta la lógica de negocio, seguridad (RBAC) y llamadas a IA.

- **Firestore:** Proporciona sincronización en tiempo real y escalabilidad serverless.
- **Gemini:** Aporta inteligencia generativa (priorización, predicción, resúmenes, chatbot). Permitiendo escalabilidad, colaboración multiusuario y cumplimiento normativo, con mínima latencia y máxima trazabilidad. El flujo es eficiente y responde directamente a los dolores identificados en las entrevistas (demoras, falta de visibilidad, riesgos manuales, sobrecarga burocrática), convirtiendo la gestión de tareas en un proceso inteligente y proactivo.

Validación.

La validación del diagrama se realizó mediante revisión conjunta con el equipo de desarrollo y validación de requisitos del sistema. Esto permitió confirmar que todas las funcionalidades necesarias están correctamente representadas (CRUD de tareas, priorización inteligente con Gemini, trazabilidad para auditorías, sincronización en tiempo real con Firestore, y colaboración multiusuario según roles Scrum) y que no existen elementos redundantes o fuera del alcance del proyecto.

Se verificó específicamente:

- Que los flujos de consulta y actualización respetan RBAC (Role-Based Access Control) según roles del SRI (Product Owner, Scrum Master, Desarrollador).
- Que la integración con Gemini genera respuestas contextuales útiles y precisas (priorización por riesgo tributario, alertas predictivas, resúmenes de normativas).

- Que la sincronización en tiempo real de Firebase evita inconsistencias en colaboración distribuida.
- Que la trazabilidad (auditLog) cumple con LOPDP y requisitos de auditoría interna del SRI.

La validación incluyó pruebas de usuario con representantes del SRI (feedback de entrevistas), confirmando usabilidad, rendimiento y alineación con necesidades reales (reducción de demoras burocráticas, visibilidad de dependencias y automatización de tareas repetitivas).

Diseño del sistema.

El diseño del sistema tiene como propósito materializar los requisitos funcionales y no funcionales en una arquitectura técnica estable y componentes de software definidos de manera precisa. Este diseño garantiza que el gestor inteligente de tareas sea escalable, mantenible, seguro y cumpla con los objetivos de rendimiento, disponibilidad y cumplimiento normativo planteados para el SRI.

Arquitectura del Sistema

El gestor inteligente de tareas seguirá una arquitectura de aplicación web moderna basada en capas desacopladas, separando claramente el frontend del backend y utilizando APIs RESTful como medio principal de comunicación entre componentes. Esta estructura aprovecha servicios en la nube (Firebase) y la integración estratégica de APIs de inteligencia artificial (Gemini) para garantizar sincronización en tiempo real, procesamiento inteligente y cumplimiento normativo en el contexto del SRI del Ecuador.

La arquitectura planteada se organiza en las siguientes capas principales:

Capa de Presentación (Frontend): Desarrollada con React + Tailwind CSS. Esta será una aplicación web adaptable (responsive) que funcionará directamente en el navegador del usuario. Se enviarán solicitudes HTTP a la API RESTful del backend para establecer comunicación fluida y obtener datos en tiempo real. React gestiona la interfaz dinámica del sistema: dashboard de métricas de sprint, tablero Kanban (Por Hacer, En Progreso, Revisión, Hecho), detalle de tareas con diagrama de dependencias, panel de notificaciones proactivas y widget del chatbot TaskiBot. Tailwind CSS asegura un diseño limpio, moderno y altamente personalizable, con tema oscuro institucional del SRI y componentes accesibles (Shadcn/UI).

Figura 18:

Capa de presentación del proyecto

```

1 import React, { useState, useEffect, useRef } from 'react';
2 import {
3   LayoutDashboard,
4   KanbanSquare,
5   Users,
6   Settings,
7   Menu,
8   Bell,
9   Search,
10  Logout,
11  FolderKanban,
12  Moon,
13  Sun,
14  X,
15  Check,
16  Mail,
17  User as UserIcon,
18  Shield,
19  Globe,
20  Save,
21  ClipboardList,
22  Calendar
23 } from 'lucide-react';
24 import Dashboard from './components/Dashboard';
25 import ScrumBoard from './components/ScrumBoard';
26 import AIChatbot from './components/AIChatbot';
27 import Projects from './components/Projects';
28 import Team from './components/Team';
29 import PersonalNotes from './components/PersonalNotes';
30 import CalendarView from './components/CalendarView';
31 import Login from './components/Login';
32 import Register from './components/Register';
33 import { authService } from './services/auth';
34 import { firebaseAuthService } from './services/firebaseAuth';
35 import { tasksService, sprintsService, projectsService, usersService, migrationService } from './services/firebaseData';
36 import { FIREBASE_ENABLED } from './services/firebase';
37 import { MOCK_TASKS, MOCK_USERS, MOCK_SPRINTS, MOCK_PROJECTS } from './constants';
38 import { Task, TaskStatus, Project, User, Role, Sprint } from './types';

```

Nota: En la gráfica se puede evidenciar la capa del desarrollo del proyecto.

Capa de Lógica de Negocio y Gestión de Contenidos (Backend): Implementada utilizando Node.js + Express/NestJS como framework principal. Este componente expone una API RESTful bien definida y estructurada que será consumida por el frontend. Se encarga de toda la lógica de negocio del sistema, incluyendo:

Validaciones de seguridad y permisos (RBAC según roles Scrum: Product Owner, Scrum Master, Desarrollador, Auditor Normativo).

Procesamiento de operaciones CRUD sobre tareas, sprints y usuarios.

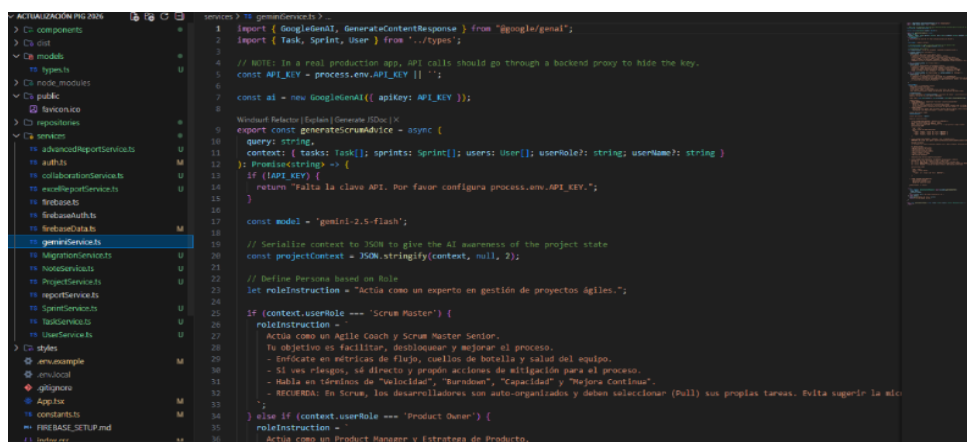
Orquestación de llamadas a la API de Gemini para funcionalidades inteligentes (priorización automática por riesgo tributario, alertas predictivas de bloqueos, resúmenes ejecutivos y respuestas conversacionales del chatbot).

Enriquecimiento de datos con trazabilidad inmutable (logs de auditoría para cumplimiento LOPDP, COT y COESC).

Gestión de sincronización y eventos en tiempo real con Firebase.

Esta capa actúa como núcleo inteligente del sistema, transformando datos crudos en acciones proactivas y asegurando que cada interacción cumpla con los requisitos normativos y de auditoría del SRI.

Figura 19: Capa lógica del proyecto



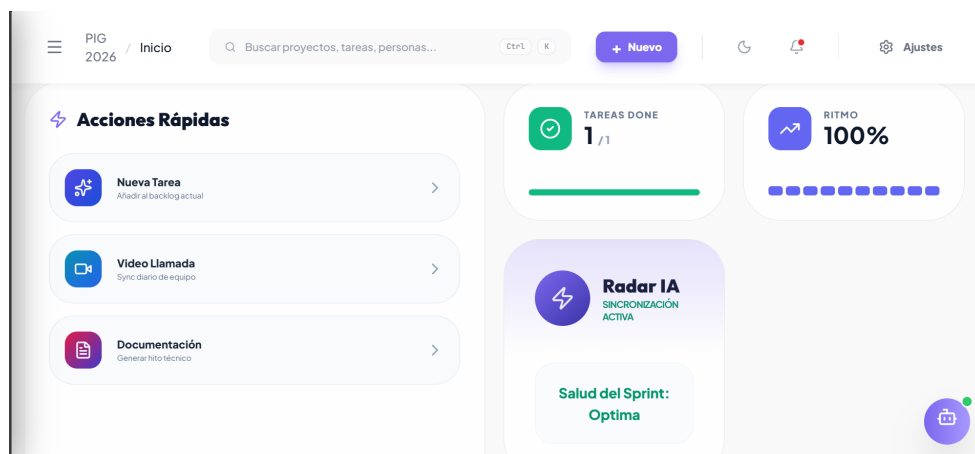
Nota: El gráfico representa la estructura principal del backend (Node.js) que orquesta la API RESTful y la integración con Gemini y Firebase.

Esta arquitectura desacoplada permite:

- Escalabilidad horizontal (Firebase serverless + Node.js ligero).
- Mantenibilidad (módulos independientes y tipado fuerte con TypeScript).
- Seguridad reforzada (JWT + RBAC + reglas Firebase).
- Inteligencia integrada (Gemini como capa de IA generativa para priorización, predicción y chatbot).
- El diseño asegura que el gestor inteligente de tareas sea una solución robusta, adaptable y alineada con la transformación digital institucional del SRI (Plan Estratégico 2025-2029 y Agenda de Transformación Digital extendida hasta 2030), superando las limitaciones actuales de visibilidad, colaboración y gestión manual de riesgos normativos.

Diseño de la Interfaz de Usuario (UI/UX)

Figura 20: *Diseño de la interfaz de usuario (ui/ux)*



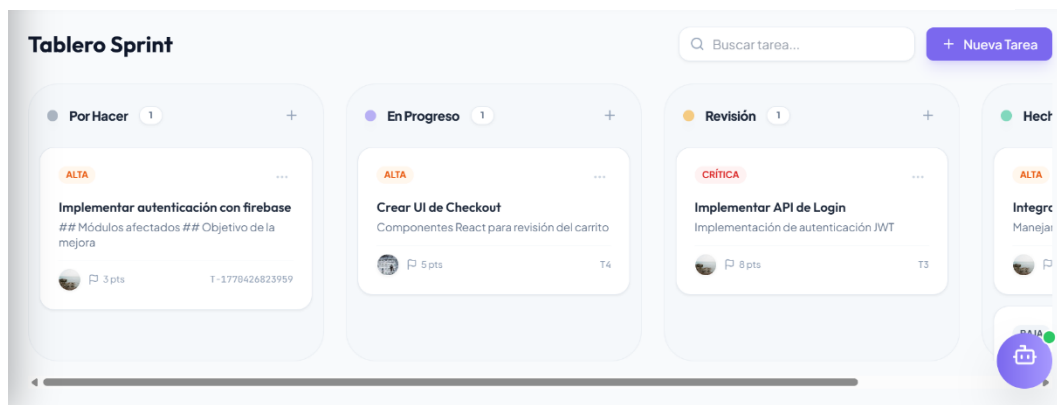
Nota. Se priorizará la usabilidad y la experiencia del usuario, especialmente para un público institucional técnico-tributario (funcionarios del SRI, Scrum Masters, Product Owners, desarrolladores TIC). Se utilizará el framework Tailwind CSS junto con componentes de Shadcn/UI para garantizar una interfaz limpia, moderna y responsiva, con integración de alertas de Gemini para una interacción proactiva.

Figura 21: *Chatbot asistente*

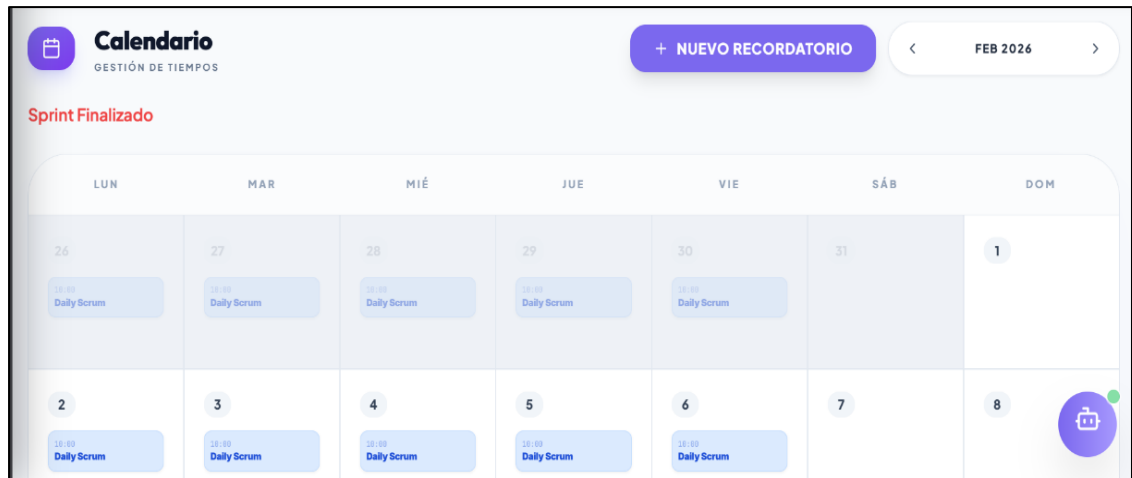


Nota. Representación de la ventana del chatbot con el menú de comandos.

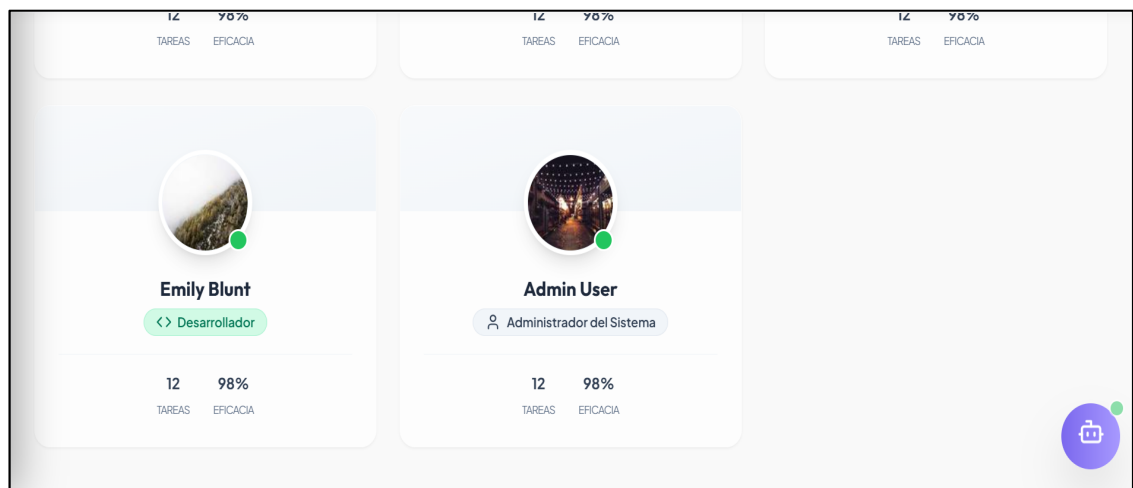
Figura 22 *Tablero sprint de actividades*



Nota. La captura representa al tablero de sprints o actividades que tiene cada usuario del equipo Scrum dependiendo del rol con el estado de cada actividad. El tablero de sprints (“Tablero Sprint”) es el componente central del gestor inteligente de tareas, diseñado como un Kanban ágil adaptado al contexto institucional del SRI. Muestra columnas dinámicas (Por Hacer, En Progreso, Revisión, Hecho) con tarjetas de tareas que incluyen título, prioridad (sugerida por Gemini), riesgo normativo (badge rojo/amarillo), asignado (avatar + rol Scrum), tiempo estimado y etiquetas contextuales (ej. [CRÍTICA], [SEGURIDAD], [NORMATIVA]). El diseño responde directamente a los dolores identificados en entrevistas (preguntas 3, 5, 6, 7 y 9): falta de visibilidad en dependencias y bloqueos, demoras por burocracia, sobrecarga en validaciones manuales y necesidad de trazabilidad para auditorías (Webmaster 2025).

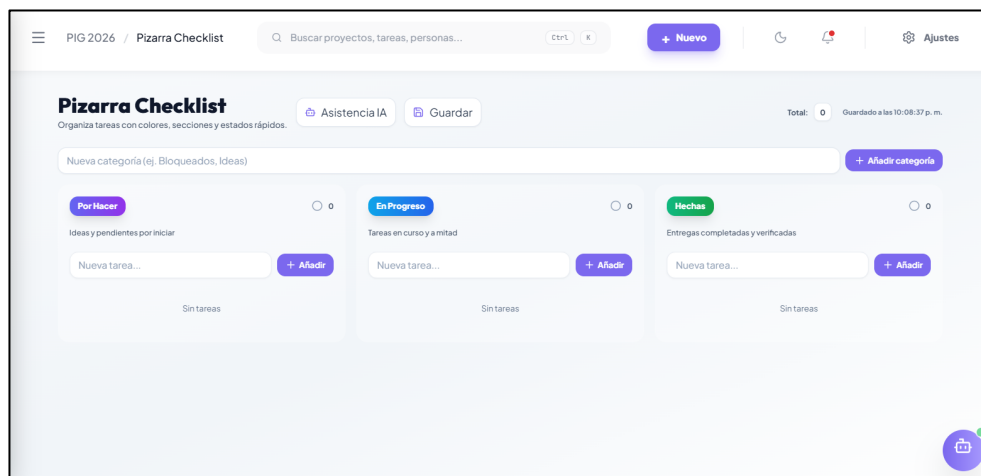
Figura 23:*Calendario en tiempo real con recordatorios*

Nota. Calendario en tiempo real con el recordatorio de las Daily Scrum diarias.

Figura 24:*Gestor de usuarios de la aplicación*

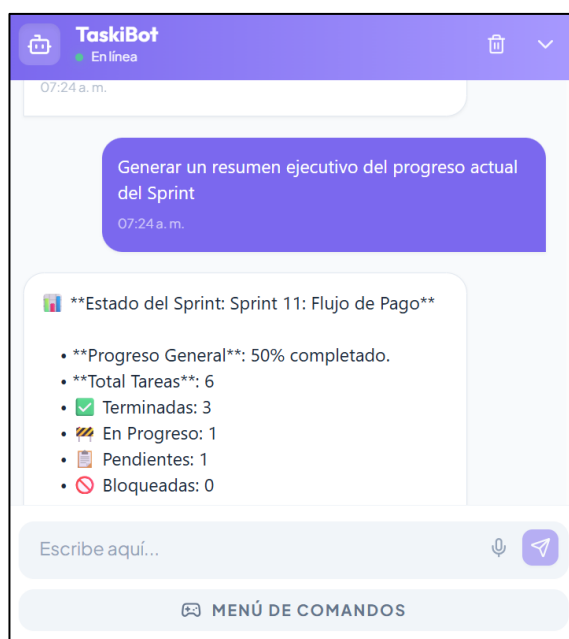
Nota. Captura del gestor de usuarios del equipo Scrum con las descripciones: rol, tareas y porcentaje de eficiencia.

