

ISIST-2430-12

**Plataforma de soporte para la evaluación de proyectos de investigación presentados
ante el Comité de Ética de la Investigación del Hospital Universitario San Ignacio**



Jose Manuel Rodríguez
Laura Valentina Ovalle
Michael Joel González
María Paula Cardona

PONTIFICIA UNIVERSIDAD JAVERIANA
FACULTAD DE INGENIERÍA
PROGRAMA DE INGENIERÍA DE SISTEMAS
BOGOTÁ, D.C.
2025

ISIST-2430-12

Plataforma de soporte para la evaluación de proyectos de investigación presentados ante el
Comité de Ética de la Investigación del Hospital Universitario San Ignacio

Autor(es):

Jose Manuel Rodríguez

Laura Valentina Ovalle

Michael Joel González

María Paula Cardona

MEMORIA FINAL DE GRADO REALIZADA PARA CUMPLIR CON UNO DE LOS
REQUISITOS PARA LA TITULACIÓN DE INGENIERÍA DE SISTEMAS

Director

Ing. Juan Erasmo Gómez Morantes

Jurados del Trabajo Fin de Grado

Ing. Alexandra Pomares Quimbaya

Ing. Andrés Oswaldo Calderón Romero

PONTIFICIA UNIVERSIDAD JAVERIANA

FACULTAD DE INGENIERÍA

PROGRAMA DE INGENIERÍA DE SISTEMAS

BOGOTÁ, D.C.

Mayo, 2025

**PONTIFICIA UNIVERSIDAD JAVERIANA
FACULTAD DE INGENIERÍA
PROGRAMA DE INGENIERÍA DE SISTEMAS**

Rector de la Pontificia Universidad Javeriana
Padre Luis Fernando Múnera Congote, S.J.

Decano de la Facultad de Ingeniería
Ing. Diego Alejandro Patiño Guevara, MA

Director de Carrera de Ingeniería de Sistemas
Ing. Carlos Andrés Parra Acevedo, PhD

Director del Departamento de Ingeniería de Sistemas
Ing. César Julio Bustacara Medina, PhD

Artículo 23 de la Resolución No. 1 de junio de 1946

“La Universidad no se hace responsable de los conceptos emitidos por sus alumnos en sus proyectos de grado. Sólo velará porque no se publique nada contrario al dogma y la moral católica y porque no contengan ataques o polémicas puramente personales. Antes bien, que se vean en ellos el anhelo de buscar la verdad y la Justicia”

AGRADECIMIENTOS

Laura Valentina Ovalle Benítez

Es un honor para mí expresar mi más profundo y sincero agradecimiento, comenzando con mi familia, quienes han sido el soporte fundamental en mi camino académico y personal. A mi madre, Claudia Benítez, gracias por creer incondicionalmente en mí, en mis habilidades y capacidades; por enseñarme siempre a ver más allá de las dificultades y ofrecerme palabras de aliento en cada paso. A mi padre, Francisco Ovalle, mi más sincera gratitud por ser una fuente constante de sabiduría y consejos que me guiaron en la toma de decisiones importantes a lo largo de mi carrera. A mi hermana, Mariana Ovalle, gracias por estar siempre presente, por escucharme con paciencia y brindarme ánimo y comprensión en mis momentos más difíciles.

A mis compañeros de tesis, quienes fueron clave para lograr este importante objetivo. A María Paula Cardona, por tu energía inagotable, por siempre mostrar disposición y por no desistir jamás ante las adversidades. A Michael González, agradezco profundamente la calma y serenidad que aportaste, especialmente en los momentos más críticos, ayudándonos a mantenernos centrados en nuestro propósito. A José Manuel Rodríguez, gracias por tu disciplina excepcional y dedicación constante, cualidades esenciales que permitieron entregar nuestro mejor proyecto realizado hasta hoy.

Extiendo un especial reconocimiento a la Pontificia Universidad Javeriana, institución que me brindó no solo formación académica de excelencia, sino también un espacio de crecimiento personal y profesional invaluable. Gracias por ser el lugar que impulsó mi desarrollo integral y me preparó para afrontar con éxito los desafíos del futuro.

De manera muy especial, expreso mi agradecimiento a Juan Erasmo Gómez Morantes, mi director de tesis, quien además de ofrecerme su guía académica, creyó profundamente en mis habilidades y en mi potencial. Su confianza y apoyo constante hicieron posible que este proyecto culminara exitosamente, brindándome enseñanzas valiosas que atesoraré siempre.

Finalmente, agradezco sinceramente a todas las personas que directa o indirectamente han formado parte de mi vida. Tanto en las experiencias positivas como en las difíciles, he encontrado aprendizajes significativos que hoy me fortalecen. Este logro es solo el comienzo de un camino lleno de oportunidades y nuevos retos.

Gracias de corazón a todos, pues sin su apoyo, inspiración y acompañamiento, este proyecto no habría sido posible.

María Paula Rodríguez Cardona

Es un honor para mí expresar mi más profundo y sincero agradecimiento a todas las personas que, de una u otra forma, hicieron posible la culminación de este proyecto y de mi etapa universitaria. En primer lugar, agradezco a Dios, por acompañarme a lo largo de este camino, por darme fortaleza en los momentos difíciles, por guiarme con sabiduría y levantarme cuando mis propias fuerzas ya no eran suficientes.

A mis padres, Jairo Alberto Cardona y Ana Karina Rodríguez, por ser el pilar de mi vida. Gracias por confiar siempre en mí, incluso en la distancia. Por su amor incondicional y por enseñarme que la perseverancia, la fe y la dedicación pueden superar cualquier obstáculo. Su apoyo constante, sus palabras de aliento y su presencia emocional en cada etapa de esta carrera fueron fundamentales para que nunca dejara de luchar por mis sueños. A mis hermanos, Santiago, María José y Juan José Cardona, gracias por ser mi motor y mi alegría diaria, por acompañarme con su cariño a distancia, sus mensajes y su energía, y por recordarme siempre por qué vale la pena seguir adelante.

A mis tíos, John Cardona y Diana Torres, mi más profundo agradecimiento por acogerme en su hogar durante este trayecto en Bogotá. Gracias por tratarme como a una hija, por brindarme un espacio de tranquilidad, cuidado y cariño que me permitió avanzar en este proceso lejos de casa, pero sin sentirme sola.

A nuestro director de tesis, Juan Erasmo Gómez Morantes, gracias por guiarnos con compromiso, paciencia y visión. Su acompañamiento nos permitió mantener el rumbo incluso cuando surgían dificultades técnicas o personales, y su confianza en nuestro potencial fue un impulso invaluable para seguir adelante.

A mis compañeros de tesis, Laura Valentina Ovalle, José Manuel Rodríguez y Michael Joel González, gracias por creer en mí, por permitirme compartir este reto académico y por convertir este proceso en una verdadera aventura de crecimiento colectivo. Agradezco profundamente su entrega, su responsabilidad y la forma en que, juntos, enfrentamos cada desafío con compromiso, empatía y trabajo en equipo.

A mis entrenadores y compañeras de equipo, gracias por construir conmigo una segunda familia lejos de casa. Gracias por enseñarme disciplina, resiliencia, trabajo en equipo y por brindarme un espacio para crecer también como persona.

Y finalmente, a mis demás familiares, profesores, amigos, personas cercanas y a mi pareja, gracias por su compañía, cariño y apoyo incondicional durante estos cuatro años. Gracias por animarme cuando flaqueaba, por celebrar conmigo cada avance, y por ayudarme a levantarme cada vez que caí. Su presencia constante fue luz en los momentos difíciles y alegría en los momentos de logro.

José Manuel Rodríguez Ardila

Es un honor para mí expresar mi más profundo agradecimiento a todas las personas que, de una u otra forma, hicieron posible la culminación de esta tesis. En primer lugar, agradezco a mi director de tesis, Juan Erasmo, por su guía paciente y la serenidad con la que nos condujo en cada etapa, aportando siempre claridad y experiencia. A mi padre, Jorge Ignacio Rodríguez, por su sabiduría y por ser un ejemplo constante de perseverancia, y a mi madre Olga Patricia Ardila, por su cariño incondicional y por ser un modelo inspirador que me enseñó con su entrega la importancia del compromiso y la dedicación.

También quiero dar las gracias a María Andrea Méndez, mi novia, que nunca dejó de creer en mí y me brindó apoyo emocional en los momentos más difíciles. A mis compañeras y compañero de tesis: a Laura Valentina Ovalle, por su compromiso incansable, dedicación y sus buenas energías incluso cuando todo parecía gris; a María Paula Cardona, por su positivismo y energía inagotable; y a Michael Joel González, por transmitir paz y tranquilidad en cada paso del proceso. A mi mejor amigo, Andrés David Mendoza, quien se convirtió en parte de mi familia, me impulsó a ser mejor cada día y me acogió en su hogar ofreciéndome un espacio de paz y concentración.

De igual modo, agradezco a mi compañero de trabajo, Juan Pablo Sánchez, por cubrirme en los momentos más críticos, por las risas compartidas y por sus consejos de alto valor que aligeraron la carga y me permitieron dedicarme casi por completo a este proyecto.

Extiendo mi gratitud a la Pontificia Universidad Javeriana por brindarme un entorno de excelencia académica y de crecimiento personal, así como a Santiago Castro, cuya motivación constante para aprender, iniciar nuevos proyectos y explorar tecnologías influyó de manera decisiva en el desarrollo de este proyecto. Con ello, cierro esta etapa llena de esfuerzos y aprendizajes y me dispongo, con ilusión, a dar inicio a la siguiente.

Michael Joel Gonzalez Díaz

Deseo expresar mi más sincero agradecimiento a todas las personas e instituciones que hicieron posible la culminación de este proyecto de grado.

En primer lugar, a mi familia, por su apoyo constante y confianza en mis capacidades a lo largo de mi formación académica. Su aliento y comprensión han sido un pilar fundamental. De manera especial a mis padres, Jair Alberto Gonzalez Mayorga y Claudia Diaz Galvis, por su invaluable respaldo y por inculcarme el valor de la perseverancia. A mi hermana Michel Gonzalez Diaz, gracias por tu apoyo incondicional y por siempre estar presente con una palabra de aliento.