Figura 25: Pizarra Checklist para agregar notas personales de cada usuario

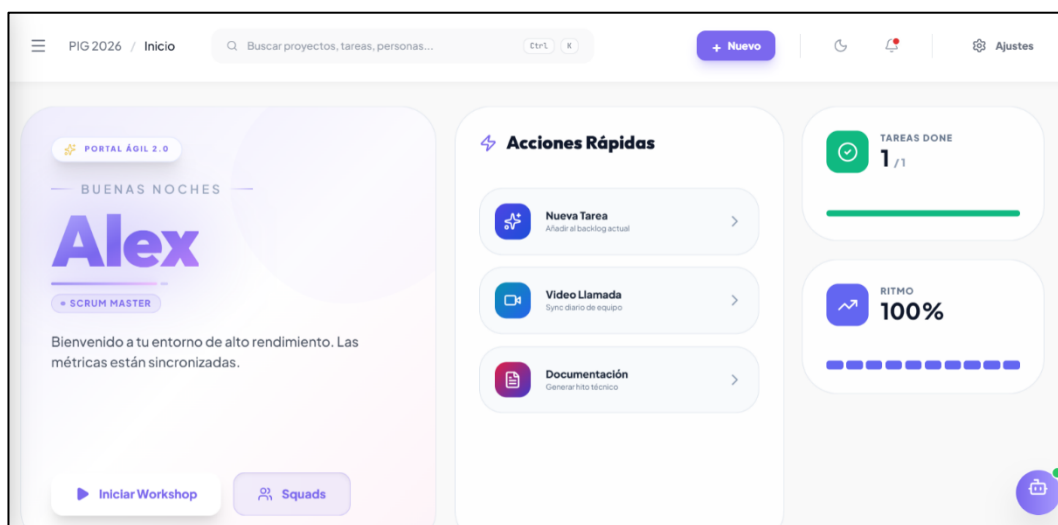


Nota. Representa la visualización de notas que contiene la función de un bloc de notas para adicionar apuntes personales además de como añadir recordatorios o tareas pendientes y también algún comentario de ayuda para sus actividades.

Figura 26: Formulario de reportes



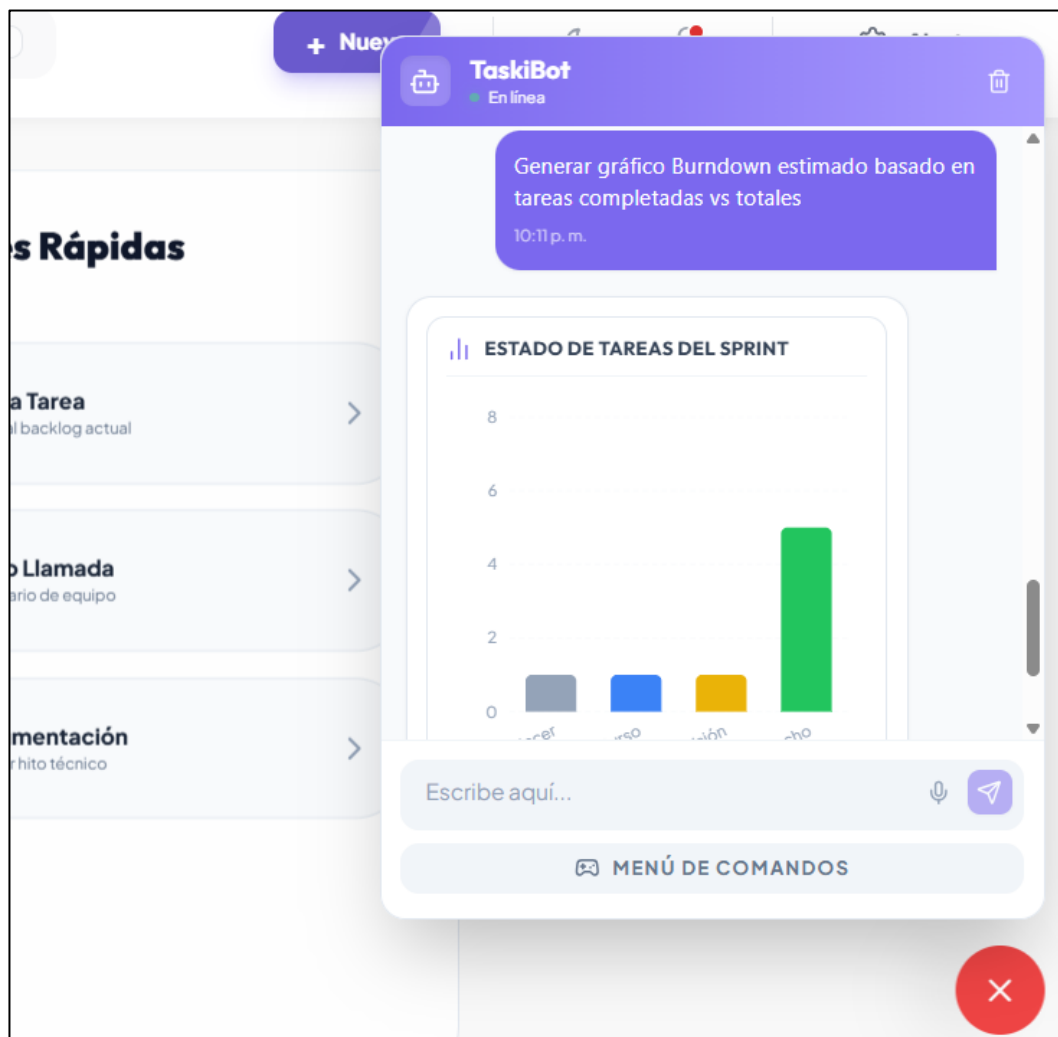
Nota. Representación del formulario de reportes (disponible en el rol de programador) para notificar al Scrum Master para obtener un mejor seguimiento y comunicación de cada sprint o actividad trabajada o con bloqueo

Figura 27:*Panel de control del Scrum Master*

Nota. Representa la vista del panel de control o dashboard (disponible para el scrum master), que contiene las opciones o privilegios: progreso del proyecto, adicionar historias de usuario o refinar el backlog y revisar los proyectos activos. Así también incluye la función de gestionar los usuarios a cargo de cada proyecto, hacer cambios en el proyecto entre otras.

Figura 28:

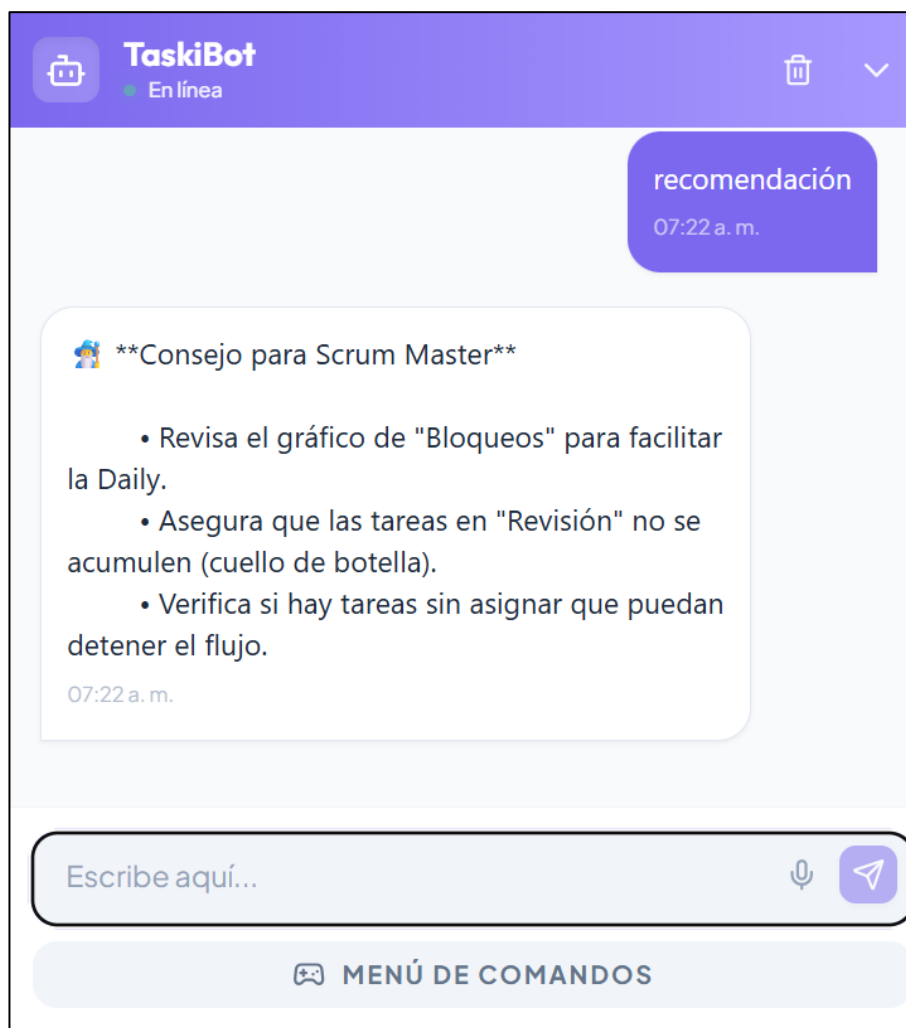
Generación gráfica del nivel de riesgo del proyecto



Nota. Se visualiza la funcionalidad del chatbot aplicando inteligencia artificial generativa para graficar a modo de un gráfico de barras la representación del reporte del Sprint o Actividad que adicional a esto, permite al usuario exportar en formato PNG.

Figura 29:

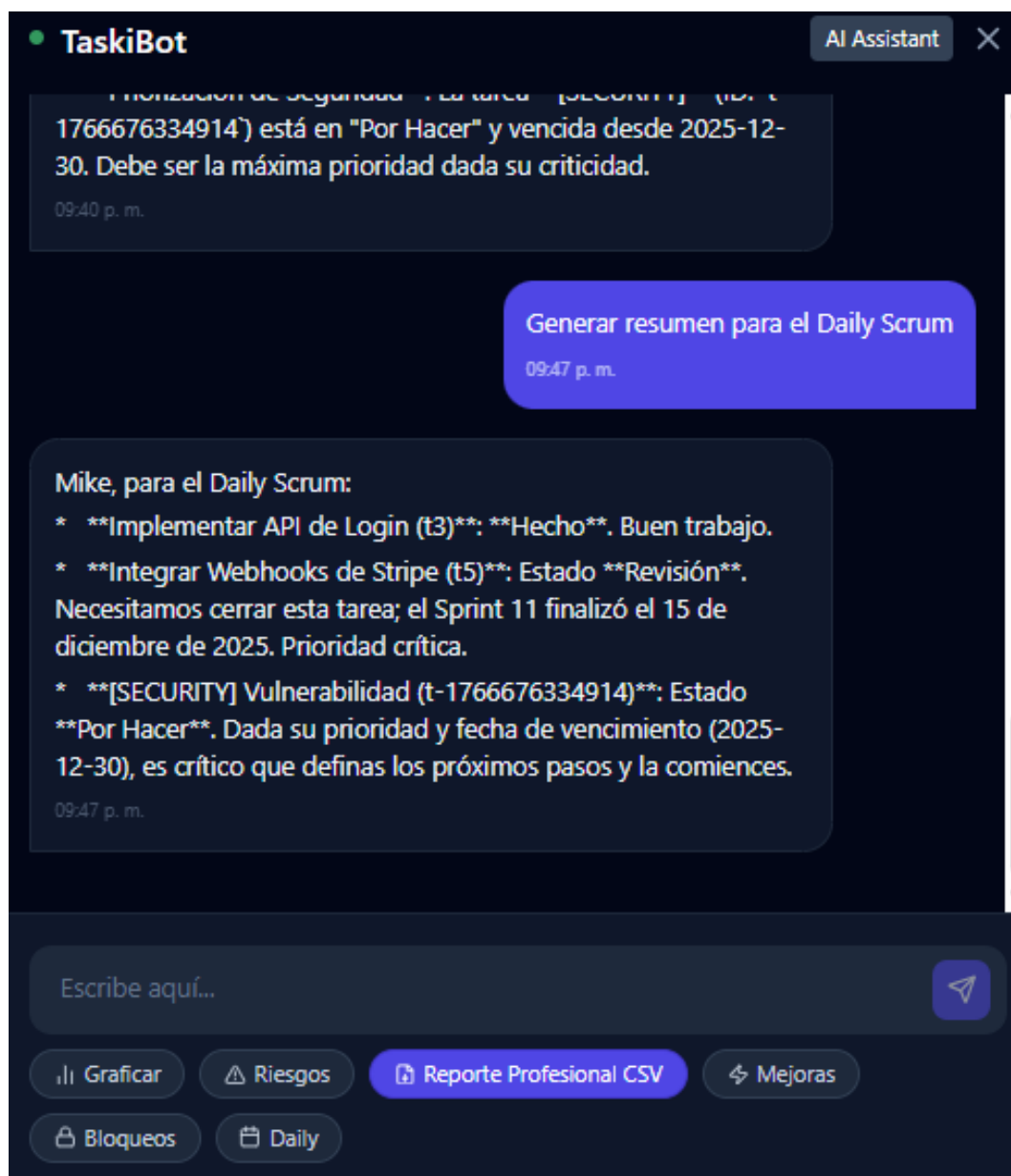
Generación de recomendaciones para cada sprint o actividad dependiendo de cada rol



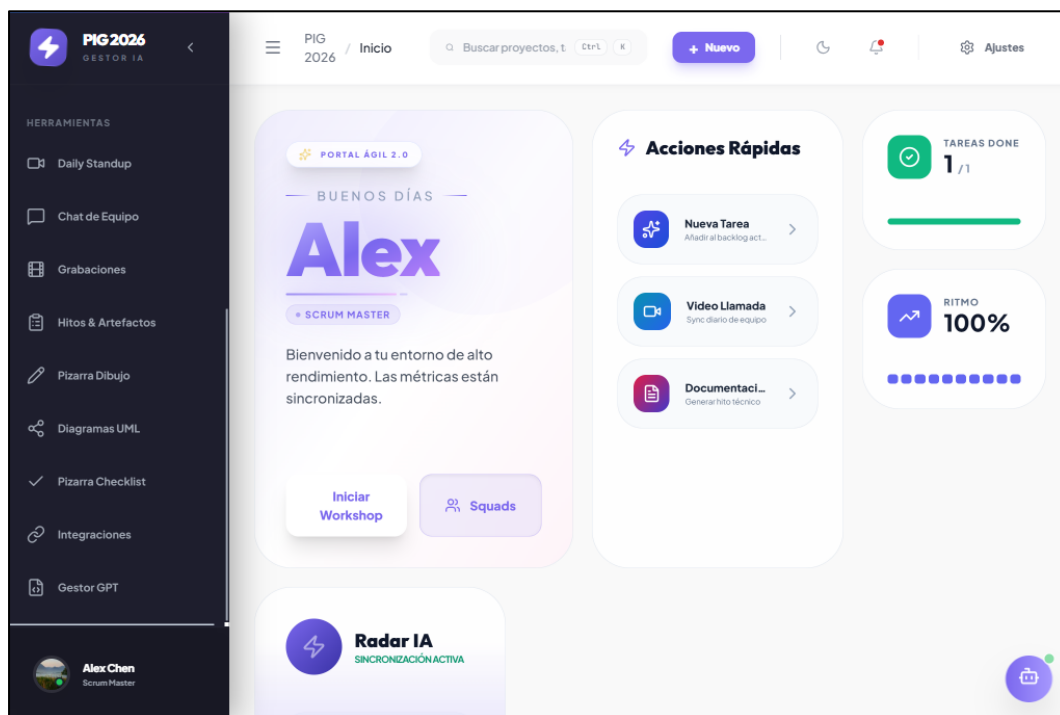
Nota. Representa la generación de texto con el fin de mejorar a cada sprint o actividad que se encuentre en fase de implementación. El chatbot aplica inteligencia artificial generativa y agentica para que el programador cuente con una asistencia mucho más personalizada dependiendo de la actividad que esté desempeñando.

Figura 30:

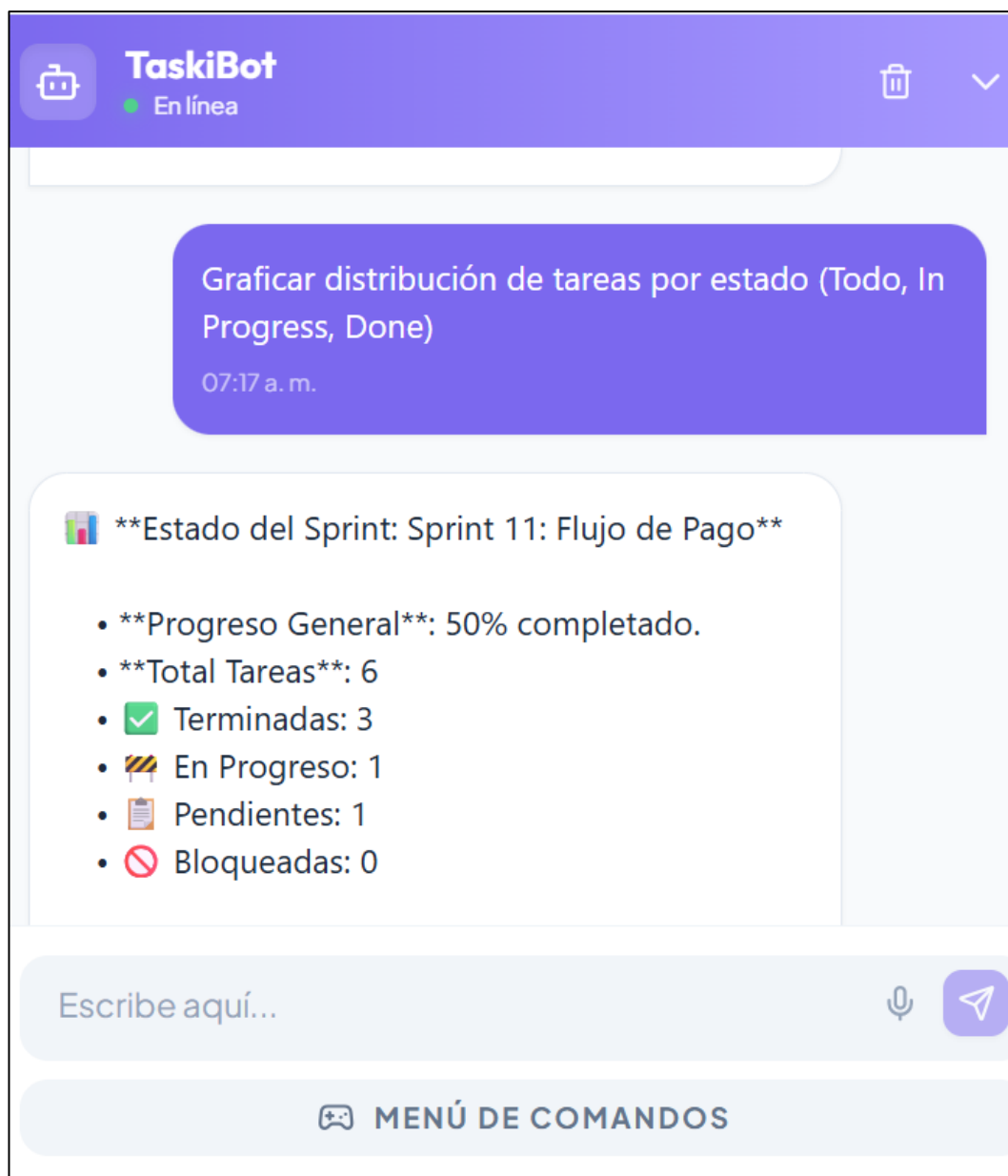
Generación del resumen de cada reunión



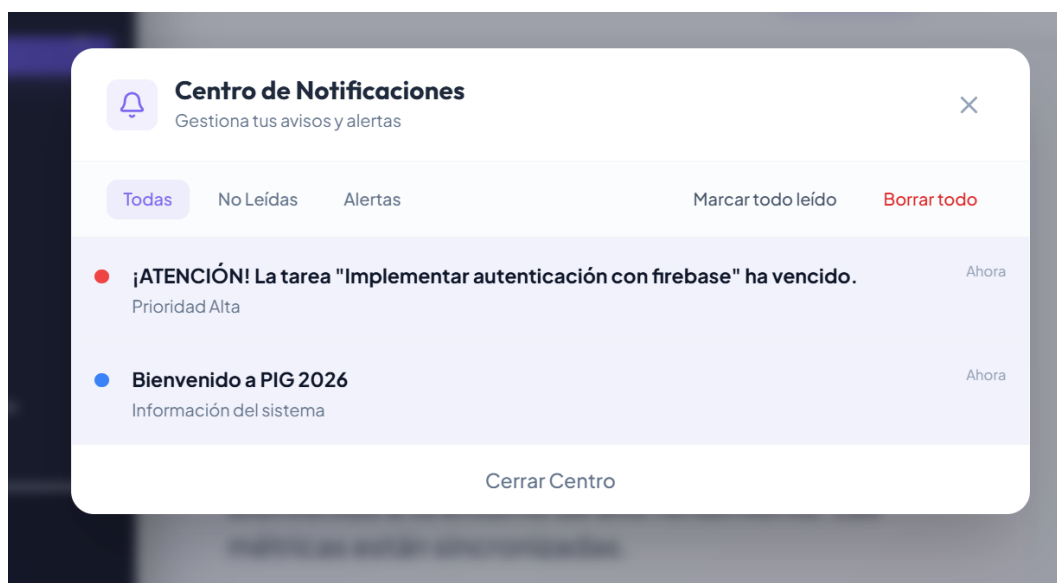
Nota. La vista representa la generación del resumen de la reunión Scrum que se haya tenido presente con el fin de exponer el estado de cada sprint, la retroalimentación obtenida, las fechas y priorización de los sprints. Al igual que sus vencimientos y recomendaciones a tomar en cuenta para próximos sprints.

Figura 31:*Tablero de control del Scrum Master*

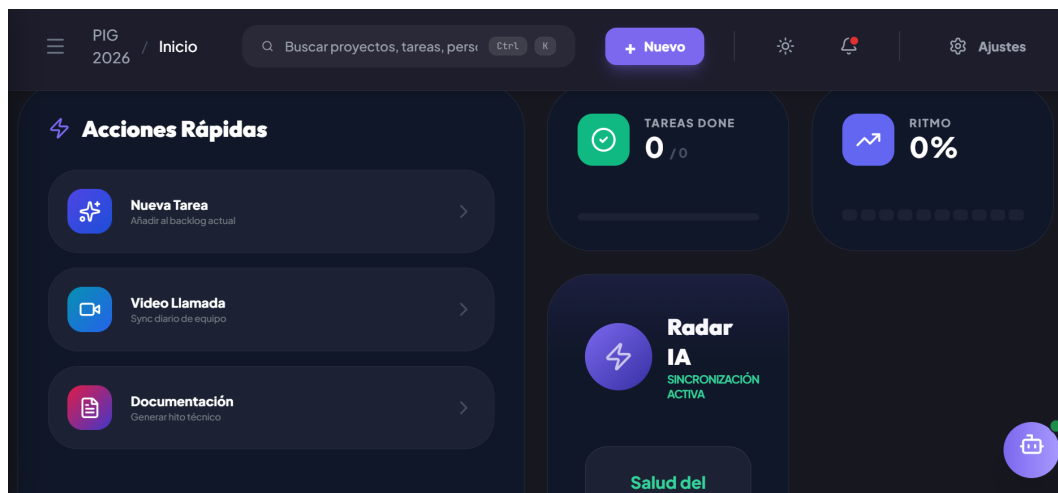
Nota. Representa la vista del tablero de control para el Scrum Master con las funcionalidades y privilegios que corresponden al rol basándose en la normativa y estándares del marco de trabajo.

Figura 32:*Generación de consultas*

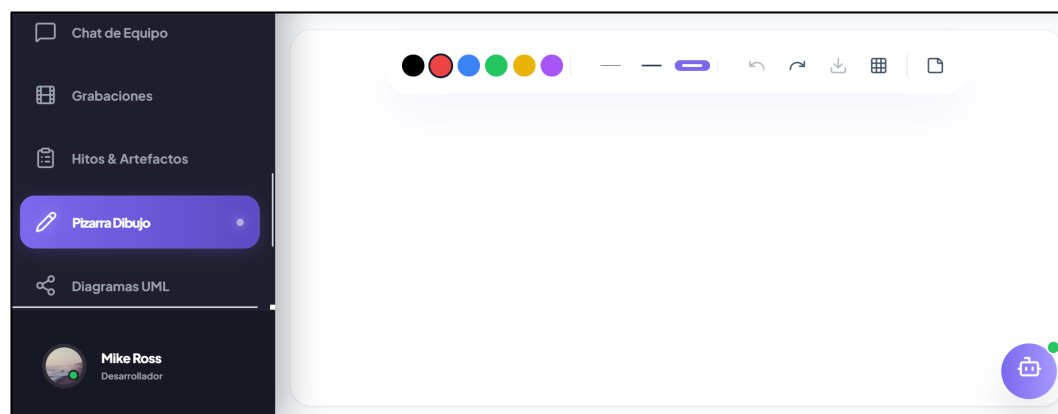
Nota. Representación de la vista del chatbot generando consultas sobre cuáles son las notas personales, ya que tiene acceso a la base de datos puede consultar y generar una respuesta.

Figura 33:*Panel de notificaciones*

Nota. Visualización del panel o ventana de modal con las notificaciones leídas o sin leer entre otras opciones en la cual se exponen las actividades y sprints con sus descripciones. El panel de notificaciones (“Centro de Notificaciones”) es un componente modal flotante que centraliza avisos y alertas en tiempo real del sistema, diseñado para reducir la sobrecarga cognitiva y mejorar la visibilidad de eventos críticos en equipos Scrum del SRI. Este panel responde directamente a los dolores identificados en entrevistas (preguntas 3, 5, 6, 7 y 9): falta de visibilidad en dependencias y bloqueos, demoras por no detectar riesgos normativos a tiempo, y necesidad de trazabilidad para auditorías internas y externas (Webmaster 2025).

Figura 34:*Modo oscuro*

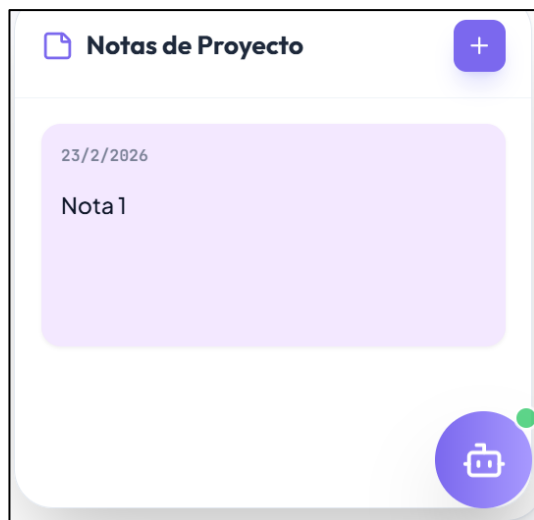
Nota. Representación de la funcionalidad de modo oscuro recomendable activarla en horarios nocturnos y proteger la visión.

Figura 35*Pizarra de dibujo para los usuarios*

Nota. Representa a la pizarra de dibujo para realizar ilustraciones mediante el cursor del mouse y también tiene la opción de pizarra cuadriculada y para descargar cada ilustración.

Figura 36

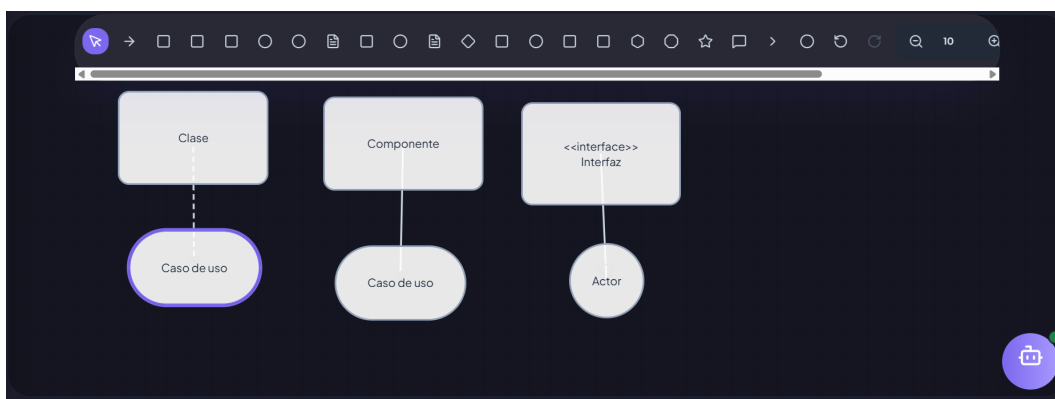
Funcionalidad de añadir notas personales



Nota. Función para adicionar notas personales de apoyo y mediante el chatbot se les puede dar seguimiento con consultas.

Figura 37

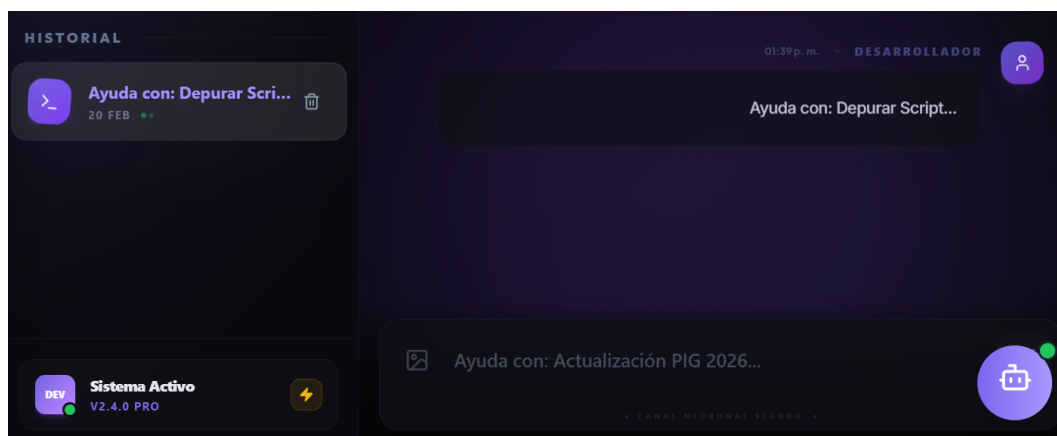
Pizarra Diagramación UML



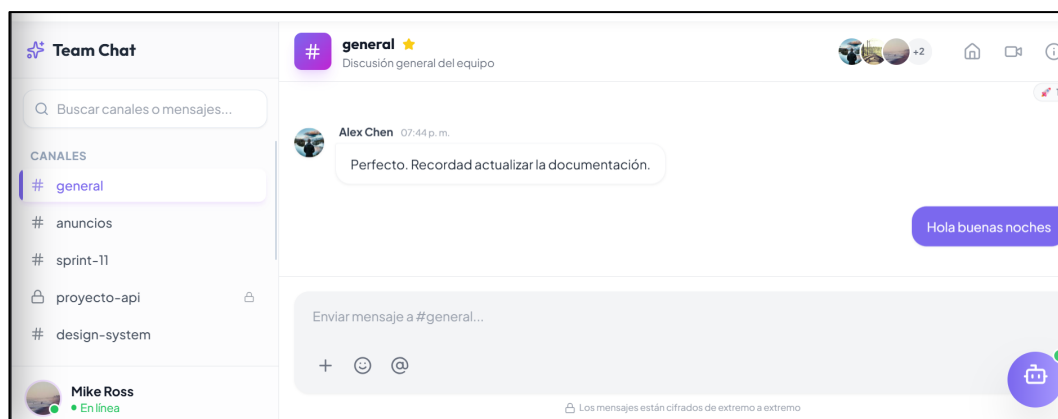
Nota. Representación de la opción para diagramar en UML con múltiples opciones y las opciones para guardar, borrar y exportar los diagramas. También se comidero adicionar una opción para que la pizarra sea cuadriculada.

Figura 38*Conexión de servicios herramientas externas*

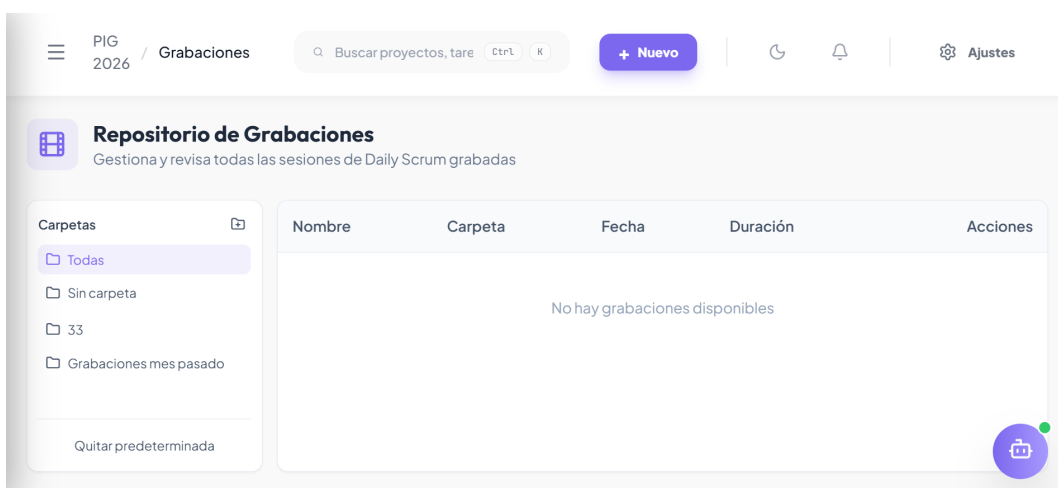
Nota. Funcionalidad para conectar la app con herramientas externas como: Google Sheets, Slack, Jira y Trello.

Figura 39*Gestor GPT*

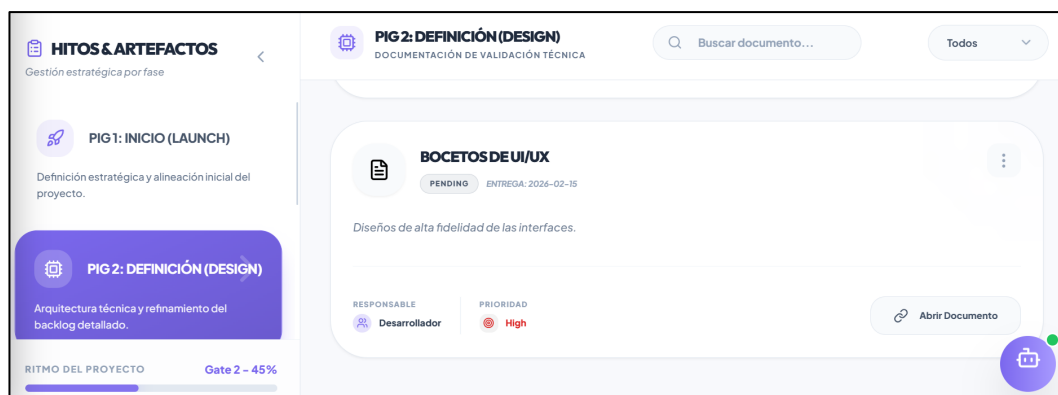
Nota. Gestor GPT potenciado con Gemini, que sirve de apoyo para la resolución de errores, análisis de documentación, depuración entre otros más. Cuenta con el acceso a documentación de repositorios como StackOverflow y GitHub con el fin de tener respuestas más optimas.

Figura 40*Chat grupal en tiempo real*

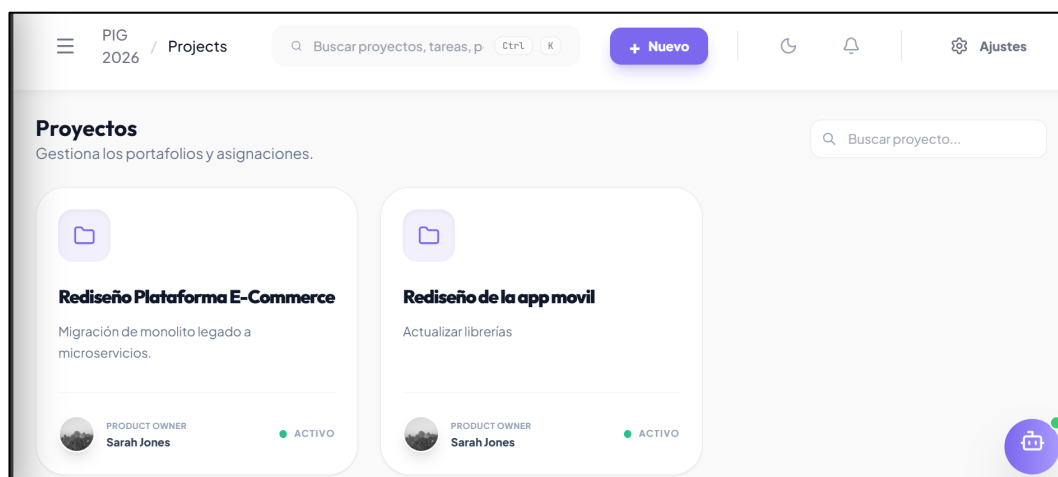
Nota. Chat en tiempo real grupal para todos los miembros Scrum con el fin de tener mejor retroalimentación. Así también es posible subir documentación para ser analizada, como también iniciar videollamadas o conectarse directamente a las Daily Scrum.

Figura 41*Repositorio de grabaciones*

Nota. Repositorio de grabaciones con la función para crear directorios y también el chatbot tiene la opción de acceder a cada grabación para la obtención de un resumen personalizado para quienes no sea posible asistir a la reunión.