Agradezco de manera especial a nuestro director de tesis, Juan Erasmo Gomez Morantes. su guía, paciencia y visión crítica fueron cruciales para orientar nuestro trabajo y superar los desafíos que se presentaron. Su confianza en nuestro equipo fue un gran motivador y su serenidad, un ejemplo a seguir.

A mis compañeros de tesis, Laura Valentina Ovalle, María Paula Cardona y Jose Manuel Rodriguez. Cada uno fue una pieza fundamental en este rompecabezas; nuestras habilidades y personalidades se complementaron de manera excepcional, permitiéndonos construir un proyecto sólido y coherente. Gracias por el compromiso, la dedicación y el excelente trabajo en equipo que demostramos. A Laura Valentina, tu incansable energía y meticulosidad fueron el motor que impulsó la calidad y el detalle de nuestro trabajo. A Maria Paula, tu positivismo constante y gran disposición fueron esenciales para mantener y el ánimo del equipo. A Jose Manuel, tu notable dedicación y rigurosidad de tus aportes aportaron a la estructura y solidez necesarias para el éxito del proyecto.

Afrontar este proyecto juntos hizo que el proceso fuera no solo exitoso, sino también una valiosa experiencia de colaboración y crecimiento profesional, donde cada aporte individual nos fortaleció como conjunto.

Extiendo mi gratitud a la Pontificia Universidad Javeriana, por brindarme una formación académica de calidad y por ser el espacio donde pude desarrollar mis habilidades profesionales y personales, preparándome para retos futuros.

Finalmente, agradezco a todos aquellos amigos, profesores y personas que, de una u otra forma, me brindaron su apoyo, consejo o simplemente una palabra de aliento durante esta etapa. Cada contribución, grande o pequeña, fue importante.

Gracias a todos.

TABLA DE CONTENIDO

INTRODUCCIÓN.....	12
DESCRIPCIÓN GENERAL.....	13
1. Oportunidad, Problema.....	13
1.1. Contexto del problema.....	14
1.1.1. Definición del contexto.....	14
1.1.2. Descripción del contexto.....	15
1.2. Formulación del problema.....	15
1.3. Propuesta de solución.....	17
1.4. Justificación de la solución.....	18
2. Descripción del proyecto.....	20
2.1. Objetivo general.....	20
2.2. Objetivos específicos.....	20
2.3. Entregables, Estándares y Justificación.....	20
2.4. Metodología.....	22
Fase 1: Levantamiento de requerimientos.....	22
Fase 2: Desarrollo iterativo de funcionalidades principales.....	22
Fase 3: Integración del modelo de inteligencia artificial.....	23
Fase 4: Pruebas funcionales y preparación de entrega.....	23
Fase 5: Cierre del proyecto.....	23
CONTEXTO DEL PROYECTO.....	23
1. Marco teórico.....	24
2. Análisis del contexto.....	25
ANÁLISIS DEL PROBLEMA.....	27
1. Requerimientos.....	28
1.1. Requerimientos no funcionales.....	28
Interfaces con el usuario.....	28
Interfaces con el hardware.....	30
Interfaces con el software.....	31
Interfaces de comunicaciones.....	32
1.2. Requerimientos funcionales.....	33
Funcionalidad de Autenticación y registro.....	33
Funcionalidad de Gestión de archivos (Evaluador).....	35
Funcionalidad de Interacción con el modelo de IA.....	36
Funcionalidad de Análisis y estadísticas.....	39
Funcionalidad de Configuración del modelo de IA.....	40
Funcionalidad de Gestión del consentimiento (Investigador).....	41
2. Restricciones.....	42
2.1. Restricciones de Hardware.....	43
2.2. Restricciones de Software.....	43
2.3. Restricciones impuestas por el cliente.....	44
2.5. Restricciones generales.....	45
3. Análisis Costo Operacional.....	46

3.1. Costo de Infraestructura.....	47
3.2. Costo de Mantenimiento.....	48
4. Especificación Funcional.....	48
4.1. Actores.....	49
4.2. Historias de usuario.....	49
DISEÑO DE LA SOLUCIÓN.....	51
1. Arquitectura del Sistema.....	51
1.1. Vista lógica.....	53
1.2. Vista de procesos.....	55
Proceso 1: Crear nuevo caso.....	56
Proceso 2: Previsualizar o eliminar un caso.....	56
Proceso 3: Subida de archivos y creación de evaluaciones.....	57
Proceso 4: Gestión de evaluaciones.....	57
Proceso 5: Visualización del resultado y envío por correo.....	58
Proceso 6: Configuración de proveedor de IA, modelo, API Key y prompts.....	58
Proceso 7: Visualización de estadísticas de evaluación.....	58
1.3. Vista física.....	59
DESARROLLO DE LA SOLUCIÓN.....	62
1. Implementación.....	62
2. Producto Final.....	62
2.1. Interfaces del Login.....	63
2.2. Interfaces Evaluador.....	64
2.3. Interfaces Investigador.....	68
VALIDACIÓN Y RESULTADOS.....	70
1. Pruebas de Software.....	70
1.1. Metodología de evaluación.....	70
2. Resultados Pruebas.....	71
2.1. Pruebas de Integración.....	71
2.2. Pruebas de Carga y estrés.....	71
2.3. Pruebas End-to-end.....	72
2.4. Pruebas de aceptación del usuario (UAT).....	73
3. Validación Arquitectónica.....	73
4. Validación de Modelo IA.....	74
CONCLUSIONES.....	75
1. Conclusiones.....	75
2. Análisis del impacto.....	76
3. Trabajos Futuros.....	76
REFERENCIAS.....	77
ANEXOS.....	80

ABSTRACT

At Hospital Universitario San Ignacio, the ethical evaluation of informed consent documents for research projects still relies on manual reviews, leading to delays and inconsistencies due to human errors. To address this issue, a web platform was designed and implemented to comprehensively manage the process, from document creation to evaluation by the Ethics Committee. This tool automates the generation of PDF documents and email notifications. Additionally, it incorporates an artificial intelligence model that aids evaluators in swiftly detecting regulatory non-compliance, along with an analytics dashboard that provides metrics on process durations and evolution. Developed using the Scrum methodology, this platform significantly enhances the efficiency, transparency, and security of the evaluation process, serving as a replicable model for other ethics committees.

RESUMEN

En el Hospital Universitario San Ignacio, la evaluación ética de documentos de consentimiento informado pertenecientes a proyectos de investigación depende todavía de revisiones manuales que generan retrasos e inconsistencias debido a errores humanos. Para abordar este problema, se diseñó e implementó una plataforma web que gestiona integralmente el proceso, desde la creación de dichos documentos hasta su revisión por parte del Comité de Ética. Esta herramienta automatiza la generación de documentos PDF y notificaciones por correo electrónico. Además, incorpora un modelo de inteligencia artificial que asiste al evaluador en la detección rápida de incumplimientos normativos, así como un panel de analítica que ofrece métricas sobre tiempos y evolución del proceso. Desarrollada con la metodología Scrum, la plataforma busca mejorar significativamente la eficiencia, transparencia y seguridad del proceso de evaluación, constituyendo un modelo replicable para otros comités de ética.

INTRODUCCIÓN

La investigación científica en seres humanos, es uno de los pilares fundamentales para el avance del conocimiento y la mejora de la calidad de vida en las sociedades contemporáneas. No obstante, su desarrollo debe estar regulado por principios éticos que salvaguarden la dignidad, los derechos y el bienestar de los participantes involucrados. En este sentido, los comités de ética cumplen una función esencial: garantizar que cada proyecto cumpla con los lineamientos normativos y éticos establecidos tanto a nivel institucional como nacional e internacional. Sin embargo, en los últimos años, el incremento significativo en la cantidad y complejidad de los proyectos presentados ha superado la capacidad operativa de muchos de estos comités, generando retrasos, cuellos de botella y una creciente carga administrativa para sus miembros.

Este fenómeno se hace particularmente evidente en el Hospital Universitario San Ignacio (HUSI), institución adscrita a la Pontificia Universidad Javeriana, donde el Comité de Ética recibe más de 200 solicitudes de evaluación al año. Cada una de estas solicitudes exige un análisis detallado, en especial de los documentos de consentimiento informado, los cuales constituyen la base legal y ética para la participación voluntaria de los sujetos en investigaciones científicas. A pesar de existir formatos y plantillas definidas, estos documentos suelen presentar errores de forma o faltas de completitud que obligan a repetir el proceso de revisión, lo que ralentiza la aprobación de los estudios y afecta la capacidad de los equipos de investigación para avanzar conforme a sus cronogramas.

Actualmente, la verificación de los consentimientos informados se realiza de manera manual, con base en un formato de evaluación que obliga al evaluador a clasificar distintos criterios como “cumple”, “no cumple” o “no aplica”, de acuerdo con la normativa ética vigente. Esta tarea, aunque necesaria, es altamente demandante en términos de tiempo y susceptible a errores humanos, especialmente cuando se enfrenta a un volumen elevado de documentos. Además, la falta de herramientas que acompañen a los investigadores en la redacción de los consentimientos propicia fallos reiterativos que podrían evitarse desde la fase inicial.

Frente a este panorama, el presente proyecto propone el diseño e implementación de una plataforma web de soporte para la evaluación ética de proyectos de investigación, enfocada específicamente en la verificación de los documentos de consentimiento informado. La solución busca asistir tanto a evaluadores como a investigadores, a través de dos ejes fundamentales:

- La integración de modelos de inteligencia artificial (LLMs) que analicen automáticamente los documentos de consentimiento informado, en busca de incumplimientos normativos
- El desarrollo de un módulo de apoyo a la construcción del consentimiento, que oriente al investigador en la redacción adecuada y completa del documento desde el inicio.

La plataforma ha sido desarrollada bajo la metodología ágil Scrum, con un enfoque centrado en la mejora de la eficiencia, la trazabilidad del proceso y la calidad de las evaluaciones, sin comprometer la meticulosidad ética exigida. Asimismo, incorpora funcionalidades como la generación automática de PDFs, el envío de resultados por correo electrónico y un panel de analítica para el seguimiento del desempeño del comité, respecto a este proceso de evaluación. Esta propuesta no solo representa una solución tecnológica innovadora para el Hospital Universitario San Ignacio, sino que también ofrece un modelo replicable y escalable para otros comités de ética en Colombia y la región, en busca de procesos más ágiles, confiables y transparentes.

DESCRIPCIÓN GENERAL

En este capítulo se ofrece una visión general del proyecto, comenzando con la identificación del problema que motivó su desarrollo: la carga operativa y las demoras en la revisión manual de los consentimientos informados por parte del Comité de Ética del Hospital Universitario San Ignacio. Se contextualiza la necesidad institucional, se formula el problema, y se presenta la solución propuesta: una plataforma web con soporte de inteligencia artificial que agiliza el proceso de evaluación ética. Además, se describen los objetivos del proyecto, sus principales entregables, y la metodología ágil utilizada para su implementación, sentando las bases para comprender el diseño y alcance de la solución desarrollada.

1. Oportunidad, Problema

En el ámbito de la investigación científica, sobre todo en instituciones académicas y hospitales universitarios, los comités de ética desempeñan aún un papel fundamental. Su responsabilidad se basa en garantizar que los proyectos de investigación cumplan los más altos estándares éticos, protegiendo así la integridad y el bienestar de las personas que participan en estos. Sin embargo, en los últimos años, estos comités se enfrentan a desafíos debido al aumento en la cantidad y complejidad de proyectos presentados para evaluación.

Gracias a esto, se ha generado una sobrecarga de trabajo para los miembros de los comités, quienes se encargan de revisar las propuestas para asegurar que vayan acorde a las normativas éticas establecidas. Uno de los principales aspectos a mejorar en este proceso es la verificación de los documentos de consentimiento informado, ya que la revisión manual de estos documentos se ha convertido en una tarea laboriosa y propensa a errores, lo que puede llegar a comprometer la validez de los resultados, la seguridad y los derechos de los participantes.

Todo esto ha hecho que surja la oportunidad de implementar soluciones tecnológicas avanzadas que, por un lado, agilicen el proceso de evaluación y, por otro, optimicen la precisión del mismo. En este sentido, la integración de herramientas basadas en inteligencia artificial no sólo puede disminuir el esfuerzo requerido para la revisión de los consentimientos informados, sino que también contribuye a reducir significativamente el tiempo dedicado a esta tarea. Además, permite minimizar la posibilidad de errores durante el proceso. Esta solución, por ende, no solo mejoraría la eficiencia operativa en los comités de ética, sino que, a su vez, fortalecería la protección de los participantes en los proyectos y elevaría la calidad de la investigación científica.

1.1. Contexto del problema

El Hospital Universitario San Ignacio (HUSI), vinculado a la Pontificia Universidad Javeriana, se ha consolidado en los últimos años como un referente en el ámbito de la investigación médica en Colombia. Su Comité de Ética recibe anualmente más de 200 solicitudes de revisión de protocolos de investigación, cada una de las cuales exige una evaluación detallada y rigurosa de los formularios de consentimiento informado.

Hasta ahora, este procedimiento se ha desarrollado de forma manual, lo que implica una alta demanda de tiempo, atención y recursos humanos por parte de los evaluadores. Esta dinámica ha derivado en demoras constantes en la aprobación de proyectos, dificultando el cumplimiento de los cronogramas establecidos por los investigadores, retrasando el desarrollo de estudios clave y aumentando el riesgo de errores o inconsistencias en la revisión ética de los documentos. En consecuencia, se ha evidenciado la necesidad urgente de implementar herramientas tecnológicas que contribuyan a optimizar este proceso sin comprometer su integridad ni su rigurosidad.

1.1.1. Definición del contexto

La evaluación ética de los proyectos de investigación representa un pilar fundamental para salvaguardar los derechos, la dignidad y el bienestar de los participantes involucrados. En el contexto colombiano, los Comités de Ética se encargan de validar cada protocolo a la luz de principios éticos internacionales y normativas nacionales vigentes, con especial énfasis en el contenido del consentimiento informado, un documento clave que debe detallar claramente los objetivos del estudio, los procedimientos a realizar, los posibles riesgos y beneficios, las garantías de confidencialidad y el derecho de los participantes a retirarse en cualquier momento.

En el caso del HUSI, el crecimiento sostenido y la creciente diversidad de los proyectos sometidos a evaluación intensificaron de manera significativa esta labor. Como resultado, se hizo evidente que el alto volumen y la complejidad de los documentos de consentimiento informado comenzaban a superar la capacidad de respuesta del comité bajo un modelo de revisión manual, sin

dejar de exigir el nivel de rigurosidad que esta tarea demanda. Esta situación planteó la necesidad de buscar nuevas estrategias que permitieran mantener los estándares éticos sin comprometer la eficiencia operativa.

1.1.2. Descripción del contexto

El crecimiento sostenido de la producción científica en el HUSI generó un importante cuello de botella operativo en torno a la gestión documental de los consentimientos informados. Aunque existían lineamientos bien definidos para su elaboración y criterios claros para su evaluación, el Comité de Ética no contaba con una plataforma digital que permitiera sistematizar y optimizar este proceso de forma efectiva. La diversidad en los formatos utilizados por los investigadores, así como las variaciones en la redacción y estructura de los documentos, dificultaban la estandarización de las evaluaciones.

A esto se sumaba la subjetividad inherente al análisis humano, que derivan en decisiones poco homogéneas, posibles omisiones y falta de trazabilidad en las observaciones realizadas. Frente a esta situación, se identificó la necesidad de implementar una solución tecnológica robusta, que integre formularios dinámicos, validaciones automáticas y un componente de inteligencia artificial capaz de detectar omisiones normativas, mejorar la uniformidad en los criterios de revisión y aumentar la eficiencia general del proceso, sin comprometer el juicio ético de los evaluadores.

1.2. Formulación del problema

El creciente número de proyectos de investigación presentados al Comité de Ética del Hospital Universitario San Ignacio (HUSI), de la Pontificia Universidad Javeriana, ha generado una carga de trabajo cada vez mayor para sus miembros, especialmente en lo relacionado con la gestión y revisión de dichas propuestas.

Este aumento en la demanda ha ocasionado demoras significativas en la aprobación de estudios, no solo en el HUSI, sino también en otros centros de investigación del país. Esta situación impacta negativamente el avance de los proyectos, dificultando el cumplimiento de los cronogramas establecidos por los investigadores y reduciendo la capacidad de los equipos de investigación para progresar de manera eficiente y oportuna.

Actualmente, el Comité de Ética del HUSI recibe más de 200 solicitudes de revisión al año, cada una de las cuales requiere una evaluación ética detallada y rigurosa. Este proceso se lleva a cabo de manera manual, lo que lo convierte en una tarea altamente demandante y susceptible a errores humanos.

La gestión documental de estos procesos en el Hospital a día de hoy se apoya en el sistema Fundanet, una herramienta indispensable ya que en ella se centraliza la administración de todos los documentos de ética. Es en Fundanet donde el investigador carga la totalidad de los documentos correspondientes a su proyecto. Posteriormente, el evaluador accede a esta plataforma para extraer específicamente el

consentimiento informado, el cual es sometido a una evaluación manual. Finalmente, una vez que el comité ha concluido la evaluación integral del proyecto, el evaluador se encarga de actualizar el estado correspondiente dentro del mismo sistema Fundanet.

La ausencia de un sistema ágil que facilite la verificación de los documentos implica que la revisión de cada proyecto puede tardar más de lo necesario, afectando tanto la eficiencia como la calidad del proceso. Además, el carácter manual de esta revisión conlleva el riesgo de inconsistencias o sesgos, dado que no siempre es realizada por los mismos evaluadores y puede depender del juicio individual de cada uno.

En el contexto del presente proyecto, nos centraremos en una parte crítica del proceso: la verificación y generación de los documentos de consentimiento informado. Habiendo dicho esto, es importante mencionar que un consentimiento informado es un documento legal que los participantes de un estudio deben firmar, indicando que han sido debidamente informados sobre los objetivos, procedimientos, riesgos y beneficios del estudio, así como sobre sus derechos, incluida la posibilidad de retirarse en cualquier momento. Este documento es fundamental para garantizar que los estudios clínicos y de investigación se realicen de manera ética, protegiendo los derechos y el bienestar de los participantes.

A pesar de que estos consentimientos siguen una estructura definida y se basan en normativas específicas y plantillas que orientan su evaluación, el proceso continúa siendo altamente demandante desde el punto de vista administrativo.

El formato que deben diligenciar los evaluadores, donde se registran los criterios de cumplimiento normativo, se encuentra disponible en el **Anexo 15** correspondiente al final del documento. Como se observa, la verificación de estos consentimientos implica clasificar cada criterio como “cumple”, “no cumple” o “no aplica” para cada norma aplicable. Este proceso exige una revisión minuciosa, ya que el evaluador debe identificar y validar todos los elementos normativos relevantes según el tipo de estudio.

Adicional a esto, el formato de documento que se tiene a evaluar, se puede observar en el **Anexo 14**. Aunque una parte considerable del contenido está prediseñado, existen múltiples campos que deben ser diligenciados manualmente por el investigador. Con frecuencia, estos campos no son completados de manera adecuada o son dejados en blanco, lo que genera retrasos adicionales en el proceso de evaluación. En tales casos, el evaluador se ve en la obligación de devolver el documento para su corrección y, posteriormente, revisarlo nuevamente. Actualmente, una proporción significativa de los consentimientos informados es devuelta por este motivo, lo que impacta directamente en la eficiencia del proceso.

Esta situación se ve agravada por la extensión y complejidad de los documentos, ya que el hecho de tener que realizar una segunda revisión implica una duplicación del trabajo y contribuye a la acumulación de tareas pendientes. Así, el proceso manual de verificación no solo resulta lento, sino también altamente susceptible a errores e inconsistencias, especialmente en un contexto de alta carga operativa.

Esta problemática ha generado un cuello de botella en la aprobación y ejecución de proyectos de investigación, limitando el desarrollo oportuno de estudios clínicos y

reduciendo la capacidad de respuesta del comité ante el creciente número de solicitudes.

En este sentido, el presente proyecto propone como propósito central el diseño e implementación de una solución tecnológica que asista el proceso de revisión ética, enfocándose particularmente en la verificación de los consentimientos informados. Esta herramienta busca agilizar el procedimiento sin comprometer la rigurosidad ni la calidad de las evaluaciones. Su implementación permitiría reducir los tiempos de revisión, disminuir errores, mejorar la consistencia en los criterios de evaluación y liberar tiempo para que los miembros del comité puedan enfocarse en tareas de mayor complejidad y valor estratégico.