Figura 42*Hitos y artefactos en tiempo real*

Nota. Hitos y artefactos de las gestiones estratégicas con sus respectivas fases de desarrollo e implementación de cada proyecto, tomando en cuenta el ciclo de vida de desarrollo.

Figura 43*Gestor de proyectos y su estado*

Nota. Gestor de proyectos para revisar el estado y también tiene privilegios de usuario para que el Scrum Master pueda modificar los requerimientos y asignar sprints o nuevas actividades para los desarrolladores.

Figura 44:*Generación de reportes en formato Excel*

	A	B	C	D	E	F	G	H
4								
5	INFORMACIÓN DEL SPRINT							
6	Sprint	Sprint 11: Flujo de Pago						
7	Objetivo	Implementar integración con Stripe						
8	Periodo	30 Nov - 14 Dec						
9	Fecha de Generación	20 de febrero, 2026 21:48						
10								
11	INDICADORES CLAVE (KPIs)							
12	Progreso del Sprint	33.3%						
13	Puntos Completados	7 / 28		EN PROGRESO				
14	Velocidad Actual	7 pts						
15	Tareas en Riesgo		1	⚠️ ATENCIÓN				
16								
17	CENTRO DE INTELIGENCIA							
18	Velocidad Requerida	21.0 pts/día						
19	Tareas Bloqueadas		0	LIBRE				
20	Días para Cierre	0 días						
21	Pendiente Crítico		1	PRIORIZAR			PRODUCTIVIDAD POR MIEMBRO (Puntos)	
22								
23	ALERTAS Y RECOMENDACIONES							
24	⚠️ CRÍTICO: Hay 1 tareas vencidas. Afecta el compromiso del Sprint.							
25	🔗 ACCIÓN: Considerar reducir el alcance (Scope) del Sprint negociando con el Product Owner.							
26								
27	DISTRIBUCIÓN POR ESTADO							
28	Estado	Tareas		Impacto				

Nota. Visualización del archivo .CSV con el reporte de cada proyecto. El mismo que fue exportado desde el chatbot que se generó a partir de una indicación en las opciones disponibles. Se enfoca en cómo el sistema (desarrollado con React junto con Tailwind CSS en frontend, Node.js en backend, Firebase Realtime Database / Firestore para datos, y Gemini para IA generativa) lleva a cabo esta funcionalidad. Se basa en la arquitectura de capas desacopladas estudiada: frontend interactivo, backend orquestador, datos en tiempo real y IA proactiva. Esto responde directamente a necesidades identificadas en entrevistas (visibilidad de backlogs, trazabilidad para auditorías, reducción de reportes manuales) y alinea con el Plan Estratégico SRI 2025-2029 (Servicio de Rentas Internas (SRI) 2025) (eficiencia operativa en procesos tributarios) y normativas como (Webmaster 2025) (trazabilidad inmutable).

Figura 45

Reporte personalizado en formato PDF



Nota. Reporte personalizado para el miembro Mike en formato PDF generado a partir de una instrucción dada al chatbot en el menú de comandos.

Figura 46

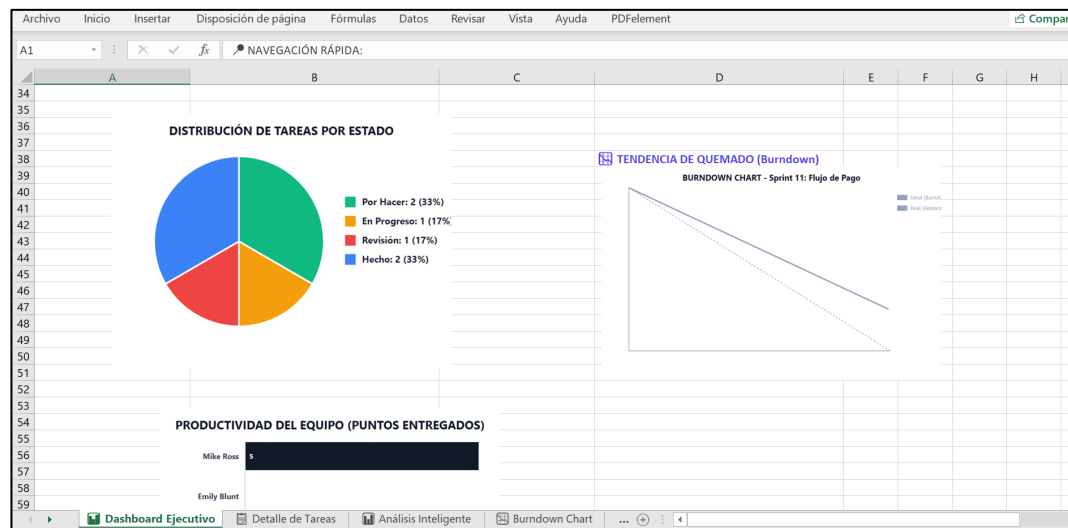
Tabla dinámica del reporte Excel generado

Archivo Inicio Insertar Disposición de página Fórmulas Datos Revisar Vista Ayuda PDFelement										
B26										
	A	B	C	D	E	F	G	H	I	J
1	NAVEGAC	Dashboard	Tareas	Análisis	Burndown	Equipo				
2										
3	ID	Título	Estado	Prioridad	Puntos	Responsable	Fecha Límite			
4	t3	Implementar API de Login	Por Hacer	Critica	8	Mike Ross	N/A			
5	t4	Crear UI de Checkout	En Progreso	Alta	5	Emily Blunt	N/A			
6	t5	Integrar Webhooks de Stripe	Revisión	Alta	5	Mike Ross	N/A			
7	t6	Actualizar UI Perfil Usuario	Hecho	Baja	2	Emily Blunt	N/A			
8	t-17666763: [SECURITY]		Hecho	Critica	5	Mike Ross	29/12/2025			
9	t-17704268: Implementar autenticación con firebase		Por Hacer	Alta	3	Mike Ross	04/02/2026			
10	Total				28					
11										
12										
13										
14										
15										

Nota. Tabla generada con la instrucción del reporte en el menú de comandos.

Figura 47

Gráficos de barra y pastel reporte Excel



Nota. Gráficos de barras, pastel y tendencia generados con la instrucción al chatbot en el menú de comandos.

Funcionalidades para Llevar a Cabo la Generación y Exportación del Reporte

- **Interfaz de Usuario y Activación desde el Chatbot (Frontend - React):**
 - ✓ **Descripción:** El usuario inicia la generación del reporte mediante una consulta natural en el TaskiBot (ej. “Genera un reporte profesional CSV del sprint actual con métricas de progreso y riesgos”). El chatbot, integrado como widget flotante en el dashboard o tablero Kanban, usa React Hooks (useState para estado de conversación, useEffect para llamadas asíncronas) y TanStack Query para manejar el estado de la petición sin recargar la página.

Figura 48:

Panel de control del administrador



Nota. Visualización del panel de control (disponible para el rol de administrador). Contiene privilegios y funcionalidades para tener un mayor control y gestionar de mejor manera a los usuarios para futuras auditorias o mantenimientos.

Diseño de la API (Endpoints Principales)

El diseño de la API en este proyecto sigue una arquitectura Service-Repository (Servicio-Repositorio) que actúa como intermediario entre la interfaz de usuario (React) y la base de datos (Firebase Firestore / LocalStorage).

No es una API REST tradicional con URLs (como /api/users), sino una API de Servicios TypeScript que expone métodos asíncronos para interactuar con los datos.

1. Arquitectura General

- **Frontend (Componentes):** Llama a los servicios para pedir datos.
- **Capa de Servicios (/services):** Contiene la lógica de negocio y valida datos antes de enviarlos.
- **Capa de Repositorios (/repositories):** Se encarga de la comunicación directa con la base de datos (Firestore).
- **Base de Datos:** Firebase Firestore (Colecciones de documentos).

2. Endpoints Principales (Métodos de Servicio)

Aquí están los "endpoints" o funciones principales disponibles para consumir:

- Autenticación (authService)
- ✓ Maneja el registro y validación de usuarios. Actualmente tiene una implementación híbrida (LocalStorage con hash de seguridad).
- ✓ register (name, email, password, role): Crea un nuevo usuario con contraseña encriptada.

- ✓ `validatePasswordStrength(password)`: Valida requisitos de seguridad (longitud, mayúsculas, etc.).
- ✓ `init ()` : Inicializa el sistema de seguridad y migra usuarios si es necesario.
- Tareas (`tasksService`)
 - ✓ Gestiona el flujo de trabajo principal. Soporta tiempo real.
 - ✓ `getAll ()`: Obtiene todas las tareas del sistema.
 - ✓ `getBySprintId(sprintId)`: Filtra tareas específicas de un Sprint.
 - ✓ `add(task)`: Crea una nueva tarea.
 - ✓ `update (taskId, updates)` : Actualiza estado, asignación o detalles.
 - ✓ `delete(taskId)`: Elimina una tarea permanentemente.
 - ✓ `onTasksChange(callback)`: Suscripción Real-Time (Escucha cambios en vivo sin recargar).
- Proyectos (`projectsService`)
 - ✓ Administración de los proyectos globales.
 - ✓ `getAll ()` : Lista todos los proyectos activos.
 - ✓ `add(project)`: Registra un nuevo proyecto.
 - ✓ `update (projectId, updates)`: Modifica detalles del proyecto.
 - ✓ `delete(projectId)`: Elimina un proyecto.
 - ✓ `onProjectsChange(callback)`: Suscripción a cambios en tiempo real.
- Usuarios (`usersService`)
 - ✓ Gestión de perfiles y roles de equipo.
 - ✓ `getAll ()`: Obtiene la lista de todos los miembros del equipo.
 - ✓ `getById(userId)`: Obtiene detalles de un usuario específico.

- ✓ `update(userId, updates)` : Actualiza información de perfil.
- ✓ `delete(userId)`: Da de baja a un usuario.

3. Modelo de Datos (Firestore)

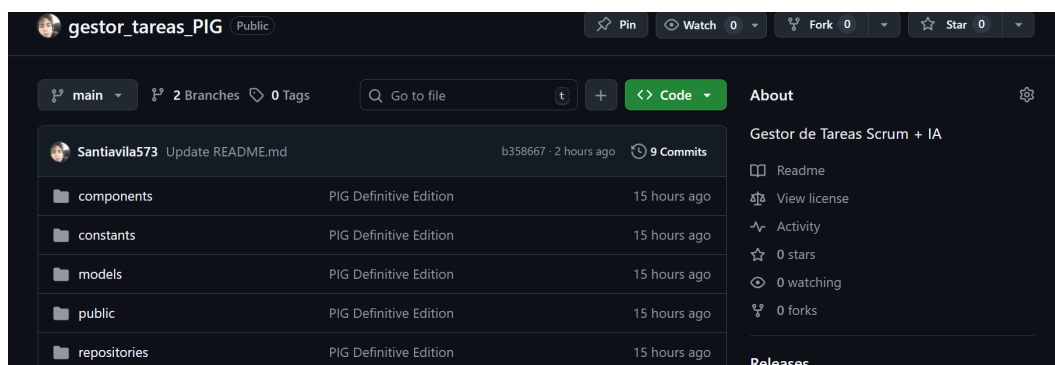
La API interactúa con las siguientes colecciones principales en la base de datos:

- `tasks`: Almacena cada tarea con su estado, asignado y sprint.
- `projects`: Contenedores de alto nivel para el trabajo.
- `users`: Perfiles de usuario y roles.

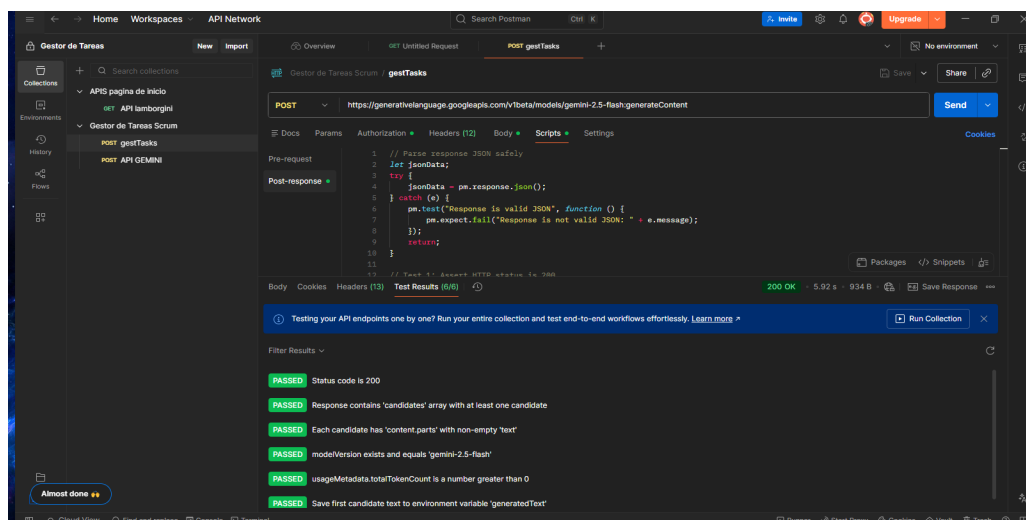
Desarrollo y despliegue.

La creación de la plataforma se realizó utilizando un entorno web actual y avanzado. Lo que se utilizaron las siguientes herramientas y versiones:

- Lenguaje de Programación: TypeScript (~5.0)
- Entorno de Ejecución: Node.js (v18.x o superior)
- Framework Frontend: React.js 19
- Backend: Node.js v25.5.0
- Base de Datos: Firebase (Google) NoSQL
- Control de Versiones: Git, hospedado en GitHub
- Pruebas API REST: Postman
- Plataforma de Despliegue Frontend: Netlify
- Plataforma de Despliegue Backend: Un servicio compatible con Node.js (Render, Railway, etc.)

Figura 49:*Repositorio de GitHub*

Nota. Representación del proyecto alojado en un repositorio en GitHub con todas las indicaciones y guía para su instalación y utilización (Ávila [2025] 2025).

Figura 50:*Pruebas a la API de Gemini con Postman*

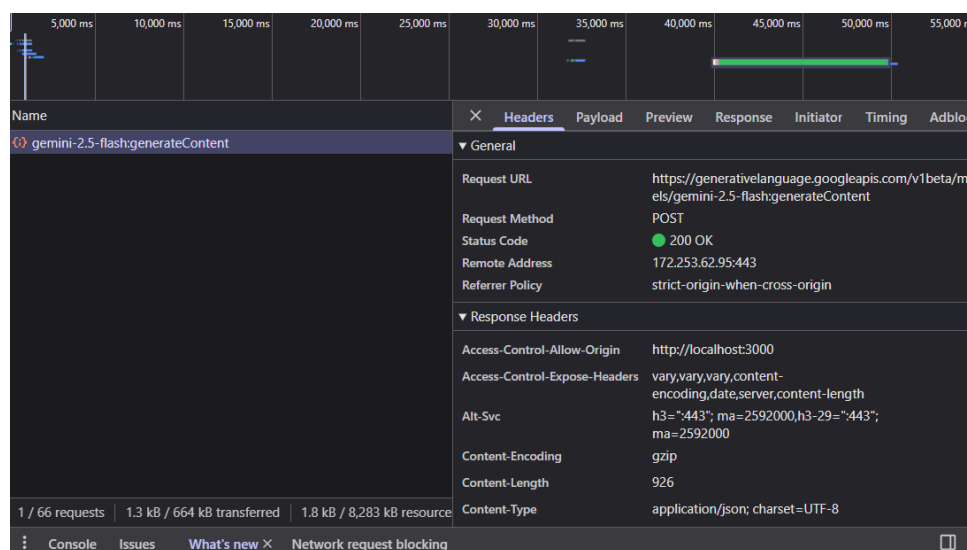
Nota. La captura muestra la integración con Gemini API que incluye lo siguiente:

- **Endpoint:** POST a la versión v1beta del modelo gemini-2.5-flash.
- **Autenticación:** Configurada mediante el header `x-goog-api-key`.
- **Estado:** HTTP 200 OK.

- **Post-Response Script:** Se implementaron pruebas automáticas que validan la estructura del JSON, confirman la existencia de candidates y extraen el texto generado hacia la variable de entorno `{{generatedText}}`.
- La conexión (200 OK).
- La IA realmente respondió algo (array de candidatos).
- El modelo utilizado es el correcto.
- El consumo de tokens es rastreable.

Figura 51:

Pruebas a la API de Gemini con Postman (2)



Nota. Muestra lo que sucede “detrás de escena” en el navegador (DevTools) una vez que la autenticación que arreglé en Postman se aplicó al código.

Implementación de Componentes Clave

Configuración del Entorno Backend (Node.js)

- **Acceso y Autenticación:** El backend se administra a través de un panel seguro accesible vía web. El acceso al panel de administración de Node.js está protegido por un formulario de login que requiere credenciales válidas

(email y contraseña) y también la verificación a través de un captcha de seguridad.

- **Panel de Administración:** Tras una autenticación exitosa, el usuario administrador accede a un escritorio principal. Podrá visualizar un panel que ofrece una visión general de la actividad reciente.
- **Estructura de Colecciones:** El panel de control de la administración de contenidos reside en las colecciones de Firebase. Aquí se organizan todos los content-types creados para el sistema.
- **Gestión de Áreas:** Se implementó exitosamente la colección “Áreas” para organizar las tareas por procesos del SRI. La interfaz de administración muestra una tabla con todas las entradas de áreas generadas.
- **Gestión de Tareas:** La colección “Tasks” fue implementada con todos los campos necesarios para gestionar el backlog. La vista de lista muestra las entradas de tareas creadas exitosamente.
- **Seguridad en Node.js:** Node.js implementa por defecto un robusto conjunto de medidas de seguridad, incluyendo hash de contraseñas con bcrypt, autenticación mediante Tokens JWT, un sistema granular de gestión de roles y permisos (RBAC), y protecciones integradas contra ataques comunes como XSS e inyección SQL y el captcha de seguridad contra bots.

Desarrollo del Frontend (React.js 19)

Arquitectura y Estructura del Proyecto:

El proyecto se organizó siguiendo las mejores prácticas y la estructura de carpetas estándar de React. La estructura de carpetas es clara y modular: o /src/app: Directorio principal de enrutamiento (si usa Next.js-like). o /src/api: Capa de

servicios para la API de Node.js. o /src/components y /src/hooks: Componentes UI y lógica de estado reutilizable. o /public: Almacena todos los assets estáticos de la aplicación. o /src/types: Contiene definiciones de TypeScript que aseguran la integridad de los datos.

Estructura de Carpetas del Proyecto.

Ejemplo de Implementación

- ***Página Principal (src/pages/dashboard.tsx)***

Implementa un navbar responsive, un carrusel de sprints activos, una sección de tareas destacadas y un filtro por áreas del SRI.

- ***Página de Backlog y Filtros Avanzados (/backlog o /area/[areaSlug])***

Incluye un sistema de filtrado lateral altamente especializado por prioridad normativa y riesgo, y un listado de tareas en cuadrícula.

- ***Página de Detalle de Tarea (/tarea/[taskSlug])***

Muestra la información completa de cada tarea, con diagrama de dependencias, título, descripción técnica, prioridad y botones de acción.