1.3. Propuesta de solución

Para abordar los desafíos asociados con la evaluación de los formatos de consentimiento informado en los proyectos de investigación sometidos al Comité de Ética del Hospital Universitario San Ignacio, se propone el desarrollo de una plataforma web con inteligencia artificial (IA) integrada, que permita agilizar y fortalecer el proceso de revisión ética, sin sustituir el criterio profesional de los evaluadores.

Esta solución, enmarcada en el campo del desarrollo de software y el análisis inteligente de documentos, hará uso de modelos existentes de lenguaje de gran escala (LLMs) combinados con técnicas de prompt engineering para analizar automáticamente los documentos cargados por los usuarios. Su propósito es reducir los tiempos de revisión, disminuir la carga operativa del comité y minimizar errores o inconsistencias derivados del proceso manual, todo ello manteniendo los más altos estándares de calidad y rigor ético.

De forma específica, la plataforma ofrecerá las siguientes funcionalidades clave:

- Detectar evidencia relevante en los consentimientos informados que respalde el cumplimiento de los criterios éticos definidos por el comité.
- Presentar dicha evidencia de forma estructurada, permitiendo que el evaluador, con base en su experiencia y juicio profesional, valide si cada criterio está adecuadamente cubierto en el documento.
- Identificar posibles omisiones, inconsistencias o contradicciones frente a los lineamientos normativos aplicables.
- Generar retroalimentación detallada y específica, orientando a los investigadores sobre los aspectos que deben ser corregidos o mejorados en sus documentos.
- Generar estadísticas con actualización en tiempo real de información como el total de documentos evaluados, tiempo promedio de evaluación, cantidad de documentos aprobados y no aprobados, entre otras.
- Ofrecer a los investigadores un formulario para generar automáticamente documentos de consentimiento informado de una manera asistida, sencilla e intuitiva.

De esta manera, la solución propuesta no solo contribuirá a agilizar el proceso de evaluación, sino que también proporcionará un insumo más preciso y organizado para

la toma de decisiones éticas, potenciando la capacidad del comité para gestionar de forma eficiente el creciente volumen de solicitudes de revisión.

1.4. Justificación de la solución

La implementación de una plataforma con un modelo de inteligencia artificial (IA) integrado para la evaluación ética en el Hospital Universitario San Ignacio, es una solución posible y viable por su capacidad para agilizar y modernizar los procesos de revisión.

Actualmente, los métodos manuales tienen un impacto negativo en este proceso de evaluación, como se explicó con más detalle en las secciones anteriores, por lo que la IA permitiría agilizar una parte fundamental de estos procesos, logrando mayor rapidez, precisión y eficiencia, lo cual es crucial para mantener los estándares éticos y la competitividad a nivel internacional.

De esta manera, los principales argumentos para justificar la propuesta de solución son los siguientes:

- Los criterios de evaluación establecidos para verificar el cumplimiento ético en los documentos de consentimiento informado son sencillos y comprensibles para un modelo de lenguaje de gran tamaño (LLM), siempre que se utilicen prompts bien diseñados.

Lo anterior se puede afirmar, ya que el modelo de IA se encargaría principalmente de buscar coincidencias entre el contenido del consentimiento informado y las normativas establecidas en los documentos de leyes de ética. Esta tarea resulta sencilla para el modelo, ya que se trata de un proceso de detección de similitudes bien definido, donde los criterios éticos están claramente expresados en el texto. Además, existen estudios académicos que respaldan esta afirmación, demostrando que los modelos de lenguaje son capaces de identificar coincidencias y discrepancias en textos legales con una alta precisión, siempre y cuando los datos y los prompts sean adecuadamente formulados.

- La calidad de la revisión no se ve comprometida, ya que el evaluador humano no es reemplazado en el proceso. Este usuario será asistido por el modelo de inteligencia artificial, que recibirá el documento de consentimiento informado y le mostrará cuáles criterios considera que se cumplen, así como la parte específica del documento donde identificó ese cumplimiento. De esta manera, el evaluador siempre tendrá el juicio final sobre el criterio de ética, utilizando la información proporcionada por el modelo como apoyo para agilizar y mejorar la precisión del proceso de revisión.
- No se recurre a una automatización total, ya que esta no es recomendable por razones éticas y de calidad. Un proceso completamente automatizado podría

comprometer la capacidad de evaluar matices y consideraciones contextuales que solo un evaluador humano es capaz de discernir.

Además de esto, la revisión ética requiere un análisis profundo y una toma de decisiones que involucra valores humanos, por lo que delegar el juicio final a un modelo de IA podría dar lugar a errores graves o interpretaciones limitadas de los criterios éticos.

Por otro lado, esta metodología asistida por IA es altamente escalable, permitiendo que el proceso sea replicable en diferentes escenarios o volúmenes de trabajo sin perder calidad, reduciendo significativamente el tiempo necesario para la revisión de documentos, ya que la IA puede identificar rápidamente coincidencias y discrepancias en el texto, y eliminando la necesidad de que los evaluadores humanos revisen cada detalle.

De esta forma, se logra una agilización del proceso, manteniendo la calidad y la integridad de las decisiones éticas, al mismo tiempo que se aprovecha el poder de la IA para acelerar las tareas más mecánicas y repetitivas.

- Es económicamente viable, dado que puede implementarse utilizando herramientas de software libre, como Llama, o soluciones comerciales con esquemas de licenciamiento flexibles, como ChatGPT. El uso de software libre permite reducir significativamente los costos asociados al desarrollo, implementación y mantenimiento, ya que no requiere inversiones iniciales elevadas en licencias de software propietarias.

Adicionalmente, Llama y otras alternativas open-source ofrecen un alto grado de personalización, permitiendo ajustar el modelo a las necesidades específicas del proceso de evaluación ética sin incurrir en costos adicionales.

Por otro lado, soluciones comerciales como ChatGPT ofrecen opciones de licenciamiento flexibles, que permiten pagar solo por el uso necesario, lo cual es ideal para instituciones que no tienen una demanda constante, reduciendo así el costo total. Estas soluciones no solo son accesibles desde una perspectiva económica, sino que también ofrecen escalabilidad y soporte técnico, lo que garantiza que la implementación sea sostenible a largo plazo. De esta forma, se logra un equilibrio entre viabilidad económica y eficiencia tecnológica, haciendo que el sistema sea factible.

2. Descripción del proyecto

2.1. Objetivo general

Desarrollar una plataforma que apoye la gestión eficiente de los procesos de evaluación del comité de ética del Hospital Universitario San Ignacio.

2.2. Objetivos específicos

- Centralizar la gestión del proceso de evaluación ética de formatos de consentimiento informado de los proyectos de investigación.
- Construir una herramienta que, usando inteligencia artificial, determine el cumplimiento o no cumplimiento de requisitos éticos en formatos de consentimiento informado usados en proyectos de investigación presentados al Comité de Ética del Hospital Universitario San Ignacio, de manera que sea capaz de reducir los tiempos y errores de evaluación.
- Integrar el modelo de Inteligencia Artificial a la plataforma desarrollada, de manera que pueda ser usado por los miembros del comité de ética y de manera intuitiva y sencilla.
- Validar el correcto funcionamiento de la plataforma con el modelo de IA integrado, mediante pruebas enfocadas tanto en la operatividad del modelo como en su integración fluida con la plataforma, garantizando que el sistema funcione correctamente y sea útil para los evaluadores.

2.3. Entregables, Estándares y Justificación

Entregable	Estándares asociados	Justificación
Mockups aprobados por el cliente	<ul style="list-style-type: none">● ISO/IEC 25010 (Calidad del producto)● ISO 9241-210 (Diseño centrado en el usuario)● IEEE 830-1998 (Especificación de Requisitos de Software)	La justificación para aplicar los estándares ISO/IEC 25010, ISO 9241-210 e IEEE 830-1998 en los mockups aprobados por el cliente se centra en garantizar una alta calidad, usabilidad y alineación con los requisitos del cliente. ISO/IEC 25010 asegura la funcionalidad y fiabilidad del diseño, mientras que ISO 9241-210 se enfoca en mejorar la experiencia del usuario. IEEE 830-1998 garantiza que los mockups cumplan con los requisitos especificados, facilitando la aprobación del cliente. Estos estándares aseguran un diseño completo y adecuado tanto para el cliente como para los usuarios finales.

Conjunto de prompts utilizados para probar el modelo.	<ul style="list-style-type: none"> • ISO/IEC 25010 (Calidad del Producto de Software) • ISO/IEC 2382 (Vocabulario de Datos) • ISO/IEC TR 24027 (Evaluación de rendimiento de sistemas de IA) 	La aplicación de los estándares ISO/IEC 25010, ISO/IEC 2382 e ISO/IEC TR 24027 en el conjunto de prompts asegura que las pruebas del modelo sean de alta calidad, con términos claros y resultados precisos. ISO/IEC 25010 garantiza que los prompts sean funcionales y fiables, ISO/IEC 2382 asegura la coherencia en el uso de términos, y ISO/IEC TR 24027 permite medir el rendimiento del modelo con objetividad. Esto asegura que los prompts sean adecuados para probar el modelo de manera efectiva.
Informe de evaluación con los resultados obtenidos	<ul style="list-style-type: none"> • ISO/IEC 29119 (Pruebas de Software) • ISO/IEC 25012 (Calidad de los datos) • ISO/IEC 15939 (Medición de Software y Sistemas) 	La aplicación de los estándares ISO/IEC 29119, ISO/IEC 25012 e ISO/IEC 15939 en el "Informe de evaluación con los resultados obtenidos" asegura que el informe sea preciso, claro y medible. ISO/IEC 29119 estructura el proceso de pruebas, ISO/IEC 25012 garantiza la calidad de los datos, y ISO/IEC 15939 asegura que los resultados sean medibles y objetivos. Esto garantiza un informe confiable y bien fundamentado.
Plataforma operativa con el modelo LLM integrado	<ul style="list-style-type: none"> • ISO/IEC 25010 (Calidad del Producto de Software) • ISO/IEC 27001 (Seguridad de la Información) • ISO/IEC 2382 (Vocabulario de Tecnologías de la Información) • ISO/IEC 12207 (Procesos del ciclo de vida del software) 	La aplicación de los estándares ISO/IEC 25010, ISO/IEC 27001, ISO/IEC 2382 e ISO/IEC 12207 en la "Plataforma operativa con el modelo LLM integrado" garantiza que la plataforma sea de alta calidad, segura y coherente. ISO/IEC 25010 asegura su funcionalidad y rendimiento, mientras que ISO/IEC 27001 protege los datos y la seguridad del sistema. ISO/IEC 2382 garantiza el uso claro de términos y conceptos, y ISO/IEC 12207 asegura una gestión adecuada durante todo el ciclo de vida de la plataforma. Estos estándares garantizan una plataforma robusta y segura.