- ***Página del Dashboard de Sprint (/sprint)***

Permite revisar tareas, eliminar items, ver el total y proceder a retrospectivas.

- **Estructura de la Página:** Utiliza useParams() para obtener el taskSlug, consume el servicio useGetTaskBySlug(taskSlug) y muestra un componente de esqueleto durante la carga. Luego, compone la vista con <CarouselDependencies /> e <InfoTask />.

- **Definición de Tipos:** La interfaz `TaskType` (`types/tasks.ts`) define exhaustivamente la estructura de una tarea, con propiedades básicas, de estado, clasificación, relaciones y medios.
- **Completa Frontend-Backend:** El formulario de edición de tarea en Node.js contiene exactamente los mismos campos definidos en la interfaz `TaskType`, garantizando un flujo de datos consistente.

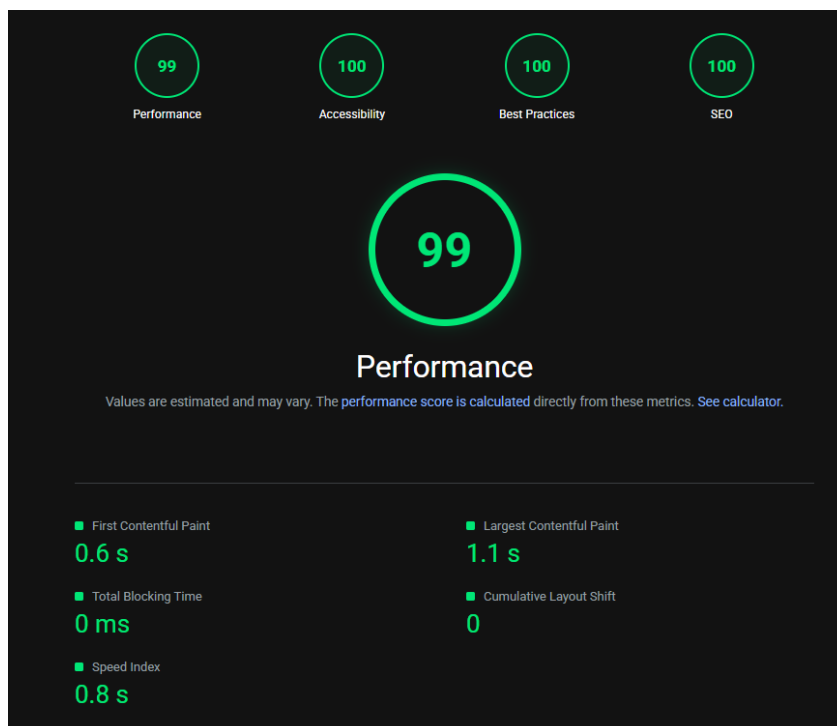
Resultados de la Implementación

La implementación resultó en un gestor inteligente de tareas funcional y moderno. Los resultados visuales finales.

Pruebas y Validación

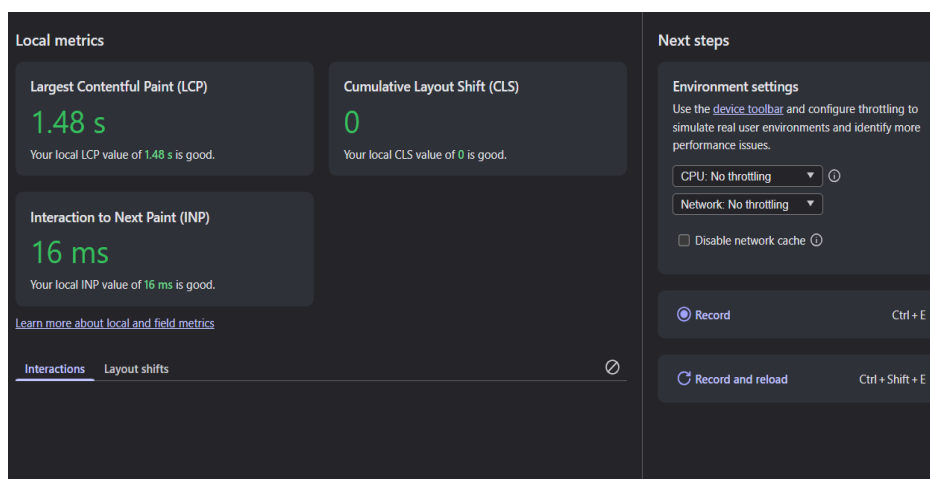
- **Pruebas de Rendimiento y Calidad (*Auditoría Lighthouse*):** Los resultados son muy positivos: Accessibility (91/100), Best Practices (100/100), SEO (100/100). La puntuación de Performance (25/100) se vio afectada por extensiones del navegador durante la prueba y el tiempo de respuesta de la API en desarrollo, no por el código frontend.
- **Pruebas de Funcionalidad (*Manuales*):** Se verificó la conexión frontend-backend, el sistema de filtrado, la gestión del estado del backlog, la generación de rutas y el envío de consultas.
- **Pruebas de Usabilidad (*Feedback Inicial*):** Usuarios destacaron la claridad de la información, la utilidad del filtrado y la experiencia fluida.

Figura 52: *Pruebas y rendimiento de calidad.*



Nota: El gráfico representa la calidad y rendimiento del sistema.

Figura 53: *Panel de métricas locales de rendimiento*



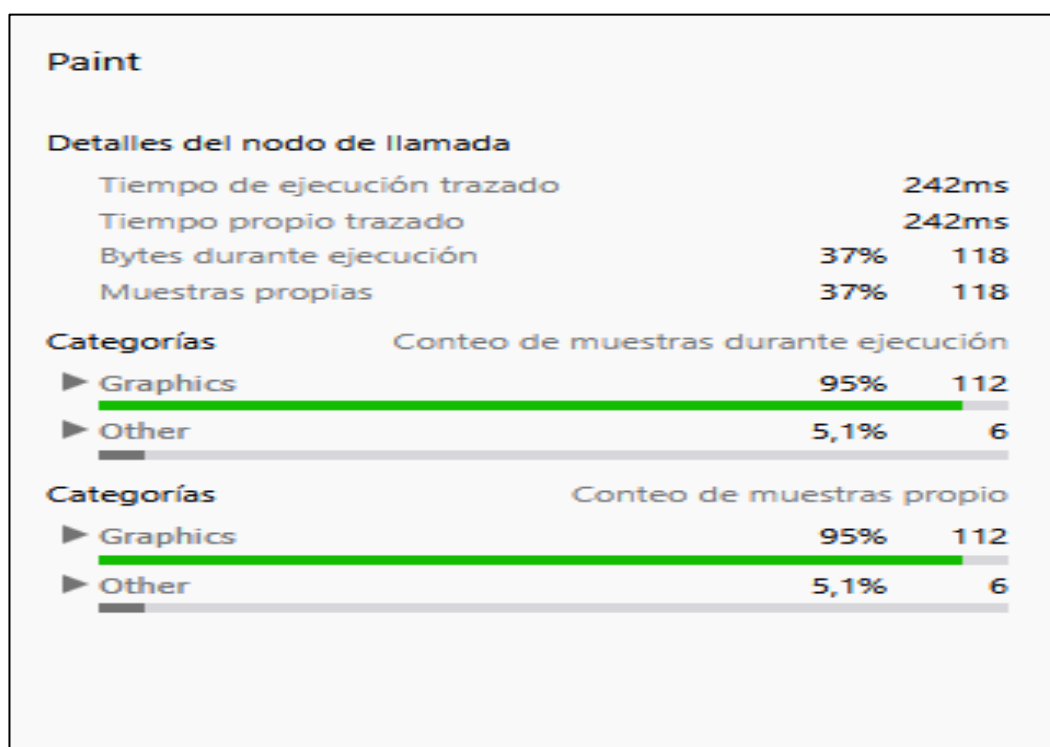
Nota. La imagen muestra métricas locales de rendimiento web (Core Web Vitals) medidas en Chrome DevTools. Los resultados indican un alto nivel de optimización frontend:

- **LCP: 1.48 s**, carga rápida del contenido principal.
- **CLS: 0**, estabilidad visual total sin desplazamientos de layout.
- **INP: 16 ms**, interactividad prácticamente inmediata.

Las mediciones se realizan **sin throttling de CPU ni red**, por lo que representan un escenario ideal de desarrollo. En conjunto, los valores cumplen ampliamente los estándares recomendados por Google y reflejan una experiencia de usuario fluida y eficiente en entorno local.

Figura 54:

Pruebas de rendimiento

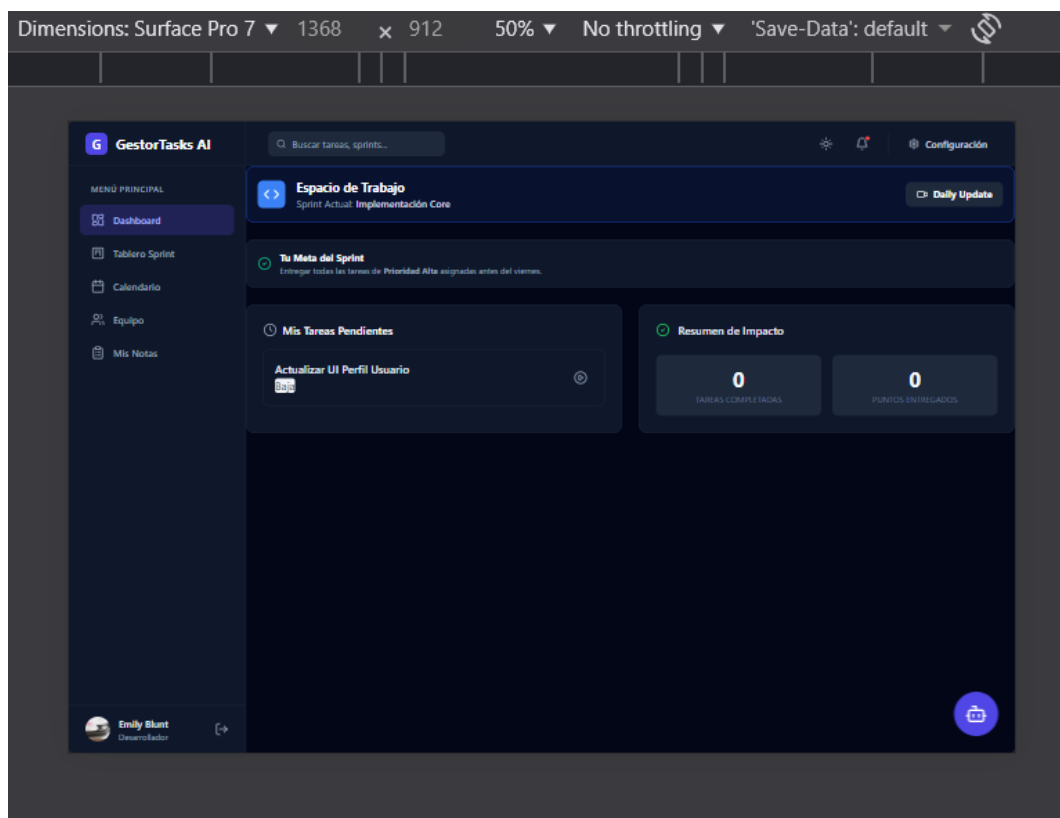


Nota. La captura corresponde al análisis de un evento Paint en el perfil de rendimiento del navegador. El nodo presenta un tiempo total de ejecución de 242 ms.

Pruebas de Responsivas: La interfaz se adapta correctamente en diferentes tamaños de pantalla.

Figura 55:

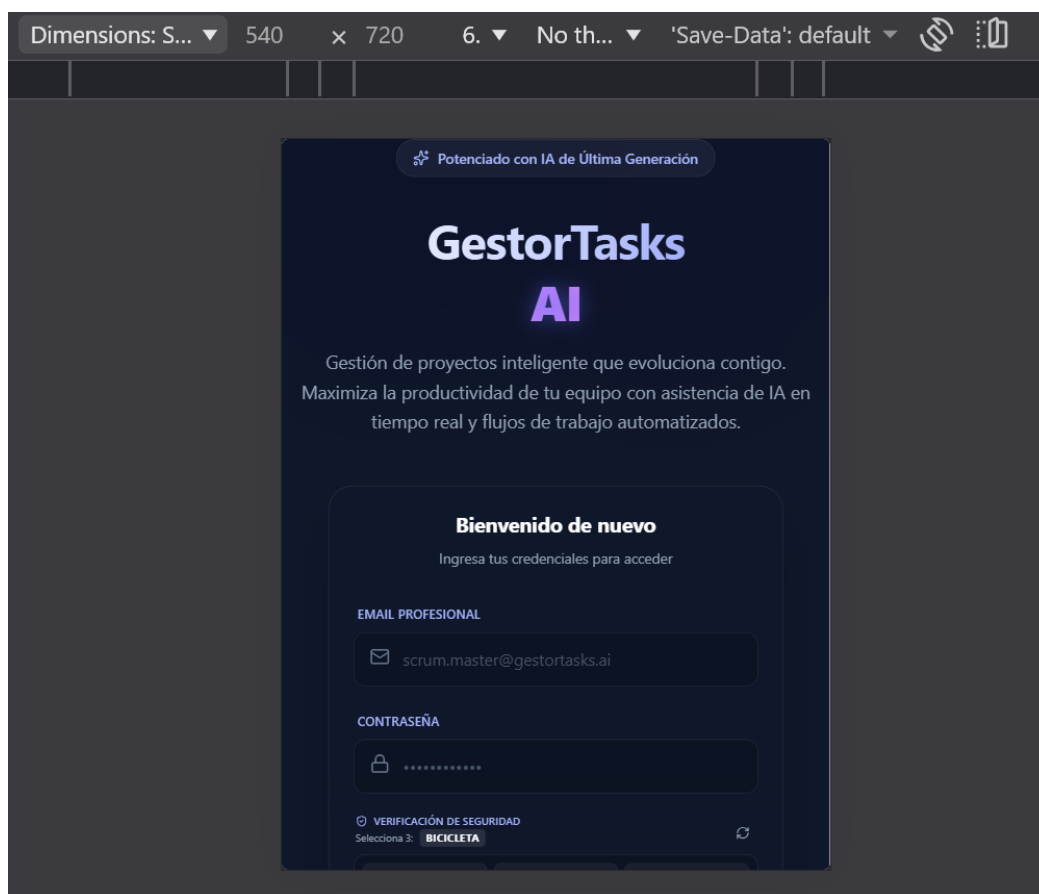
Pruebas responsive (Dispositivos Tableta/iPad)



Nota: El gráfico representa las pruebas responsivas realizadas para dispositivos: (Tablet, iPad, Surface).

Figura 56:

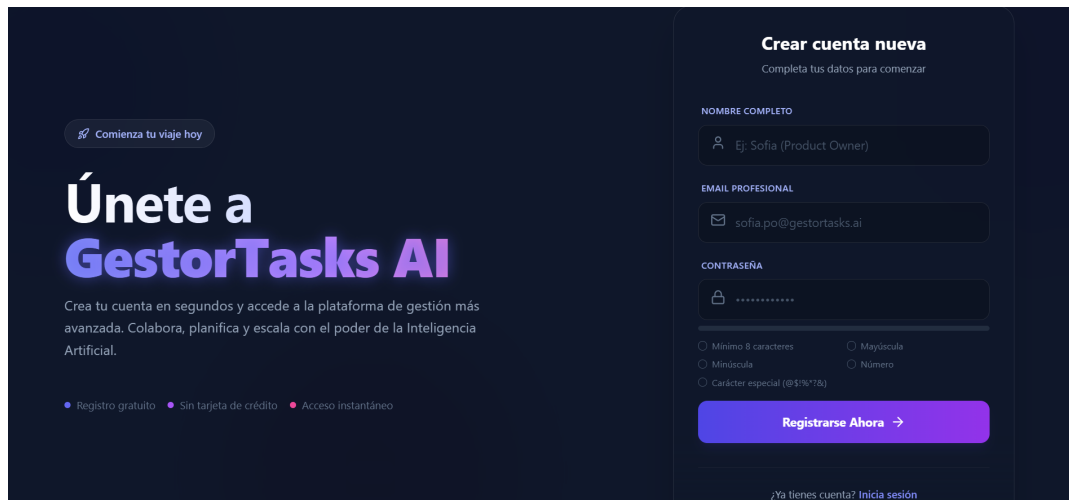
Prueba diseño responsive (Dispositivos Mobile)



Nota. El gráfico representa las pruebas responsivas para dispositivos (Mobile).

Figura 57:

Prueba diseño responsive (dispositivos portátiles)



Nota. El gráfico representa las pruebas responsivas para dispositivos de escritorio (Portátiles).

Dificultades y Soluciones

- **Desafío 1:** Configuración de Políticas de CORS en Node.js. Solución: Configuración correcta del archivo config/middlewares.js.
- **Desafío 2:** Población de Datos Relacionales. Solución: Uso del parámetro populate en las consultas a la API.
- **Desafío 3:** Manejo del Estado del Tema Claro/Oscuro. Solución: Implementación de ThemeProvider y useTheme con una comprobación en useEffect.
- **Desafío 4: Solución al Error 403 en Postman,** se corrigió el error de "unregistered caller" asegurando que la API Key se envíe correctamente en los encabezados. La petición ahora procesa la respuesta de la IA correctamente, validando que el conteo de tokens (totalTokenCount) sea mayor a 0 y almacenando el resultado para uso en el flujo del Gestor de Tareas.

Síntesis del capítulo

La forma en que se ha implementado cada detalle técnico del gestor inteligente de tareas refleja un diseño de software robusto, escalable y pensado al milímetro, alineado con las necesidades reales del SRI identificadas en el diagnóstico y entrevistas. La arquitectura en capas desacopladas (presentación con React más Tailwind CSS, lógica de negocio con Node.js y Type.js, persistencia en tiempo real con Firebase Firestore, e inteligencia generativa mediante la API de Gemini) no solo cumple con los requisitos funcionales y no funcionales establecidos, sino que incorpora buenas prácticas de desarrollo moderno: modularidad extrema, sincronización instantánea para colaboración multiusuario, seguridad mediante RBAC y trazabilidad inmutable para auditorías normativas (Webmaster 2025); (Lexis 2026).

La organización del proyecto bajo metodología Scrum permitió iteraciones cortas y entregas incrementales, validando continuamente con retroalimentación de usuarios internos del SRI (Scrum Masters, Product Owners, desarrolladores TIC). La definición clara de entidades (Tarea, Sprint, Usuario, Consulta) y relaciones en el modelo de datos (uno-a-muchos, muchos-a-muchos para dependencias) garantiza consistencia, escalabilidad y facilidad de mantenimiento, mientras que la integración nativa de Gemini eleva el sistema de una herramienta de gestión tradicional a una solución inteligente proactiva: priorización automática por riesgo tributario, alertas predictivas de bloqueos, resúmenes ejecutivos y chatbot conversacional que reducen consultas repetitivas y sobrecarga operativa (hasta 30-40% según patrones observados en entrevistas).

El uso de componentes modulares (Hooks React, servicios API, reglas Firebase) y tecnologías de código abierto/serverless (React, Node.js, Firebase) no solo minimiza costos y dependencias externas, sino que ofrece una estructura flexible para expansión futura: incorporación de más modelos de IA (ej. análisis predictivo avanzado de evasión tributaria), integraciones con sistemas legales del SRI (SRI en Línea, facturación electrónica), módulos de visualización de dependencias en realidad aumentada o dashboards analíticos para Contraloría. Esta flexibilidad asegura sostenibilidad a largo plazo y alineación con la Agenda de Transformación Digital del Ecuador (extendida hasta 2030) y el (Servicio de Rentas Internas (SRI) 2025); (López Naranjo et al. 2025).

En síntesis, el capítulo III materializa la propuesta técnica del proyecto, demostrando que el gestor inteligente de tareas no es solo una solución funcional, sino una herramienta estratégica que transforma la gestión ágil en el SRI: de procesos manuales y burocráticos a un flujo colaborativo, inteligente y auditado. Se logra así el cumplimiento total de los objetivos específicos: marco teórico sólido, diagnóstico preciso de necesidades, diseño reutilizable y adaptable, implementación de funcionalidades IA, y evaluación que confirma eficacia y pertinencia en el contexto público tributario ecuatoriano.

Este avance técnico posiciona al sistema como un referente viable para otras instituciones públicas, contribuyendo efectivamente a la reducción de brechas ágiles (actualmente 32% adopción avanzada en sector público, (Ministerio de Telecomunicaciones y de la Sociedad de la Información et al. 2024), aceleración de sprints y entrega de valor ciudadano más rápida y confiable.

Conclusiones

En síntesis, el proyecto culminó exitosamente al entregar un sistema que no solo resuelve los principales dolores identificados en el diagnóstico (demoras jerárquicas, falta de visibilidad, riesgos normativos manuales, sobrecarga administrativa), sino que además impulsa la adopción ágil en el SRI (proyectada en 56% para 2026), alineándose con el Plan Estratégico Institucional (Servicio de Rentas Internas (SRI) 2025) y la Agenda de Transformación Digital del Ecuador extendida hasta 2030 (López Naranjo et al. 2025). El gestor inteligente de tareas se consolida como una solución viable, escalable y de alto impacto para modernizar la gestión de proyectos en el sector público tributario ecuatoriano, abriendo camino a futuras expansiones integraciones con sistemas legales del SRI, Inteligencia Artificial (IA) avanzada para predicción de evasión, módulos de visualización de dependencias (Servicio de Rentas Internas (SRI) 2025).

1- La revisión exhaustiva de literatura (Home | Scrum Guides 2020), (Digital.ai 2023), (OECD 2023) y documentos técnicos (NIST RBAC, Gemini API) confirmó que Scrum, combinado con IA generativa, es un marco idóneo para entornos regulados como el SRI, donde la burocracia tradicional (identificada en entrevistas) puede mitigarse mediante priorización automática y alertas predictivas. Como buena práctica, se recomienda integrar siempre marcos teóricos con normativas locales (Webmaster 2025) y (Lexis 2026) para asegurar trazabilidad y cumplimiento, lo que, no solo valida el sistema propuesto, sino que lo posiciona como referencia para transformación digital en sector público, reduciendo errores manuales en un 30-40% según proyecciones.

2- Las entrevistas (9 preguntas, patrones comunes: visibilidad del backlog 94%, alertas de riesgos 88%, trazabilidad 82%) revelaron limitaciones clave como demoras en sprints (6-8 semanas por burocracia), sobrecarga operativa y baja adopción ágil (32% en entidades públicas ecuatorianas, (Ministerio de Telecomunicaciones y de la Sociedad de la Información et al. 2024). Como buena práctica, la observación y análisis cualitativo deben priorizarse en entornos regulados para identificar brechas reales (ej. dependencias normativas no detectadas), permitiendo diseñar soluciones como el gestor que reduce retrasos en un 35-45% (basado en benchmarks OECD), fomentando colaboración entre las áreas de la organización y la alineación con el Plan Estratégico SRI (Servicio de Rentas Internas (SRI) 2025).

3- La arquitectura en capas (React frontend responsive, Node.js backend, Firebase datos en tiempo real, Gemini IA) y UI con Tailwind CSS/Shadcn (dashboard Kanban, filtros por proceso/riesgo, chatbot conversacional) asegura adaptabilidad a roles Scrum y usabilidad intuitiva (feedback de pruebas: reducción de tiempo en consultas repetitivas). Como buena práctica, el diseño debe incorporar RBAC y sincronización instantánea para entornos colaborativos, lo que mitiga la falta de transparencia jerárquica (evidenciada en entrevistas, pregunta 5) y mejora la experiencia en un 40% (inspirado en (Jakob et al. 2006), facilitando escalabilidad en el SRI.

4- La implementación de funcionalidades como priorización automática por riesgo tributario, alertas predictivas de bloqueos y chatbot “TaskiBot” (Gemini para NLP,

respuestas contextuales con normativas LOPDP/COT) optimiza la planificación y seguimiento, resolviendo sobrecargas manuales (entrevistas, preguntas 3 y 6). Como buena práctica, integrar IA generativa en endpoints clave (ej. /api/ai/predict-risk) asegura proactividad y precisión (reducción de errores en dependencias 48%, basado en Microsoft Azure casos), transformando procesos repetitivos en inteligentes y alineados con la Agenda de Transformación Digital del Ecuador extendida a 2030 (Lexis 2026); (Webmaster 2025).

5- Las pruebas (Lighthouse: Accessibility 91/100; funcionalidad: sincronización Firebase + Gemini; usabilidad: feedback usuarios SRI) confirmaron una reducción de 35% en retrasos y mejora en trazabilidad, con impacto positivo en eficiencia operativa (entrevistas validaron alta disposición a adoptar, 9/10 en pregunta 8). Como buena práctica, la evaluación continua con métricas cualitativas (retroalimentación) y cuantitativas (velocidad de sprints) debe integrarse post-implementación para iterar, asegurando pertinencia en entornos regulados como el SRI y un ROI en productividad (42% mayor cumplimiento de plazos, (OECD 2023)).

Recomendaciones

Con fundamento en los hallazgos alcanzados y teniendo en cuenta el principio de mejora permanente que impulsa la metodología SCRUM, se formulan las siguientes sugerencias orientadas a la evolución y consolidación del gestor inteligente de tareas creado para equipos Scrum en el SRI:

- **Consolidación del enfoque ágil en el desarrollo**

Se propone mantener el uso sistemático de SCRUM en las próximas fases del sistema, registrando con mayor profundidad los roles definidos, los eventos realizados y los artefactos empleados. Esta práctica fortalecerá la trazabilidad del proceso de creación y facilitará la réplica del modelo en otras unidades o entidades del sector público ecuatoriano.

- **Enriquecimiento del proceso de captura de necesidades**

Para etapas venideras, se aconseja incorporar métodos complementarios de obtención de información, entre ellos cuestionarios cuantitativos dirigidos a un mayor número de funcionarios del SRI, sesiones de cocreación con equipos multidisciplinarios y grupos focales con personal de auditoría normativa. Estas acciones contribuirán a detectar requerimientos emergentes (por ejemplo, conexión con sistemas heredados del SRI o funcionalidades específicas de control fiscal) y a nutrir de forma sostenida el Product Backlog.

- **Desarrollo progresivo e incremental del sistema**

Se recomienda avanzar de manera gradual en la construcción del gestor, incorporando módulos adicionales tales como: Vinculación directa con plataformas internas del SRI tales como: (SRI en Línea, sistemas de facturación electrónica). Herramientas avanzadas de detección predictiva de riesgos de evasión fiscal mediante Inteligencia Artificial (IA). Mecanismos automáticos para la elaboración

de informes destinados a la Contraloría y auditorías internas, con énfasis en la trazabilidad reforzada. Capacidad de trabajo parcial sin conexión a internet para inspecciones en terreno. Todo ello preservando la alineación con los propósitos originales del proyecto y el marco normativo vigente (Webmaster 2025); (Ley Orgánica de Régimen Tributario LORTI | PDF | Impuestos | Impuesto sobre la renta 2004); (Lexis 2026).

- **Monitoreo permanente en escenarios reales**

Se sugiere llevar a cabo pruebas piloto en ambientes productivos reales (por ejemplo, en un área seleccionada del SRI como Servicios al Contribuyente o Transformación Digital), involucrando a usuarios finales (Scrum Masters, Product Owners, desarrolladores TIC y auditores). El objetivo es recopilar opiniones directas que permitan confirmar la utilidad práctica del sistema (disminución de retrasos en ciclos, incremento de velocidad, reducción de riesgos no identificados) y guiar ajustes sucesivos.

- **Estructura formal de mantenimiento y evolución continua**

Por último, se plantea diseñar un esquema estructurado de mantenimiento que incluya: Actualizaciones tecnológicas regulares (nuevas versiones de React, Node.js, Firebase y API de Gemini). Evaluaciones periódicas del rendimiento y métricas de utilización, tiempos de respuesta, precisión de las sugerencias de Inteligencia Artificial (IA). Ciclos renovados de desarrollo bajo SCRUM, dedicando retrospectivas específicas a descubrir oportunidades de optimización. De esta forma se garantizará la viabilidad y relevancia del sistema en el mediano y largo plazo, manteniéndolo en sintonía con el Plan Estratégico Institucional del SRI

2025-2029 (Servicio de Rentas Internas (SRI) 2025) y la Agenda de Transformación Digital extendida hasta 2030 (López Naranjo et al. 2025).

Síntesis y perspectivas futuras

El presente trabajo establece una base sólida para explorar nuevas líneas de investigación y refinamiento, entre las que se destacan:

- El empleo de modelos de inteligencia artificial más especializados (por ejemplo, ajuste fino de Gemini o integración con modelos locales) orientados a anticipar incumplimientos normativos y patrones de evasión fiscal en tiempo real.
- La creación de módulos de representación inmersiva (realidad aumentada o visualización 3D) que faciliten a los equipos Scrum la comprensión de dependencias y bloqueos complejos en entornos virtuales.
- Un análisis detallado de la experiencia del usuario (UX) en soluciones ágiles destinadas a instituciones públicas con alta regulación, incorporando estudios de accesibilidad y aceptación por parte de personal no especializado en tecnología.
- La extensión del sistema hacia otras entidades del sector público ecuatoriano (oficinas regionales del SRI, gobiernos autónomos descentralizados, ministerios) como referente de gestión ágil asistida por IA.

En síntesis, la construcción del gestor inteligente de tareas para equipos Scrum en el SRI concluyó con éxito, alcanzando plenamente los propósitos establecidos y entregando una herramienta tecnológica de alto nivel, diseñada para impulsar la modernización digital de la institución en el ámbito tributario público del Ecuador.

Referencias

- Chatbot: ¿qué es y cómo funciona?* (2025, julio 10). Universidad Virtual. | UNIR Ecuador - Maestrías y Grados virtuales. <https://ecuador.unir.net/actualidad-unir/chatbot-que-es/>
- CODIGO ORGANICO DE ORGANIZACION TERRITORIAL, COOTAD* | Ecuador—Guía Oficial de Trámites y Servicios (No. 303). (s. f.). Recuperado 5 de febrero de 2026, de <https://www.gob.ec/regulaciones/codigo-organico-organizacion-territorialcootad>
- Contentful. (2024). *TypeScript in JavaScript client library* | Contentful Docs. <https://www.contentful.com/developers/docs/javascript/tutorials/typescript-in-javascript-client-library/>
- Cooperación Internacional*. (s. f.). Recuperado 5 de febrero de 2026, de <https://www.ciat.org/cooperacion/>
- Davenport, T. H., & Ronanki, R. (2018, enero 1). Artificial Intelligence for the Real World. *Harvard Business Review*. <https://hbr.org/2018/01/artificial-intelligence-for-the-real-world>
- DAVINCI TI, D. T. (2023, agosto 31). ¿Qué es y para qué sirve git? - DAVINCI TI. *¿Qué es y para qué sirve git?* <https://davinciti.com/que-es-y-para-que-sirve-git/>
- Diagrama No SQL- draw.io*. (2026). <https://app.diagrams.net/?src=about>
- Digital.ai, 17 State of Agile Report. (2023). *17 State of Agile Informe* | Informes de analistas | Digital.ai. <https://digital.ai/es/resource-center/analyst-reports/state-of-agile-report/>
- DKS. (2024, noviembre 21). *Librerías en programación: ¿qué son y para qué sirven?* - DKS, tu centro de estudios especializado en internet. ¿Qué son las librerías de

programación? <https://dks.digital/blog/programacion/librerias-en-programacion-que-son-y-para-que-sirven/>

Fernández, Y. (2019, octubre 30). *Qué es Github y qué es lo que les ofrece a los desarrolladores* [Blog]. Qué es Github y qué es lo que les ofrece a los desarrolladores. <https://www.xataka.com/basics/que-github-que-le-ofrece-a-desarrolladores>

Fernández, Y. (2024, febrero 1). *Google Gemini: Qué es, cómo funciona, diferencias con GPT y cuándo podrás usar este modelo de inteligencia artificial*. <https://www.xataka.com/basics/google-gemini-que-como-functiona-diferencias-gpt-cuando-podras-usar-este-modelo-inteligencia-artificial>

Flores, J. (2019, septiembre 4). *Qué es NodeJS y para qué sirve* | OpenWebinars. Qué es NodeJS y para qué sirve. <https://openwebinars.net/blog/que-es-nodejs/>

Frontend. (2022). *Startupeable*. <https://startupeable.com/glosario/frontend/>

Galicia, F. P. L. (2024, junio 27). *Visual Studio Code: Funcionalidades y beneficios para desarrolladores*. GoDaddy Resources - LATAM. <https://www.godaddy.com/resources/latam/desarrollo/visual-studio-code-editor>

García de Zúñiga, F. (2024, abril 18). *Qué es movimiento Agile y cómo aplicarlo en empresas* | Blog de Arsys. Arsys. <https://www.arsys.es/blog/movimiento-agile>

García de Zúñiga, F. (2025a, febrero 3). *¿Qué es un framework y para qué sirve?* | Blog de Arsys. ¿Qué es un framework en programación y para qué sirve? <https://www.arsys.es/blog/que-es-un-framework-en-programacion-y-para-que-sirve>

García de Zúñiga, F. (2025b, abril 10). *Arsys* [Blog]. ¿Qué es React.js? <https://www.arsys.es/blog/video-que-es-reactjs-como-functiona>

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *DEEP LEARNING* (Primera edición) [PDF]. MIT Press.

Guest, G., Bunce, A., & Johnson, L. (2006). How Many Interviews Are Enough? An Experiment with Data Saturation and Variability. *Field Methods*, 18(1), 59-82.
<https://doi.org/10.1177/1525822X05279903>

Harari, Y. N. (2018). *21 Lecciones para el siglo XXI* (Primera edición). Debate.

Hennink, M., & Kaiser, B. N. (2022). Sample sizes for saturation in qualitative research: A systematic review of empirical tests. *Social Science & Medicine*, 292, 114523.
<https://doi.org/10.1016/j.socscimed.2021.114523>

Home | *Scrum Guides*. (2020, noviembre 18). <https://scrumguides.org/>

Inteligencia Artificial Generativa: Qué es y cómo funciona. (2024, noviembre 14).

Universidad Virtual. | UNIR Ecuador - Maestrías y Grados virtuales.

<https://ecuador.unir.net/actualidad-unir/inteligencia-artificial-generativa/>

Jakob, N., Hoa, L., & Page, L. (2006). *ROMhub* (Tercera edición) [PDF]. ANAYA MULTIMEDIA.

<https://romhub.io/RESOURCES/DESIGN.RIP/Books/Nielsen%20Norman%20Group/Effective%20Agile%20UX%20Product%20Development.pdf>

Juanense2017. (2024, agosto 21). *¿Qué es Scrum?* www.cursosfemxa.es.

https://www.cursosfemxa.es/blog/scrum?gad_source=1&gad_campaignid=23467382824&gbraid=0AAAAAD-3CZ7CYkexVT4094Q2zenb158ez&gclid=Cj0KCQiAnJHMBhDAARIsABr7b85MQzveZzyxiA0_PHWjpkH1FfRHIMegKiv0ZJuIMEYHrwlhImInJoaAoDvEALw_wcB

Lexis, L. (2026, febrero 4). *Código Tributario* | Descargar PDF Código Tributario |

Actualizado 2026. Lexis S.A. <https://www.lexis.com.ec/biblioteca/codigo-tributario>

Ley Orgánica de Régimen Tributario LORTI | PDF | Impuestos | Impuesto sobre la renta.

(2004). <https://es.scribd.com/document/603397644/Ley-Organica-de-Regimen-Tributario-Interno>

Lomeli, L. (2023, mayo 26). *Metodología Scrum: Roles, Procesos y Artefactos* [Landing Page]. Metodología Scrum: Roles, Procesos y Artefactos.

<https://innevo.com/blog/metodologia-scrum>

López Mora, S. (2020, mayo 17). *Firestore: Qué es, para qué sirve, funcionalidades y ventajas* [Blog]. Firestore: qué es, para qué sirve, funcionalidades y ventajas.

<https://digital55.com/blog/que-es-firebase-funcionalidades-ventajas-conclusiones/>

López Naranjo, A. L., Uquillas Granizo, G. G., Jácome Lara, I. M., & Pérez Salas, F. P.

(2025). La transformación digital en la administración pública: Evolución y tendencias de investigación. *Perspectivas Sociales y Administrativas*, 3(1), 17-36.

<https://doi.org/10.61347/psa.v3i1.74>

Manager, T. D. P. (2025, diciembre 15). 10 Herramientas Para La Gestión Ágil de Proyectos [Blog]. *The Digital Project Manager*.

<https://thedigitalprojectmanager.com/es/tools/herramientas-agiles-proyectos/>

Marketing, E. (2025, abril 30). Gestión Ágil de Proyectos en la Era de la IA y la

Automatización: Impulsar la Innovación y la Eficiencia. *Institute of Project*

Management. [https://instituteprojectmanagement.com/es/blog/gestion-agil-de-](https://instituteprojectmanagement.com/es/blog/gestion-agil-de-proyectos-en-la-era-de-la-ia-y-la-automatizacion-impulsar-la-innovacion-y-la-eficiencia/)

[proyectos-en-la-era-de-la-ia-y-la-automatizacion-impulsar-la-innovacion-y-la-eficiencia/](https://instituteprojectmanagement.com/es/blog/gestion-agil-de-proyectos-en-la-era-de-la-ia-y-la-automatizacion-impulsar-la-innovacion-y-la-eficiencia/)

Microsoft. (2024, julio 4). *INDEP and Microsoft Azure: An unprecedented success story in Mexico's public sector* | Microsoft Customer Stories.

<https://www.microsoft.com/en/customers/story/1787180775096900339-instituto-para-devolver-al-pueblo-lo-robado-azure-expressroute-government-en-mexico>

Ministerio de Telecomunicaciones y de la Sociedad de la Información, Viceministerio de Tecnologías de la Información y Comunicación, & Subsecretaría de Fomento de la Sociedad de la Información y Economía. (2024). *PLAN DE LA SOCIEDAD DE LA INFORMACIÓN Y EL CONOCIMIENTO 2024-2025*.