Manual de usuario	<ul style="list-style-type: none"> • ISO/IEC 26514 (Requisitos para diseñadores y desarrolladores de documentación de usuario) • ISO/IEC 25010 (Calidad del producto de software) • IEEE 1063 (Documentación de usuario de software) 	La aplicación de ISO/IEC 26514 garantiza que el manual contenga toda la información necesaria (instalación, configuración, uso y solución de problemas) de forma clara y estructurada. ISO/IEC 25010 asegura que la documentación cumpla con atributos de calidad (usabilidad, comprensibilidad y mantenibilidad), mientras que IEEE 1063 provee buenas prácticas para la redacción y organización de la documentación de usuario. En conjunto, estos estándares aseguran un manual completo, fácil de entender y alineado con las necesidades de los usuarios finales.
-------------------	---	---

Tabla 1. Entregables, estándares y justificación

2.4. Metodología

El desarrollo de este proyecto se realizará bajo el marco de trabajo ágil Scrum, el cual permite construir soluciones de manera iterativa e incremental, con entregas funcionales en ciclos cortos que facilitan la retroalimentación constante y la mejora continua. Scrum es especialmente útil en proyectos como este, donde se requiere validar avances de manera temprana con los stakeholders y adaptar el producto a sus necesidades reales. El trabajo se organiza en sprints de dos semanas y se estructura en cinco fases principales que guían el proceso completo, desde la conceptualización hasta el cierre formal del proyecto.

Fase 1: Levantamiento de requerimientos

Esta fase tiene como objetivo alinear a todos los actores involucrados en el proyecto, incluyendo el equipo de desarrollo, el Comité de Ética y el Director de Tesis, en torno a la visión, alcance y funcionalidades clave de la plataforma. Se llevarán a cabo reuniones iniciales, entrevistas con evaluadores, análisis de requerimientos y prototipado rápido para definir historias de usuario que se tendrán en cuenta para el desarrollo de la plataforma en etapas posteriores. También se validarán los primeros mockups con los usuarios y se configurará el entorno de trabajo técnico, incluyendo el repositorio de código. Esta fase concluye con un backlog priorizado que refleja con precisión las expectativas del comité y que servirá de guía para los sprints de desarrollo.

Fase 2: Desarrollo iterativo de funcionalidades principales

Durante esta fase se desarrollará el núcleo funcional de la plataforma mediante varios sprints consecutivos. El enfoque se centrará en construir las funcionalidades básicas que permitan a los investigadores crear documentos de consentimiento informado y a

los evaluadores gestionarlos adecuadamente. Se implementarán módulos como autenticación, gestión de casos, carga y descarga de documentos, y almacenamiento en MinIO entre otros. El backend será desarrollado bajo Clean Architecture con Express y TypeScript, mientras que el frontend se construirá con React y TypeScript utilizando el enfoque de Atomic Design. Al final de cada sprint se realizarán pruebas manuales de desarrollo, demostraciones de avance y sesiones de retrospectiva para identificar mejoras continuas.

Fase 3: Integración del modelo de inteligencia artificial

Una vez que el sistema base esté operativo, se integrará uno o varios proveedores de modelos de inteligencia artificial, en caso de ser requeridos, que permitirá asistir la verificación de los consentimientos informados. En esta fase se seleccionarán los modelos más adecuados, se diseñarán y afinarán los prompts necesarios, y se desarrollará el flujo de análisis entre backend e interfaz. Se conectará el servicio de IA al backend, se establecerá el formato de respuesta esperado y se modificará la interfaz para que los evaluadores puedan visualizar los criterios analizados, el cumplimiento por cada uno y la justificación textual correspondiente. Esta integración se validará con documentos reales proporcionados por el Comité.

Fase 4: Pruebas funcionales y preparación de entrega

En esta etapa se realizarán pruebas para validar el correcto funcionamiento del sistema, su estabilidad y cumplimiento de los requisitos establecidos. Se ejecutarán pruebas de integración (Jest), pruebas end-to-end (Playwright), pruebas de carga (Artillery) y pruebas de aceptación del usuario (UAT) simulando escenarios reales de uso. También se consolidarán los documentos técnicos requeridos: manual de usuario, guía de despliegue, memoria técnica e informe de pruebas. Al final de esta fase, se generará una versión beta del sistema completamente funcional, que será presentada al Comité de Ética y al Director de Tesis para su aprobación formal.

Fase 5: Cierre del proyecto

La fase final tiene como propósito formalizar la entrega del proyecto y documentar las lecciones aprendidas. Se revisarán todos los artefactos desarrollados, se realizará una demostración final del sistema en funcionamiento y se corregirán posibles observaciones menores. Además, se transferirá el conocimiento necesario al Comité de Ética para el uso autónomo de la plataforma, incluyendo el repositorio, documentación técnica y manuales. Finalmente, se preparará la presentación del proyecto y se firmará el acta de cierre, certificando que todos los objetivos han sido cumplidos.

CONTEXTO DEL PROYECTO

En la actualidad, la evaluación ética de proyectos de investigación representa un pilar fundamental para garantizar la protección de los derechos y el bienestar de los participantes. No obstante, comprender los desafíos reales que enfrentan los comités de ética, especialmente en instituciones con una alta carga investigativa como el Hospital Universitario San Ignacio (HUSI), requiere examinar el contexto operativo en el que desarrollan su labor. A continuación, se presenta un análisis del entorno en el que surge este proyecto, con el fin de evidenciar las limitaciones actuales del proceso de revisión de consentimientos informados y justificar la necesidad de implementar soluciones tecnológicas que optimicen dicha labor sin comprometer su rigurosidad.

1. Marco teórico

La ética en la investigación científica constituye un pilar fundamental para proteger la dignidad, los derechos y el bienestar de los participantes. Su aplicación se realiza a través de mecanismos como los Comités de Ética en Investigación (CEI), encargados de revisar y aprobar protocolos antes de su ejecución. En instituciones de alto nivel como el Hospital Universitario San Ignacio (HUSI), esta función cobra especial relevancia debido al volumen y la complejidad de los estudios recibidos anualmente. Sin embargo, la gestión de estos procesos éticos se ha visto limitada por la falta de herramientas tecnológicas especializadas que permitan optimizar la verificación de documentos clave, como los consentimientos informados.

Actualmente, muchos CEI dependen de procedimientos manuales para evaluar estos documentos, utilizando matrices en papel, formatos de validación físicos y revisiones descentralizadas. Esto genera cuellos de botella, demoras en la aprobación de proyectos y una alta carga operativa para los evaluadores. Adicionalmente, la variabilidad en la redacción de los consentimientos y la subjetividad en la interpretación de los criterios éticos aumenta la probabilidad de errores o inconsistencias.

En respuesta a estas limitaciones, este proyecto propone el diseño e implementación de una plataforma web que digitaliza y asiste el proceso de revisión de consentimientos informados, implementando técnicas de *prompt engineering* que permiten que un modelo de lenguaje basado en inteligencia artificial (IA) detecte omisiones o incumplimientos éticos. Esta solución busca no solo mejorar la eficiencia operativa del comité, sino también fortalecer la transparencia, la trazabilidad y la calidad de las evaluaciones.

Para comprender adecuadamente el alcance y potencial de esta solución, es importante tener en cuenta los siguientes conceptos clave:

- *Consentimiento Informado*: Documento legal que formaliza la participación voluntaria del sujeto en un estudio, y que debe contener información clara sobre objetivos, riesgos, beneficios, derechos y condiciones de confidencialidad. Su correcta

elaboración y evaluación es un requisito obligatorio para cualquier investigación ética.

- *Comités de Ética en Investigación (CEI)*: Organismos encargados de evaluar los aspectos éticos de los estudios. En el caso del HUSI, revisan más de 200 proyectos al año, siendo uno de los puntos críticos la verificación detallada de los consentimientos informados.
- *Inteligencia Artificial aplicada a Procesos Éticos*: El uso de modelos de lenguaje (LLMs) y técnicas de procesamiento de lenguaje natural (NLP) para analizar, interpretar y validar documentos textuales, como los consentimientos. Esta tecnología permite evaluar de forma automatizada si un documento cumple con las normativas éticas establecidas.
- *Prompt Engineering*: Técnica que consiste en diseñar instrucciones específicas para guiar el comportamiento de un modelo de IA sin necesidad de entrenarlo desde cero. Esta estrategia permite adaptar modelos existentes a contextos institucionales, optimizando tiempo, recursos y precisión en las tareas de verificación.
- *Automatización de Procesos Éticos*: Aplicación de reglas predefinidas y flujos digitales para reducir la intervención manual en la revisión de criterios normativos, el llenado de formatos y la validación de requisitos éticos, aumentando la eficiencia, la estandarización y la capacidad de respuesta de los comités.

Con base en estos fundamentos, la solución tecnológica planteada se perfila como una herramienta innovadora para modernizar la gestión ética en investigación, alineada con buenas prácticas internacionales y con el potencial de ser replicada en otras instituciones académicas y hospitalarias del país.

2. Análisis del contexto

A nivel global, varias instituciones han comenzado a incorporar herramientas tecnológicas para apoyar los procesos de evaluación ética. Por ejemplo, la Administración de Alimentos y Medicamentos de los Estados Unidos (FDA) y la Agencia Europea de Medicamentos (EMA) han iniciado la implementación de sistemas basados en inteligencia artificial para agilizar la revisión de documentación regulatoria y consentimientos informados. Estas soluciones, aunque pioneras, están enfocadas en entornos normativos específicos y requieren infraestructuras tecnológicas avanzadas, lo cual dificulta su adopción directa en contextos hospitalarios latinoamericanos con recursos limitados.