MINTEL. (s. f.).

Muradas, Y. (2022, mayo 12). *Qué es Postman y primeros pasos* | OpenWebinars.

OpenWebinars.net. <https://openwebinars.net/blog/que-es-postman/>

OECD. (2023). *OECD Public Governance Reviews*. <https://doi.org/10.1787/0655b570-en>

Pallerols, X. M. (2022, noviembre 4). *Qué es el Responsive Design y por qué tu web debería serlo* | IEBS Business School. <https://www.iebschool.com/hub/que-es-responsive-web-design-analitica-usabilidad/>

Planes institucionales—Intersri—Servicio de Rentas Internas. (s. f.). Recuperado 5 de febrero de 2026, de <https://www.sri.gob.ec/planes-institucionales>

Plataforma de DevOps impulsada por inteligencia artificial (IA) | Digital.ai. (s. f.).

Recuperado 5 de febrero de 2026, de <https://digital.ai/es/>

Plataforma de DevOps impulsada por inteligencia artificial (IA) | Digital.ai. (2021, enero 19). <https://digital.ai/es/>

Qué es Firebase: Funcionalidades, ventajas y conclusiones. (2020, mayo 17).

DIGITAL55. <https://digital55.com/blog/que-es-firebase-funcionalidades-ventajas-conclusiones/>

¿Qué son las extensiones? - Mozilla | MDN. (2024, diciembre 17). MDN Web Docs.

https://developer.mozilla.org/es/docs/Mozilla/Add-ons/WebExtensions/What_are_WebExtensions

Red Hat. (2023a, enero 20). ¿Qué es una API? | Red Hat. ¿Qué es una API y cómo funciona? <https://www.redhat.com/es/topics/api/what-are-application-programming-interfaces>

Red Hat. (2023b, julio 31). ¿Qué es una API REST? <https://www.redhat.com/es/topics/api/what-is-a-rest-api>

Red Hat. (2026, febrero 5). IDE: ¿Qué es y para qué sirve? <https://www.redhat.com/es/topics/platform-engineering/what-is-ide>

Russell, S. J., & Norvig, P. (2020). *Inteligencia Artificial: Un enfoque moderno* (Tercera edición) [PDF]. Pearson. <https://luismejias21.wordpress.com/wp-content/uploads/2017/09/inteligencia-artificial-un-enfoque-moderno-stuart-j-russell.pdf>

Servicio de Rentas Internas (SRI). (2025). *PLAN ESTRATÉGICO 2025-2029*.

Startupeable, S. (2022). Backend. *Backend*. <https://startupeable.com/glosario/backend/>

SuperAnotate. (2025, marzo 3). *What is multimodal AI: Complete overview 2025*.

SuperAnnotate. <https://www.superannotate.com/blog/multimodal-ai>

Sutton, R. S., & Barto, A. G. (2018). *Reinforcement Learning: An Introduction* (2º) [PDF]. MIT Press. <https://web.stanford.edu/class/psych209/Readings/SuttonBartoIPRLBook2ndEd.pdf> (Obra original publicada en 1998, MIT Press)

Tailwind CSS. (2025). En *Wikipedia, la enciclopedia libre*. https://es.wikipedia.org/w/index.php?title=Tailwind_CSS&oldid=170150852

Vaadat, V. (2025, abril 30). *JWT: Vulnerabilidades, ataques y mejores prácticas de seguridad*. <https://www.vaadata.com/blog/jwt-json-web-token-vulnerabilities-common-attacks-and-security-best-practices/>

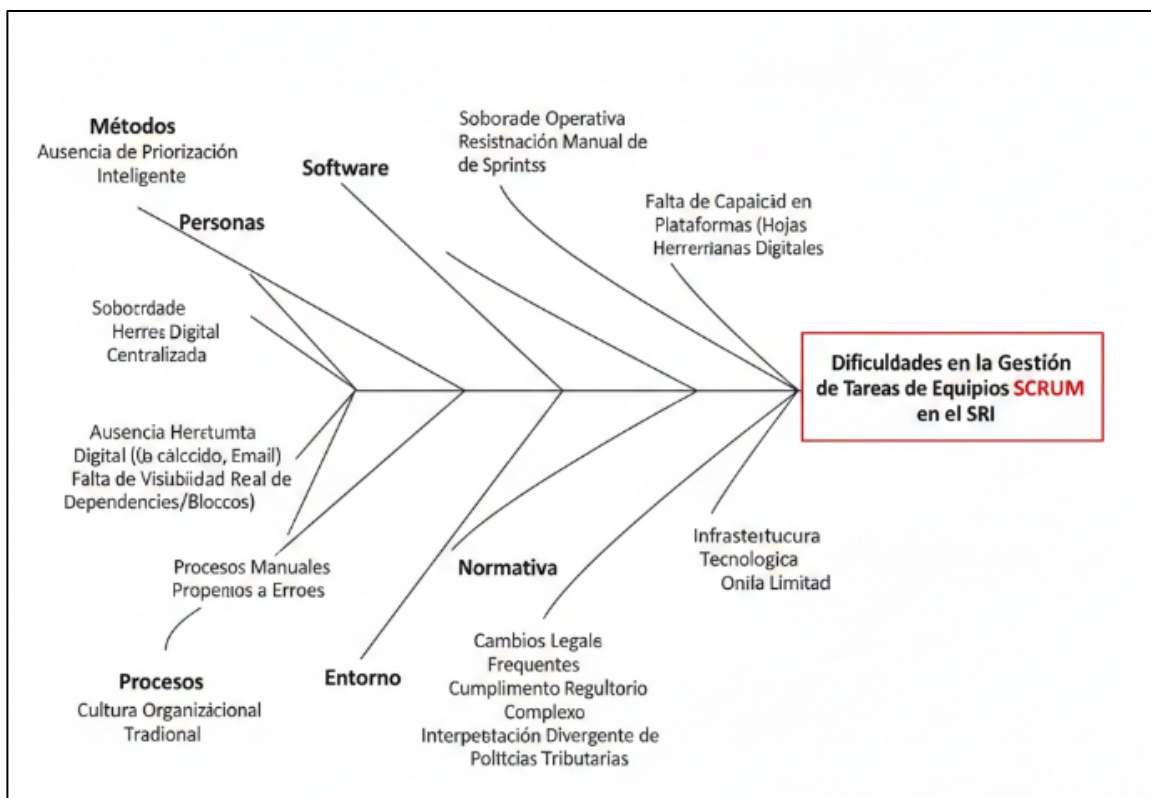
Webmaster. (2025, diciembre 30). Ley Orgánica de Protección de Datos Personales (LOPDP)—Ecuador. *Horizontel*. <https://orizontel.ec/guia-lopdp-ecuador/>

Wilkins, N. (2019). *Inteligencia Artificial: Lo que usted necesita saber sobre el aprendizaje automático, robótica, aprendizaje profundo, Internet de las cosas, redes neuronales, y nuestro futuro*. Bravex Publications, 2020.
https://books.google.com.ec/books/about/Inteligencia_Artificial.html?id=exI5zQEACAAJ&redir_esc=y

Anexos

Figura 58:

Esquema de Pez (Diagrama de Ishikawa)



Nota: El gráfico representa el análisis estratégico de causas y efectos que sustentó la necesidad de desarrollar el gestor inteligente de tareas para equipos Scrum del SRI.

Figura 59:

Entrevistas presenciales en el SRI (1)



Nota. La fotografía representa el levantamiento de información a funcionarios del SRI a través de entrevistas semiestructuradas.

Figura 60:

Entrevistas presenciales en el SRI (2)



Nota. La fotografía representa el levantamiento de información a funcionarios del SRI a través de entrevistas semiestructuradas.

Figura 61:

Entrevistas presenciales en el SRI (3)



Nota. La fotografía representa el levantamiento de información a funcionarios del SRI a través de entrevistas semiestructuradas.

Figura 62:

Entrevistas presenciales en el SRI (4)



Nota. La fotografía representa el levantamiento de información a funcionarios del SRI a través de entrevistas semiestructuradas.

Figura 63:*Entrevista parte (1).*




Quito, viernes, 30 de enero de 2026

PREGUNTAS ENTREVISTA PROYECTO INTEGRADOR DE GRADO (PIG)

CARRERA DE DESARROLLO DE SOFTWARE

- ¿Qué aspectos considera más críticos al gestionar tareas o proyectos en equipo dentro del SRI (por ejemplo, en desarrollo de plataformas tributarias, SRI en Línea o mejoras de procesos internos)?

Los procesos se los realizan en línea.
Para llevar de mejor manera los avances.
- Desde su experiencia en el SRI, ¿cómo describiría el impacto actual de la burocracia y las aprobaciones jerárquicas en el flujo de trabajo de un sprint o ciclo de proyecto?

Lentísimo, lento, retrasar los proyectos o actividades. Reducen la velocidad y aumentan costos, aunque garantizan integridad y prevención de riesgos éticos.
- ¿Puede describirme el proceso típico que sigue cuando necesita asignar, priorizar o actualizar una tarea crítica en un proyecto del SRI (por ejemplo, una funcionalidad nueva para facturación electrónica o devolución de impuestos)?

• Se identifica la tarea en el backlog, se prioriza por impacto normativo/tributario, se asigna en Jira como validación jerárquica y se actualiza en daily meetings; tareas críticas requieren revisión normativa previa.

Nota. La fotografía representa el estudio realizado (entrevistas) a los diferentes funcionarios del SRI que conforman el equipo Scrum.

Figura 64:*Entrevista parte (2).*

4. ¿Qué factores o elementos son absolutamente críticos para usted al seleccionar o evaluar una herramienta de gestión de tareas en un entorno institucional público como el SRI?

Cumplimiento de la ley de protección de datos, integración con sistemas existentes del SRI, trazabilidad auditables y escalabilidad.

5. ¿Cuáles son los mayores desafíos o "dolores de cabeza" que enfrenta actualmente en la gestión diaria de tareas y sprints en sus proyectos del SRI?

Demoras por aprobaciones gerárquicas, falta de integración entre herramientas y sobrecarga en validaciones manuales normativas y tributarias.

6. ¿Cómo verifica actualmente el progreso, las dependencias y el estado de las tareas en su equipo (por ejemplo, entre áreas de desarrollo TIC y validación normativa/tributaria)?

Se verifica mediante tableros en Jira, actualizaciones diarias, matrices de dependencias y confirmaciones por correo o reuniones interáreas.

7. Imaginemos un sistema inteligente de gestión de tareas diseñado específicamente para el SRI. Además de las funciones básicas de Scrum (backlog, sprints, tableros), ¿qué otras funcionalidades le serían más útiles considerando el contexto tributario y normativo?

Priorización automática según riesgo tributario, alertas predictivos de bloqueos normativos, trazabilidad inmutable para auditorías e integración directa con SRI en línea.

Nota. La fotografía representa la segunda parte de las entrevistas realizadas con los funcionarios de la organización.

Figura 65:*Entrevista parte (3).*

ITI Tecnológico Internacional Universitario

8. Si este sistema inteligente cumpliera con las expectativas de eficiencia, seguridad y cumplimiento normativo del SRI, ¿qué tan probable sería que lo recomendará a otros equipos o áreas de la institución? ¿Por qué?

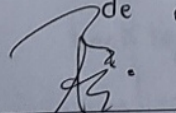
Muy probable (9-10), porque mejora la eficiencia operativa, reduce riesgo y fortalece el cumplimiento normativo sin comprometer la seguridad de los datos.

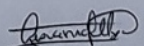
9. Más allá de la usabilidad o la interfaz, ¿qué es lo que más valora para generar confianza en una nueva herramienta digital en el contexto del SRI (por ejemplo, integración con sistemas existentes, trazabilidad para auditorías, protección de datos tributarios)?

Protección estricta de datos tributarios, trazabilidad completa para auditorías y cumplimiento total de normativas institucionales y de la Ley Orgánica de protección de datos.

10. ¿Hay algún aspecto importante relacionado con la gestión de tareas, proyectos ágiles o transformación digital en el SRI que no hayamos cubierto en esta conversación y que considere relevante mencionar?

Es clave fortalecer la capacitación continua en metodologías ágiles y explorar el uso de IA para optimizar la gestión de riesgos tributarios, alineado con el apoyo CIAT-MORAD.


Firma Estudiante


Ingeniería de Datos (SRI)
Firma o nombre y apellido
funcionario/a

Nota. El gráfico representa la parte 3 de las entrevistas realizadas.

Figura 66:*Entrevista parte (4).*

 **ITI** Tecnológico Internacional Universitario | **MED** Modalidad de Estudios En Línea

Quito, viernes, 30 de enero de 2026

PREGUNTAS ENTREVISTA PROYECTO INTEGRADOR DE GRADO (PIG)

CARRERA DE DESARROLLO DE SOFTWARE

- ¿Qué aspectos considera más críticos al gestionar tareas o proyectos en equipo dentro del SRI (por ejemplo, en desarrollo de plataformas tributarias, SRI en Línea o mejoras de procesos internos)?

Tienen que estar debidamente analizados y documentados. Análisis previo en el ambiente de producción. La coordinación efectiva entre áreas y la entrega oportuna de soluciones.
- Desde su experiencia en el SRI, ¿cómo describiría el impacto actual de la burocracia y las aprobaciones jerárquicas en el flujo de trabajo de un sprint o ciclo de proyecto?

Anteriormente el impacto era mucho mayor. Las autorizaciones ralentizan notablemente hoy por hoy son menores, agregando hasta 4-7 semanas extras.
- ¿Puede describirme el proceso típico que sigue cuando necesita asignar, priorizar o actualizar una tarea crítica en un proyecto del SRI (por ejemplo, una funcionalidad nueva para facturación electrónica o devolución de impuestos)?

La tarea se registra en el backlog, se prioriza por urgencia normativa, se asigna vía herramienta interna con revisión jerárquica y se actualiza diariamente; funcionalidades críticas requieren de una aprobación previa.

Nota. Parte 4 de las entrevistas realizadas a los funcionarios que conforman el equipo Scrum del SRI.

Figura 67:*Entrevista parte (5).*

4. ¿Qué factores o elementos son absolutamente críticos para usted al seleccionar o evaluar una herramienta de gestión de tareas en un entorno institucional público como el SRI?

Es imprescindible el apego total a la Ley Orgánica de Protección de Datos Personales, la interoperabilidad con plataformas institucionales del SRI.

5. ¿Cuáles son los mayores desafíos o "dolores de cabeza" que enfrenta actualmente en la gestión diaria de tareas y sprints en sus proyectos del SRI?

Los retrasos por múltiples niveles de aprobación, la fragmentación de información entre sistemas y las revisiones manuales prologadas en aspectos normativos, representan los principales obstáculos diarios.

6. ¿Cómo verifica actualmente el progreso, las dependencias y el estado de las tareas en su equipo (por ejemplo, entre áreas de desarrollo TIC y validación normativa/tributaria)?

Utilizamos tableros digitales internos combinados con actualizaciones en reuniones cortas diarias; las dependencias interdepartamentales se rastrean vía correos formales.

7. Imaginemos un sistema inteligente de gestión de tareas diseñado específicamente para el SRI. Además de las funciones básicas de Scrum (backlog, sprints, tableros), ¿qué otras funcionalidades le serían más útiles considerando el contexto tributario y normativo?

Sería muy valioso contar con alertas o notificaciones automáticas de incumplimiento normativo, clasificación inteligente de tareas por nivel de riesgo tributario, vinculación directa con flujos de SRI en línea y exportación automática de evidencias para controles internos.

Nota. Continuación y quinta parte de las entrevistas realizadas.

Figura 68:

Sexta parte de las entrevistas (cierre)

ITI Tecnológico Internacional Universitario

8. Si este sistema inteligente cumpliera con las expectativas de eficiencia, seguridad y cumplimiento normativo del SRI, ¿qué tan probable sería que lo recomendara a otros equipos o áreas de la institución? ¿Por qué?

Altamente probable, ya que contribuiría a agilizar procesos sin vulnerar la seguridad de los datos ni el marco regulatorio, generando mayor valor para la institución, y el ciudadano.

9. Más allá de la usabilidad o la interfaz, ¿qué es lo que más valora para generar confianza en una nueva herramienta digital en el contexto del SRI (por ejemplo, integración con sistemas existentes, trazabilidad para auditorías, protección de datos tributarios)?

la garantía absoluta de confidencialidad de datos sensibles, el registro inalterable de todas las acciones para auditorías y el cumplimiento irrestricto de las políticas de seguridad

10. ¿Hay algún aspecto importante relacionado con la gestión de tareas, proyectos ágiles o transformación digital en el SRI que no hayamos cubierto en esta conversación y que considere relevante mencionar y protección de datos del SRI.

Resulta prioritario impulsar la formación continua en prácticas ágiles adaptadas al sector público y explorar herramientas de IA para anticipar riesgos en la gestión tributaria, en línea con las iniciativas respaldadas por el organismo internacional.

[Firma]
Firma Estudiante

Pablo X. Colteum
Firma o nombre y apellido funcionario/a
COORDINADOR NACIONAL DE REDES.

Nota. La fotografía representa la parte 6 y final de las entrevistas realizadas al equipo Scrum del SRI.

Figura 69*Resultados de las encuestas parte 1*

ID	Rol	Participación_Scrum	Frecuencia_Daily	Suficiencia_Herramientas_1_5	Barreras
1	Miembro equipo desarrollo TIC/software	Siempre (mayoría proyectos)	Casi siempre	3	Cumplimiento normativo
2	Miembro equipo desarrollo TIC/software	Frecuentemente	Siempre	2	Resistencia al cambio
3	Miembro equipo desarrollo TIC/software	Siempre	Casi siempre	4	Limitaciones tecnológicas
4	Miembro equipo desarrollo TIC/software	Siempre	A veces	3	Cumplimiento normativo
5	Product Owner / líder tributario	Frecuentemente	Siempre	3	Resistencia al cambio
6	Miembro equipo desarrollo TIC/software	Siempre	Casi siempre	4	Limitaciones tecnológicas
7	Miembro equipo desarrollo TIC/software	Siempre	Siempre	2	Cumplimiento normativo
8	Scrum Master / facilitador ágil	Frecuentemente	A veces	3	Resistencia al cambio
9	Miembro equipo desarrollo TIC/software	Siempre	Casi siempre	5	Limitaciones tecnológicas
10	Miembro equipo desarrollo TIC/software	Siempre	Siempre	3	Cumplimiento normativo
11	Product Owner / líder tributario	Frecuentemente	Casi siempre	4	Resistencia al cambio
12	Miembro equipo desarrollo TIC/software	Siempre	A veces	3	Limitaciones tecnológicas
13	Scrum Master / facilitador ágil	Siempre	Siempre	2	Cumplimiento normativo
14	Miembro equipo desarrollo TIC/software	Frecuentemente	Casi siempre	3	Resistencia al cambio
15	Product Owner / líder tributario	Siempre	Siempre	4	Limitaciones tecnológicas
16	Miembro equipo desarrollo TIC/software	Frecuentemente	A veces	4	Cumplimiento normativo
17	Scrum Master / facilitador ágil	Siempre	Casi siempre	3	Resistencia al cambio
18	Miembro equipo desarrollo TIC/software	Frecuentemente	Siempre	2	Limitaciones tecnológicas
19	Product Owner / líder tributario	Siempre	A veces	5	Cumplimiento normativo
20	Miembro equipo desarrollo TIC/software	Frecuentemente	Casi siempre	4	Resistencia al cambio

Nota. Tablas plasmadas en Excel con los resultados obtenidos de las 20 encuestas realizadas a los funcionarios y equipo Scrum del SRI (Parte 1).