Adicionalmente, plataformas como IRBNet o eProtocol ofrecen sistemas completos para la gestión de protocolos de investigación. Aunque integran funciones para el seguimiento de proyectos, presentación de documentos y aprobaciones, su uso implica altos costos de licenciamiento, una curva de aprendizaje elevada y una escasa adaptación a marcos regulatorios locales como la Ley 1581 de 2012 en Colombia. En la mayoría de los casos, no incluyen módulos especializados para la validación automatizada de criterios éticos en los consentimientos informados ni emplean modelos de lenguaje ajustados a la realidad operativa de instituciones como el HUSI.

Si bien las soluciones internacionales representan avances importantes en la automatización de procesos éticos, muchas de ellas no se ajustan a las realidades operativas, presupuestales y normativas del entorno colombiano. Por esta razón, se analizaron otras alternativas viables que podrían adaptarse mejor a las necesidades del Comité de Ética del HUSI, considerando su impacto técnico, organizacional y ético. A continuación, se describen estas opciones.

Alternativas de Solución Consideradas

1. Vectorización de documentos de consentimiento informado

Esta alternativa consiste en convertir el texto de los documentos en vectores numéricos mediante técnicas de procesamiento de lenguaje natural (NLP), utilizando modelos como Word2Vec, GloVe o Doc2Vec. Esta representación permite comparar semánticamente los textos frente a criterios éticos predefinidos.

- *Impacto técnico:* Mejora la precisión al identificar similitudes o discrepancias, pero requiere expertos en NLP y personalización compleja.
- *Impacto organizacional:* Puede implementarse sin grandes inversiones, pero su escalabilidad depende de las capacidades internas de la institución.

2. Sistema basado en reglas y expresiones regulares

Este enfoque emplea patrones predefinidos para identificar si ciertos términos o secciones obligatorias están presentes en los consentimientos informados.

- *Impacto técnico:* Es rápido de implementar, pero rígido y propenso a fallar ante variaciones en el lenguaje.
- *Impacto organizacional:* Bajo costo y facilidad de integración, aunque con mejoras limitadas en eficiencia y profundidad de análisis.

3. Plataformas comerciales de análisis documental

Herramientas como Google Cloud Document AI, Amazon Textract o IBM Watson Discovery permiten procesar documentos y extraer información clave mediante IA.

- *Impacto técnico:* Elevada precisión y velocidad, pero los costos son altos y el tratamiento de datos sensibles en la nube presenta riesgos legales y éticos.
- *Impacto organizacional:* Disminuyen significativamente los tiempos de revisión, pero comprometen la soberanía de los datos.

4. Entrenamiento de un modelo de IA con datos locales

Esta opción plantea entrenar un modelo desde cero usando documentos evaluados previamente por el Comité. El modelo aprendería a reconocer patrones asociados al cumplimiento ético.

- *Impacto técnico:* Alto potencial de precisión, pero requiere gran cantidad de datos etiquetados, infraestructura especializada (GPUs) y un proceso delicado de anonimización y manejo de datos sensibles.
- *Impacto organizacional:* Implica una inversión significativa y un proceso de desarrollo más largo, con riesgos éticos y técnicos asociados.

Frente a estas alternativas, nuestra solución presentada previamente en la sección de “*Propuesta de Solución*”, basada en modelos de lenguaje grandes (LLMs) ajustados mediante ingeniería de prompts representa el mejor balance entre eficiencia, precisión, costo, facilidad de implementación y respeto por los marcos normativos locales.

Esta solución:

- *Evita el entrenamiento desde cero*, utilizando modelos pre entrenados que pueden ser adaptados fácilmente mediante instrucciones bien diseñadas (prompts).
- *No requiere una infraestructura computacional costosa*, ya que puede implementarse con herramientas comerciales asequibles como ChatGPT o gratuitas como LLaMA.
- *Preserva la privacidad de los datos*, al permitir su despliegue en entornos controlados por la institución.
- *Se adapta a múltiples estilos de redacción*, a diferencia de los enfoques basados en reglas fijas o estructuras rígidas.
- *Mantiene al evaluador humano como figura central del proceso*, sirviendo como asistente inteligente para facilitar la verificación inicial sin sustituir el juicio profesional.

Además, su arquitectura modular, basada en tecnologías abiertas como React, Node.js y MongoDB, permite su futura adaptación y escalabilidad en otras instituciones, fortaleciendo la investigación ética en el contexto colombiano. Esta solución es una propuesta alineada con los estándares internacionales y adaptada a las necesidades locales del HUSI, con un valor diferencial claro frente a las alternativas disponibles.

ANÁLISIS DEL PROBLEMA

En este capítulo se presenta un resumen estructurado de los requerimientos del sistema, tanto funcionales como no funcionales, junto con las principales restricciones de la aplicación. Asimismo, se describen las funcionalidades clave del producto, enfocado en la plataforma de gestión de documentos dirigida al Comité de Ética del Hospital Universitario San Ignacio (HUSI).

1. Requerimientos

El sistema ha sido diseñado para brindar apoyo al Comité de Ética en la revisión de consentimientos informados, mediante el uso de herramientas digitales que permiten automatizar partes del proceso sin sustituir el juicio humano. A continuación, se presentan los requerimientos funcionales y no funcionales más relevantes del proyecto, los cuales han sido extraídos del documento de Especificación de Requerimientos (SRS) (*Anexo 10*). Para consultar el detalle completo de cada requerimiento, se recomienda revisar dicho documento.

1.1. Requerimientos no funcionales

Interfaces con el usuario

En este proyecto se ha desarrollado una interfaz única basada en una aplicación web, que actúa como el principal punto de interacción entre el sistema y sus usuarios finales: investigadores y evaluadores. A través de esta plataforma, los usuarios pueden ingresar información, consultar datos, cargar documentos y ejecutar acciones fundamentales como iniciar evaluaciones éticas o enviar los resultados correspondientes.

La interfaz ha sido diseñada siguiendo principios de usabilidad y accesibilidad empleando un enfoque modular inspirado en el modelo Atomic Design. Previamente, se han definido mockups que especifican la estructura visual y la distribución de los componentes, con el objetivo de optimizar la experiencia general del usuario.

A continuación, se presentan los principales requerimientos no funcionales relacionados con la interfaz del usuario, indicando claramente su descripción y su justificación:

- ***RNF01 - Interfaz gráfica accesible vía navegador web (Usabilidad (U))***

- Descripción:

El sistema debe disponer de una interfaz gráfica clara, intuitiva y accesible desde cualquier navegador web moderno, permitiendo que los usuarios autenticados (investigadores y evaluadores) interactúen fácilmente con los distintos módulos del sistema.

- Debe soportar tareas esenciales de manera intuitiva, tales como ingresar *información de casos*, *cargar documentos* y *realizar evaluaciones* mediante modelos de inteligencia artificial.

- Justificación:

Este requerimiento es crucial para proporcionar una experiencia de usuario satisfactoria, reduciendo el tiempo requerido para aprender el uso del sistema y minimizando errores operativos cotidianos. Además, facilita un acceso universal al sistema, independientemente del dispositivo o entorno tecnológico empleado.

- ***RNF02 - Compatibilidad con navegadores modernos (Portabilidad (S - Soporte))***

- Descripción:

La plataforma deberá garantizar compatibilidad total con los navegadores más comunes y ampliamente utilizados en la actualidad, tales como:

- *Google Chrome, Microsoft Edge y Safari*, en sus versiones más recientes.

Se deberá asegurar que todas las funcionalidades visuales y operativas sean consistentes y estables en estos entornos.

- Justificación:

Este requerimiento garantiza la accesibilidad y eficiencia operativa del sistema desde diferentes entornos tecnológicos frecuentes en instituciones académicas y de investigación, evitando incompatibilidades técnicas que puedan afectar negativamente la experiencia de uso.

- ***RNF03 - Adaptabilidad a diversas resoluciones de pantalla (Usabilidad (U))***

- Descripción:

La interfaz deberá adaptarse dinámicamente a diversas resoluciones de pantalla, incluyendo laptops, monitores estándar de escritorio y otros dispositivos comúnmente utilizados, asegurando en todo momento la claridad, legibilidad y completa funcionalidad del sistema.

- Justificación:

Esto es esencial para garantizar una experiencia homogénea y efectiva para los usuarios, permitiendo que investigadores y evaluadores puedan trabajar cómodamente desde diversos dispositivos y contextos laborales, sin enfrentar obstáculos visuales o de operación.

- ***RNF04 - Usabilidad intuitiva y coherente (Usabilidad (U))***

- Descripción:

La interfaz debe diseñarse siguiendo estrictamente las mejores prácticas del diseño UX/UI, incluyendo la agrupación lógica y clara de elementos, el uso consistente y comprensible de iconografía visual, así como una retroalimentación visual precisa frente a las acciones realizadas por el usuario.

- Justificación:

Este requerimiento es fundamental para facilitar una rápida familiarización con el sistema por parte de los usuarios, reducir errores operativos y mejorar significativamente la productividad y la satisfacción general durante su utilización diaria.

Interfaces con el hardware

Los siguientes requerimientos no funcionales aseguran que la comunicación, operatividad y despliegue de la plataforma se integren adecuadamente con los componentes físicos y las infraestructuras tecnológicas estándar, tanto en entornos de desarrollo como de producción. Se prioriza la simplicidad en la conectividad, la compatibilidad con hardware accesible y la eficiencia operativa sin necesidad de equipos especializados.

- ***RNF05 – Comunicación vía protocolo TCP/IP y puerto HTTP estándar (80) (Fiabilidad (R) / Soporte (S))***
 - Descripción:
El sistema debe comunicarse utilizando el protocolo TCP/IP a través del puerto HTTP 80, permitiendo su acceso mediante navegador web sin necesidad de configuraciones de red avanzadas. Esta arquitectura facilita la conexión entre frontend, backend y servicios externos.
 - Justificación:
El uso de un puerto estándar y soportado garantiza una implementación sin problemas técnicos en redes institucionales comunes. Además, contribuye a la interoperabilidad del sistema en diferentes entornos de trabajo sin requerir permisos especiales ni configuraciones complejas.
- ***RNF06 – Infraestructura basada en servicios externos (APIs LLM) (Soporte (S) / Funcionalidad (F))***
 - Descripción:
La plataforma se apoya en APIs externas que alojan modelos de lenguaje de gran escala (LLMs), como Groq o Gemini, lo que evita la necesidad de contar con servidores propios o hardware con GPU dedicado para procesamiento de lenguaje natural.
 - Justificación:
Al utilizar servicios externos, se asegura una alta capacidad computacional sin elevar los costos de infraestructura física. Esto permite un despliegue ágil y escalable, adaptable a futuras actualizaciones del modelo sin dependencia de equipos especializados.
- ***RNF07 – Ejecución en dispositivos personales de desarrollo (Soporte (S))***
 - Descripción:
La plataforma debe poder ser ejecutada de forma completa en los dispositivos personales del equipo de desarrollo, sin necesidad de hardware adicional o recursos especiales.
 - Justificación:
Este requerimiento garantiza que los procesos de desarrollo, prueba y mantenimiento se realicen en entornos accesibles y prácticos, lo que

reduce barreras técnicas y permite una evolución continua del sistema a lo largo del tiempo.

Interfaces con el software

El funcionamiento eficiente de la plataforma depende en gran medida de su correcta integración con componentes de software clave, incluyendo entornos de ejecución, bases de datos, servicios externos de inteligencia artificial y tecnologías de desarrollo modernas. A continuación, se presentan los principales requerimientos no funcionales relacionados con las interfaces de software, todos ellos validados durante el desarrollo del sistema:

- ***RNF08 – Independencia del sistema operativo mediante contenedores Docker (Portabilidad (S) / Fiabilidad ®)***
 - Descripción:
La plataforma debe ser desplegada utilizando contenedores Docker, lo que garantiza su funcionamiento en cualquier sistema operativo compatible con esta tecnología (Linux, Windows o macOS), sin necesidad de ajustes adicionales.
 - Justificación:
El uso de Docker facilita la portabilidad del sistema, permite mantener entornos de desarrollo y producción coherentes y simplifica tareas como el despliegue, la actualización y la escalabilidad de los servicios.
- ***RNF09 – Integración con base de datos MongoDB (Funcionalidad (F) / Rendimiento (P))***
 - Descripción:
La aplicación debe conectarse de forma nativa con MongoDB, una base de datos NoSQL orientada a documentos, ideal para manejar la estructura flexible de los consentimientos informados y sus evaluaciones asociadas.
 - Justificación:
MongoDB proporciona un almacenamiento eficiente de datos semi-estructurados, permite realizar consultas ágiles y escalables, y se adapta fácilmente a la evolución del esquema de datos sin requerir grandes refactorizaciones.
- ***RNF10 – Uso de tecnologías estándar (Express.js, React, TypeScript) (Soporte (S) / Mantenibilidad (S))***
 - Descripción:
El backend de la plataforma debe estar construido con Express.js, mientras que el frontend debe desarrollarse con React y TypeScript, siguiendo principios de arquitectura limpia y diseño modular.
 - Justificación:
Estas tecnologías modernas aseguran la mantenibilidad del sistema, optimizan el rendimiento y permiten una experiencia de desarrollo

consistente. Además, su amplio respaldo en la comunidad facilita el soporte y la incorporación de futuras mejoras.

- ***RNF11 – Integración con modelos de lenguaje mediante APIs externas (Gemini, Groq) (Funcionalidad (F) / Rendimiento (P))***

- Descripción:

La plataforma debe consumir APIs externas de proveedores como Groq o Gemini para realizar evaluaciones automáticas de documentos mediante modelos LLM (Large Language Models), sin requerir infraestructura local de cómputo intensivo.

- Justificación:

Este enfoque maximiza la eficiencia al aprovechar modelos avanzados alojados en la nube, reduce los costos asociados al hardware especializado y permite una escalabilidad flexible sin comprometer la calidad de las evaluaciones automáticas.

Interfaces de comunicaciones

Esta sección contempla los requerimientos no funcionales que garantizan la interoperabilidad, seguridad y robustez de las comunicaciones tanto internas como externas de la plataforma. La solución se apoya en APIs de modelos de lenguaje externos, así como en comunicaciones constantes entre frontend, backend, base de datos y servicios de almacenamiento. Estos requerimientos aseguran una transmisión segura, estructurada y estandarizada de los datos, crítica para el contexto ético en el que se aplica el sistema.

- ***RNF12 – Comunicaciones seguras vía HTTPS (Seguridad (S))***

- Descripción:

Todas las comunicaciones con APIs externas (como Groq o Gemini) deben realizarse mediante el protocolo HTTPS.

- Justificación:

Esta medida garantiza la integridad y confidencialidad de los datos durante la transmisión, cumpliendo estándares de seguridad recomendados para aplicaciones que manejan información sensible y reduciendo el riesgo de ataques de intermediarios.

- ***RNF13 – Formato estándar JSON para intercambio de datos (Interoperabilidad (P))***

- Descripción:

Todos los datos intercambiados entre la plataforma y las APIs externas deben utilizar el formato JSON como estructura estándar.

- Justificación:

JSON es un formato ampliamente adoptado que facilita la serialización y deserialización de datos, mejora la interoperabilidad con otras herramientas y servicios, y simplifica la integración con modelos de lenguaje de terceros.

- ***RNF14 – Autenticación mediante Tokens JWT (Seguridad (S))***

- Descripción:

- Todas las interacciones internas de la plataforma deben estar autenticadas mediante tokens JWT, asegurando que únicamente usuarios con credenciales válidas puedan acceder a funcionalidades protegidas.

- Justificación:

- Este mecanismo ofrece una forma segura, escalable y estandarizada de verificar sesiones activas y prevenir accesos no autorizados, protegiendo los datos del sistema.

- ***RNF15 – Control de acceso basado en roles y sesiones (Seguridad (S))***

- Descripción:

- La plataforma debe aplicar validaciones de acceso basadas en sesiones y roles de usuario, restringiendo el uso de funcionalidades según los permisos asignados. Aunque actualmente no se han implementado firewalls, se recomienda su incorporación en ambientes de producción.

- Justificación:

- El control de acceso por roles permite gestionar el uso responsable del sistema, reforzando su seguridad lógica y previniendo manipulaciones indebidas o accidentales de la información sensible.

1.2. Requerimientos funcionales

El sistema desarrollado contempla una amplia gama de requerimientos funcionales, orientados a satisfacer las necesidades operativas tanto del Comité de Ética como de los investigadores. Esta sección agrupa dichos requerimientos, organizándolos en función de los módulos y funcionalidades clave del sistema, donde se reflejan los flujos principales de interacción entre los usuarios y la plataforma.

Los requerimientos aquí descritos se enfocan especialmente en los dos roles principales: el evaluador (miembro del comité) y el investigador. Para una descripción detallada y exhaustiva de todos los requerimientos, se recomienda consultar el **Anexo 13** (Documento de requerimientos) y el **Anexo 10** (documento de Especificación de Requerimientos de Software (SRS)).

A continuación, se presentan los requerimientos funcionales más importantes correspondientes a la plataforma:

Funcionalidad de Autenticación y registro

La funcionalidad de autenticación y registro constituye uno de los pilares fundamentales del sistema, ya que permite gestionar de manera segura el acceso de los diferentes tipos de usuarios. Esta sección agrupa los requerimientos orientados a garantizar que tanto investigadores como evaluadores puedan ingresar al sistema, crear cuentas, validar credenciales y administrar sus contraseñas de forma segura y

autónoma. Los mecanismos implementados aseguran la protección de la información sensible y establecen una base para el control de acceso a las funcionalidades críticas de la plataforma. A continuación, se describen los requerimientos funcionales más relevantes asociados a este módulo.

- ***RF01 – Registro de investigadores***

- Descripción:

El sistema debe permitir a los nuevos investigadores registrarse proporcionando su nombre, correo institucional y una contraseña segura. La plataforma debe validar que el correo no esté duplicado y que la contraseña cumpla con criterios de robustez.

- Justificación:

Este requerimiento habilita el ingreso autónomo de investigadores al sistema, permitiendo su participación activa sin intervención del comité técnico.

- ***RF02 – Inicio de sesión para investigadores***

- Descripción:

Los investigadores deben poder autenticarse con su correo institucional y contraseña para acceder al sistema.

- Justificación:

Garantiza que solo usuarios autorizados accedan al sistema, asegurando confidencialidad y control de acceso.

- ***RF03 – Usuario evaluador inicial***

- Descripción:

El sistema debe generar automáticamente un usuario evaluador raíz al momento de la primera ejecución. No se permite crear otros evaluadores desde la interfaz.

- Justificación:

Este requerimiento garantiza un acceso controlado al sistema desde su fase inicial y reduce riesgos de manipulación no autorizada.

- ***RF04 – Inicio de sesión para el evaluador***

- Descripción:

El evaluador debe autenticarse mediante correo electrónico y contraseña definidos previamente.

- Justificación:

Permite que el evaluador acceda de forma segura al módulo de revisión, asegurando trazabilidad de sus acciones.

- ***RF05 – Cambio de contraseña (investigador o evaluador)***

- Descripción:

La plataforma debe permitir a cualquier usuario autenticado modificar su contraseña previa validación de la actual.

- Justificación:

Refuerza la seguridad de la cuenta del usuario, evitando accesos indebidos y promoviendo buenas prácticas de gestión de credenciales.

- **RF06 - *Encriptar contraseña (Investigador o evaluador)***

- Descripción:

La plataforma debe aplicar un proceso de hashing seguro a la contraseña del usuario antes de almacenarla en la base de datos.

- Justificación:

Esta medida es fundamental para proteger la información sensible de los usuarios y garantizar la seguridad de sus credenciales, cumpliendo con buenas prácticas en el manejo de datos personales.

Funcionalidad de Gestión de archivos (Evaluador)

Uno de los componentes fundamentales del sistema es la gestión eficiente de los documentos de consentimiento informado, para el rol del Evaluador. Esta funcionalidad permite a los evaluadores cargar archivos PDF, almacenarlos de manera segura, consultar su historial y obtener los resultados de la evaluación automática generada por el modelo de lenguaje. A continuación, se presentan los principales requerimientos funcionales relacionados con esta funcionalidad.