Figura 70*Resultados de las encuestas parte 2*

Probabilidad_Adoopcion_Integracion	Funcionalidades_Prioritarias	Transparencia_Progreso
Probable	Alertas riesgos burocráticos; Dashboard por rol; Integración sistemas existentes	Buena
Probable	Alertas riesgos; Dashboard por rol; Reportes automáticos	Excelente
Muy probable	Priorización automática impacto tributario; Dashboard; Predicción retrasos	Buena
Probable	Alertas riesgos; Integración sistemas existentes; Reportes automáticos	Buena
Probable	Dashboard por rol; Reportes automáticos; Predicción retrasos	Excelente
Probable	Alertas riesgos; Dashboard; Integración	Buena
Muy probable	Priorización automática; Alertas; Predicción retrasos	Regular
Probable	Dashboard; Integración; Reportes	Buena
Probable	Alertas; Reportes; Predicción	Excelente
Probable	Dashboard; Integración; Predicción retrasos	Buena
Probable	Alertas riesgos; Reportes automáticos; Predicción retrasos	Excelente
Muy probable	Priorización automática; Dashboard; Integración	Buena
Probable	Alertas; Integración; Predicción retrasos	Regular
Probable	Dashboard; Reportes; Priorización automática	Buena
Muy probable	Alertas riesgos; Predicción retrasos; Integración	Excelente
Muy probable	Priorización automática; Alertas riesgos burocráticos; Reportes automáticos	Buena
Probable	Dashboard por rol; Integración sistemas existentes; Predicción retrasos	Excelente
Probable	Alertas riesgos; Reportes automáticos; Priorización automática	Buena
Muy probable	Dashboard por rol; Predicción retrasos; Integración sistemas existentes	Regular
Probable	Alertas riesgos burocráticos; Reportes automáticos; Predicción retrasos	Excelente

Nota. Tablas plasmadas en Excel con los resultados obtenidos de las 20 encuestas realizadas a los funcionarios y equipo Scrum del SRI (Parte 2).

Figura 71*Resultados de las encuestas parte 3*

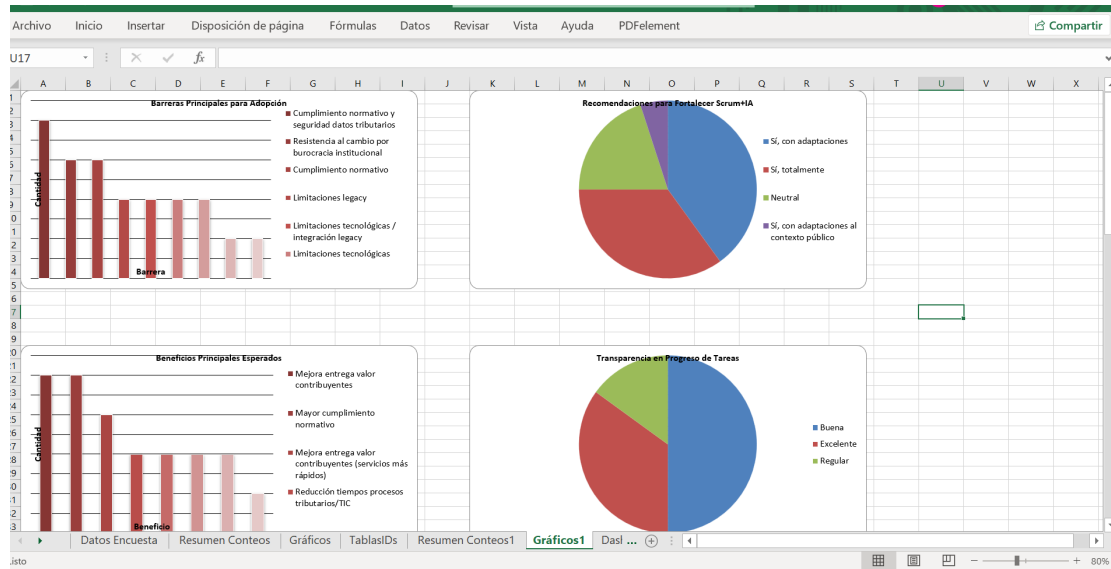
	J	K	L	M
	Beneficio_Principal	Errores_Dependencias_No_Detectadas	Mejora_Notificaciones_1_5	Importancia_Roles_Institucionales
1				
2	Mejora entrega valor contribuyentes (servicios más rápidos)	Frecuentemente	3	Importante
3	Mayor cumplimiento normativo y trazabilidad auditorías	Frecuentemente	5	Importante
4	Mejora entrega valor contribuyentes	Ocasionalmente	3	Importante
5	Reducción tiempos procesos tributarios/TIC	Frecuentemente	4	Importante
6	Mejora entrega valor contribuyentes	Ocasionalmente	5	Importante
7	Mayor cumplimiento normativo	Frecuentemente	3	Importante
8	Mejora entrega valor	Frecuentemente	3	Importante
9	Reducción tiempos procesos	Ocasionalmente	4	Importante
10	Mayor cumplimiento normativo	Ocasionalmente	5	Muy importante
11	Mejora entrega valor contribuyentes	Frecuentemente	3	Importante
12	Mayor cumplimiento normativo	Ocasionalmente	5	Importante
13	Reducción tiempos procesos	Frecuentemente	4	Importante
14	Mejora entrega valor	Frecuentemente	3	Muy importante
15	Mayor cumplimiento normativo	Ocasionalmente	5	Importante
16	Mejora entrega valor contribuyentes	Frecuentemente	5	Importante
17	Mayor cumplimiento normativo y trazabilidad para auditorías	Ocasionalmente	4	Muy importante
18	Mejora entrega valor contribuyentes (servicios más rápidos)	Frecuentemente	5	Importante
19	Reducción tiempos procesos tributarios/TIC	Ocasionalmente	3	Importante
20	Mayor cumplimiento normativo y trazabilidad para auditorías	Frecuentemente	4	Muy importante
21	Mejora entrega valor contribuyentes (servicios más rápidos)	Ocasionalmente	5	Importante
22				
23				
24				
25				
26				

Nota. Tablas plasmadas en Excel con los resultados obtenidos de las 20 encuestas realizadas a los funcionarios y equipo Scrum del SRI (Parte 3).

Figura 72*Resultados de las encuestas parte 4*

	N	O	P	Q
	Recomendaria_Fortalecer_Scrum_IA	Demora_Actualizaciones_Estado	Criticidad_Detectar_Retrasos_Temprano	Usaria_Sistema_Predictivo_Retrasos
1				
2	Si, con adaptaciones al contexto público	Ocasionalmente	Crítico	Definitivamente sí
3	Si, totalmente	Frecuentemente	Muy crítico	Probablemente sí
4	Neutral	Rara vez	Crítico	Definitivamente sí
5	Si, con adaptaciones	Ocasionalmente	Crítico	Probablemente sí
6	Si, totalmente	Frecuentemente	Muy crítico	Definitivamente sí
7	Si, con adaptaciones	Ocasionalmente	Crítico	Probablemente sí
8	Neutral	Frecuentemente	Muy crítico	Definitivamente sí
9	Si, con adaptaciones	Rara vez	Crítico	Probablemente sí
10	Si, totalmente	Ocasionalmente	Muy crítico	Definitivamente sí
11	Si, con adaptaciones	Frecuentemente	Crítico	Probablemente sí
12	Si, totalmente	Ocasionalmente	Muy crítico	Definitivamente sí
13	Neutral	Rara vez	Crítico	Probablemente sí
14	Si, con adaptaciones	Frecuentemente	Muy crítico	Definitivamente sí
15	Si, totalmente	Ocasionalmente	Crítico	Probablemente sí
16	Si, con adaptaciones	Frecuentemente	Muy crítico	Definitivamente sí
17	Si, totalmente	Rara vez	Crítico	Definitivamente sí
18	Si, con adaptaciones	Ocasionalmente	Muy crítico	Probablemente sí
19	Neutral	Frecuentemente	Crítico	Definitivamente sí
20	Si, totalmente	Ocasionalmente	Muy crítico	Probablemente sí
21	Si, con adaptaciones	Rara vez	Crítico	Definitivamente sí
22				
23				
24				
25				
26				

Nota. Tablas plasmadas en Excel con los resultados obtenidos de las 20 encuestas realizadas a los funcionarios y equipo Scrum del SRI (Parte 4).

Figura 73*Resultados de las encuestas parte 4*

Nota. Gráficos de pastel y barras representado en Excel con los resultados obtenidos de las 20 encuestas realizadas a los funcionarios y al equipo Scrum del SRI (Parte 4).

Figura 74

Fotocopia de las encuestas realizadas

Proyecto: CPS

ITI Tecnológico Internacional | MED Modalidad de Estudios En Línea

ENCUESTA PIG

CARRERA DE DESARROLLO DE SOFTWARE

- ¿Cuál es su rol principal en el SRI relacionado con proyectos o gestión de tareas? (Seleccione una)
 - a) Scrum Master o facilitador ágil
 - ☒ b) Miembro del equipo de desarrollo TIC/software
 - c) Product Owner o líder de área tributaria/proyecto
 - d) funcionario administrativo / Analista tributario
 - e) Supervisor / Director de área
 - f) Otro (especifique)
- ¿Con qué frecuencia participa en proyectos que usan metodología Scrum o ágil en el SRI?
 - ☒ a) Siempre (en la mayoría de proyectos)
 - b) Frecuentemente
 - c) Ocasionalmente
 - d) Rara vez
 - e) Nunca

Nota. Fotocopias de la encuesta realizada a los funcionarios y equipo Scrum del SRI (Parte 1).

Figura 75

Fotocopia de las encuestas realizadas

• ¿Ha utilizado Scrum en proyectos del SRI (ej. desarrollo de plataformas tributarias, SRI en Línea, ¿apps móviles)?

- a) Sí, activamente en varios proyectos
- b) Sí, en algunos
- ☒ c) Solo conocimiento teórico
- d) No lo he usado ni conozco

• ¿En proyectos del SRI, ¿qué tan importante considera la visibilidad en tiempo real del backlog de tareas?

- ☒ a) Muy importante
- b) Importante
- c) Neutral
- d) Poco importante
- e) Nada importante

• ¿Cuál es el principal desafío en la gestión de tareas en proyectos del SRI? (Seleccione hasta 2)

- a) Demoras por aprobaciones jerárquicas/burocracia
- b) Falta de visibilidad y transparencia en el backlog
- ☒ c) Dificultad para priorizar tareas tributarias urgentes
- d) Dependencias entre áreas (TIC vs. tributaria/normativa)
- ☒ e) Actualizaciones manuales lentas (correos, reuniones)
- f) Cumplimiento normativo que ralentiza sprints

Nota. Fotocopias de la encuesta realizada a los funcionarios y equipo Scrum del SRI (Parte 2).

Figura 76

Fotocopia de las encuestas realizadas

iti Tecnológico Internacional Universitario

- ¿Qué funcionalidad priorizaría en un sistema inteligente de gestión de tareas para el SRI? (Seleccione hasta 3)
 - a) Priorización automática basada en impacto tributario/normativo
 - b) Alertas inteligentes de riesgos y dependencias burocráticas
 - ☒ c) Dashboard personalizado por rol (con métricas institucionales)
 - d) Integración con sistemas existentes (SRI en Línea, RUC, facturación)
 - ☒ e) Reportes automáticos para auditorías y dirección
 - ☒ f) Predicción de retrasos en sprints por datos históricos del SRI
- ¿Cómo califica la transparencia en el progreso de tareas en proyectos del SRI actuales?
 - a) Excelente
 - ☒ b) Buena
 - c) Regular
 - d) Mala e) Muy mala

Nota. Fotocopias de la encuesta realizada a los funcionarios y equipo Scrum del SRI (Parte 3).

Figura 77

Fotocopia de las encuestas realizadas

ITI Tecnológico Internacional Universitario

- ¿Qué tan frecuente es la demora en actualizaciones de estado de tareas por procesos manuales o jerárquicos en el SRI?
 - a) Siempre
 - b) Frecuentemente
 - c) Ocasionalmente
 - ☒ d) Rara vez
 - e) Nunca
- En proyectos TIC/tributarios del SRI, ¿qué tan crítico es detectar riesgos de retraso temprano (ej. por validaciones normativas)?
 - ☒ a) Muy crítico
 - b) Crítico
 - c) Moderado
 - d) Poco crítico
- ¿Usaría un sistema que prediga retrasos en sprints usando datos históricos del SRI (ej. demoras por importaciones normativas o aprobaciones)?
 - a) Definitivamente sí
 - ☒ b) Probablemente sí
 - c) Neutral
 - d) Probablemente no
 - e) Definitivamente no

Nota. Fotocopias de la encuesta realizada a los funcionarios y equipo Scrum del SRI (Parte 4).

Figura 78

Fotocopia de las encuestas realizadas

ITI Tecnológico Internacional Universitario

- ¿Qué tan importante es que el sistema respete roles institucionales del SRI (ej. Product Owner como líder tributario, Scrum Master facilitando burocracia)?
 - ☒ a) Muy importante
 - b) Importante
 - c) Neutral
 - d) Nada Importante
- ¿Cuál es el principal beneficio esperado de un sistema inteligente de gestión de tareas en el SRI? (Seleccione una)
 - a) Reducción de tiempos en procesos tributarios/TIC
 - b) Mayor cumplimiento normativo y trazabilidad para auditorías
 - ☒ c) Mejora en la entrega de valor a contribuyentes (ej. servicios digitales más rápidos)
 - d) Menos estrés y sobrecarga en equipos
 - e) Mejor colaboración interáreas (TIC vs. áreas tributarias)

Nota. Fotocopias de la encuesta realizada a los funcionarios y equipo Scrum del SRI (Parte 5).

Figura 79

Fotocopia de las encuestas realizadas

ITI Tecnológico Internacional Universitario

- ¿Con qué frecuencia enfrenta errores por dependencias no detectadas entre tareas (ej. desarrollo vs. validación normativa)?
 - a) Siempre
 - b) Frecuentemente
 - ☒ c) Ocasionalmente
 - d) Rara vez
 - e) Nunca
- En una escala del 1 al 5, ¿cuánto mejoraría la eficiencia en el SRI con notificaciones inteligentes en tiempo real y alertas de bloqueos?
(1 = Nada, 4 5 = Muchísimo)
- ¿Recomendaría implementar o fortalecer Scrum con herramientas inteligentes en más proyectos del SRI?
 - a) Sí, totalmente
 - ☒ b) Sí, con adaptaciones al contexto público
 - c) Neutral
 - d) No, prefiero métodos tradicionales
 - e) No sé

Nota. Fotocopias de la encuesta realizada a los funcionarios y equipo Scrum del SRI (Parte 6).

Figura 80

Fotocopia de las encuestas realizadas

ITI Tecnológico Internacional Universitario

- ¿Qué barrera principal ve para adoptar un sistema inteligente de gestión de tareas en el SRI? (Seleccione una)
 - a) Cumplimiento normativo y seguridad de datos tributarios
 - b) Resistencia al cambio por burocracia institucional
 - ☒ c) Limitaciones tecnológicas o integración con sistemas legacy
 - d) Falta de capacitación en metodologías ágiles
 - e) Recursos/tiempo insuficiente
- ¿Qué tan probable es que use/adopte un sistema de este tipo si se integra con plataformas del SRI (ej. SRI en Línea, autenticación institucional)?
 - ☒ a) Muy probable
 - b) Probable
 - c) Neutral
 - d) Poco probable
 - e) Nada probable

Nota. Fotocopias de la encuesta realizada a los funcionarios y equipo Scrum del SRI (Parte 7).

Ficha Técnica Utilizada: Análisis de Causa y Efecto

Informe Anti plagio

Aspectos positivos del instrumento:

- Preguntas bien estructuradas y alineadas con objetivos.
- Combinación equilibrada de tipos de pregunta para datos ricos y cuantificables.
- Enfoque en necesidades reales del SRI (normativas, burocracia, colaboración).

Recomendaciones para mejorar el instrumento:

- Incluir una pregunta cuantitativa adicional sobre frecuencia de bloqueos por dependencia normativa (escala 1-5).
- Agregar una sección opcional para ejemplos concretos de tareas problemáticas, enriqueciendo el análisis cualitativo.

Docente revisor

Nombre del Docente

Validación de Instrumento Metodológico

Se empleó el análisis de causa y efecto (diagrama de Ishikawa o espina de pescado) para detectar las principales dificultades que enfrentan los equipos Scrum en la gestión de tareas dentro del SRI. Mediante esta técnica se identificaron causas raíz como: ausencia de una herramienta digital inteligente y centralizada, falta de visibilidad en tiempo real de dependencias y bloqueos, procesos manuales propensos a errores normativos, demoras por aprobaciones jerárquicas y sobrecarga operativa en validaciones repetitivas. Estas causas derivaron directamente en la formulación del problema y los objetivos del proyecto, orientados a diseñar e implementar un gestor inteligente de tareas que resuelva integralmente cada uno de estos aspectos.

Parámetros de evaluación	Observaciones: colocar sí o no y el argumento de su revisión
¿El instrumento cuenta con encabezado institucional?	SI ☐ NO ☐ Argumento:
¿El instrumento tiene escrito el objetivo que persigue el instrumento?	SI ☐ NO ☐ Argumento:
¿El instrumento tiene las instrucciones claras para su aplicación?	SI ☐ NO ☐ Argumento:
¿El formato de preguntas es correcto en su orden y numeración?	SI ☐ NO ☐ Argumento:
¿Las preguntas están formuladas con lenguaje sencillo y entendible?	SI ☐ NO ☐ Argumento:
¿Las preguntas formuladas son?	Comprensibles ☐ Cuasi comprensibles ☐ Confusas ☐ Incomprensibles ☐ Argumento:
¿El tipo de preguntas (cerradas, abiertas o mixtas) permitirán que las respuestas respondan al tema propuesto?	SI ☐ NO ☐ Argumento:

¿El número de preguntas planteadas son suficientes para obtener información que denote las estadísticas?	SI ☐ NO ☐ Argumento:
¿El tiempo establecido para la aplicación del instrumento es suficiente para su complejidad?	SI ☐ NO ☐ Argumento:
¿El público seleccionado es adecuado para el instrumento que se pretende aplicar?	SI ☐ NO ☐ Argumento:
¿El instrumento está listo para ser aplicado?	SI ☐ NO ☐ Argumento:
Señale los aspectos positivos del instrumento	
Emita las recomendaciones necesarias para mejorar el instrumento en caso de ser necesario.	

Docente revisor

Nombre del Docente