- **RF07 – *Carga de archivos PDF***

- Descripción:

El sistema debe permitir a los evaluadores subir archivos PDF de consentimientos informados para ser evaluados éticamente.

- Justificación:

Esta acción representa el punto de entrada del documento al sistema y es el paso inicial para activar todo el flujo de evaluación y trazabilidad.

- **RF08 – *Almacenamiento seguro en MinIO***

- Descripción:

Los archivos cargados deben almacenarse de forma estructurada y segura en el sistema de almacenamiento MinIO, compatible con el estándar S3.

- Justificación:

Garantiza la disponibilidad, persistencia y organización de los documentos, permitiendo su recuperación eficiente y segura. Adicionalmente es gratuito y se puede desplegar en el entorno del HUSI.

- **RF09 – *Listado de documentos cargados***

- Descripción:

El sistema debe mostrar una tabla con todos los documentos previamente cargados, incluyendo su estado, fecha de carga y acciones disponibles como visualización o descarga.

- Justificación:

Facilita la trazabilidad y gestión de los documentos por parte de los evaluadores, ofreciendo visibilidad sobre el historial de actividades realizadas.

- ***RF10 – Descarga de resultados de evaluación IA***

- Descripción:

El sistema debe permitir a los evaluadores descargar un archivo PDF con los resultados generados por el modelo de lenguaje, que contenga los criterios evaluados y su estado (Aprueba/No aprueba/No aplica).

- Justificación:

Esta funcionalidad permite compartir los resultados fuera del sistema, mantener un respaldo documental y, en consecuencia, facilitar la toma de decisiones por parte del comité o del investigador, ya que dichos resultados pueden ser enviados directamente a los destinatarios correspondientes.

Funcionalidad de Interacción con el modelo de IA

Esta funcionalidad representa el núcleo innovador del sistema, ya que permite interactuar con modelos de lenguaje de gran escala (LLMs) para apoyar la evaluación automatizada de los consentimientos informados. A través de esta interacción, se busca reducir la carga operativa del Comité de Ética y proporcionar retroalimentación estructurada y verificable para cada documento analizado. La plataforma no reemplaza el juicio humano, pero proporciona información preliminar valiosa para apoyar la toma de decisiones. A continuación, se presentan los requerimientos funcionales más relevantes para esta sección:

- ***RF011 – Envío de documento al modelo de IA***

- Descripción:

El sistema debe permitir enviar el documento PDF cargado al modelo de inteligencia artificial externo para su evaluación ética.

- Justificación:

Este paso es fundamental, ya que constituye la interacción con el modelo de IA. Sin él, no sería posible realizar evaluaciones asistidas, lo cual representa el principal valor agregado de la plataforma.

- ***RF012 – Visualización de resultados automáticos***

- Descripción:

La plataforma debe mostrar de manera clara y estructurada si el documento evaluado cumple o no con los criterios éticos definidos.

- Justificación:
Presentar esta información de forma inmediata y comprensible agiliza significativamente el proceso de revisión, reduciendo el tiempo que el evaluador necesita para analizar cada documento de forma manual.
- ***RF013 – Identificación de evidencias dentro del documento***
 - Descripción:
El sistema debe señalar los fragmentos específicos del consentimiento que respaldan el cumplimiento o incumplimiento de los criterios, según lo determinado por la IA.
 - Justificación:
Facilita la trazabilidad del análisis automático y brinda transparencia al evaluador para validar las conclusiones del modelo.
- ***RF014 – Edición de la retroalimentación generada por IA***
 - Descripción:
La plataforma debe permitir al evaluador editar la retroalimentación generada por la IA antes de enviarla al investigador o al comité.
 - Justificación:
Esto garantiza que el criterio humano siga siendo el eje central del proceso de evaluación, permitiendo al evaluador ajustar los resultados o aclarar observaciones en caso de que el modelo de inteligencia artificial presente imprecisiones.
- ***RF015 – Comentar retroalimentación generada por IA***
 - Descripción:
La plataforma debe permitir al evaluador agregar observaciones adicionales o notas explicativas complementarias sobre los resultados generados por IA.
 - Justificación:
Esta funcionalidad permite complementar la evaluación con observaciones personalizadas, aclarando posibles dudas que el investigador pueda llegar a tener. Además, resulta útil para registrar elementos relevantes que el modelo de inteligencia artificial no identifica de forma automática
- ***RF016 - Opción de evaluar documentos***
 - Descripción:
La plataforma debe ofrecer una opción (por ejemplo, un botón "Evaluar") que permita iniciar el proceso de evaluación para cada documento en la lista de forma individual.

Funcionalidad de Gestión de correos

Esta funcionalidad permite establecer un canal de comunicación eficiente y

documentado entre el evaluador, el investigador y el Comité de Ética. A través del sistema, los resultados generados por el modelo de inteligencia artificial pueden ser compartidos por correo electrónico, asegurando la trazabilidad en el proceso y la integridad en la entrega de información crítica para la toma de decisiones.

- ***RF017 – Enviar correo al investigador***

- Descripción:

- El sistema debe permitir que el evaluador envíe, mediante correo electrónico, la retroalimentación generada por la plataforma al investigador correspondiente.

- Justificación:

- Este proceso automatiza la comunicación con el investigador, acelerando la entrega de resultados y facilitando su revisión sin necesidad de intermediarios manuales.

- ***RF018 – Generar automáticamente el contenido del correo***

- Descripción:

- La plataforma debe construir automáticamente el contenido del mensaje, incluyendo la retroalimentación generada por el modelo de IA en un formato legible, claro y completo.

- Justificación:

- Generar el contenido automáticamente garantiza homogeneidad, reduce el margen de error y asegura que toda la información relevante sea comunicada correctamente.

- ***RF019 – Enviar correo al Comité de Ética***

- Descripción:

- El sistema debe permitir al evaluador enviar al Comité de Ética el resultado consolidado de la evaluación, incluyendo si el consentimiento fue aprobado o requiere ajustes, junto con las justificaciones.

- Justificación:

- Esto formaliza la entrega del análisis a los responsables de tomar la decisión final, asegurando un flujo de comunicación transparente y respaldado por el sistema.

- ***RF020 – Reenvío de correo***

- Descripción

- Permitir al evaluador reenviar manualmente el correo electrónico al investigador en caso de error, notificando claramente el resultado de cada intento.

- Justificación

- En caso de que ocurra algún error con el envío del correo, ya sea al comité de ética o al investigador, el evaluador debe ser capaz de poder reenviar nuevamente este correo desde la plataforma

Funcionalidad de Análisis y estadísticas

Esta funcionalidad proporciona al evaluador herramientas visuales y métricas consolidadas para el seguimiento y análisis del proceso de evaluación. A través de gráficos dinámicos e indicadores clave, se facilita la toma de decisiones, el monitoreo del progreso y la identificación de cuellos de botella en el proceso de revisión de consentimientos informados. Los datos se presentan de manera clara y se actualizan en tiempo real conforme se completan nuevas evaluaciones.

- ***RF021 – Visualización de estadísticas generales***

- Descripción:

- La plataforma debe mostrar un resumen numérico con el total de evaluaciones realizadas, su estado (aprobadas, rechazadas) y la tasa de devoluciones.

- Justificación:

- Este resumen ofrece al evaluador una vista general del rendimiento del sistema, permitiendo detectar tendencias y áreas que requieren atención.

- ***RF022 – Estadísticas comparativas con periodos anteriores***

- Descripción:

- El sistema debe mostrar comparativas respecto al periodo anterior (mes pasado) para las métricas generales de evaluación.

- Justificación:

- La comparación temporal facilita el análisis de la evolución del desempeño y permite identificar mejoras o retrocesos en los procesos de evaluación.

- ***RF023 – Tiempo promedio de evaluación***

- Descripción:

- El sistema debe calcular automáticamente y mostrar el tiempo promedio utilizado para completar una evaluación ética.

- Justificación:

- Esta métrica permite medir la eficiencia operativa y sirve como base para implementar acciones de mejora o ajustes en la carga de trabajo del comité.

- ***RF024 – Presentar información estadística mediante gráficos***

- Descripción:

- La información estadística debe ser presentada mediante gráficos e indicadores visualmente intuitivos, que permitan su interpretación rápida.

- Justificación:

- Una buena visualización de datos mejora la comprensión y facilita la toma de decisiones informadas por parte de los evaluadores.

- ***RF025 – Actualización dinámica de estadísticas***

- Descripción:

El sistema debe actualizar automáticamente los gráficos y métricas mostradas cada vez que se registre una nueva evaluación o se modifique una existente.

- Justificación:

Garantiza que el evaluador tenga acceso en tiempo real a información actualizada, lo que fortalece la trazabilidad y mejora la toma de decisiones basadas en datos recientes.

Funcionalidad de Configuración del modelo de IA

Esta funcionalidad permite al evaluador ajustar dinámicamente los criterios utilizados por el modelo de lenguaje para realizar la evaluación ética. El objetivo es garantizar que el comportamiento del modelo se mantenga alineado con los estándares del comité, sin requerir conocimientos técnicos complejos. Las opciones de configuración incluyen la edición de prompts, el control de versiones y la activación o desactivación de criterios, permitiendo una personalización flexible del sistema.

- ***RF026 – Modificación de prompts de evaluación***

- Descripción:

El sistema debe permitir al evaluador modificar y ajustar los prompts utilizados por el modelo de IA, a través de una interfaz accesible que no requiera conocimientos técnicos avanzados.

- Justificación:

Esta capacidad ofrece autonomía al comité para adaptar los criterios evaluativos según cambios normativos, sin necesidad de intervención del equipo técnico.

- ***RF027 – Visualización de prompts***

- Descripción:

El sistema debe mostrar claramente los prompts actuales configurados para cada evaluación, facilitando su consulta y revisión.

- Justificación:

Brinda transparencia y control sobre el funcionamiento del modelo de IA, asegurando que los evaluadores siempre estén al tanto de los criterios aplicados en cada caso.

- ***RF028 - Restablecer prompts***

- Descripción:

El sistema debe permitir devolver el estado de los prompts a su configuración inicial según lo requiera el evaluador.

- Justificación:

El evaluador debe contar con la funcionalidad para restablecer los prompts a su versión inicial sugerida, en caso de que se presenten errores durante la evaluación y sea necesario regresar a la configuración original.

- **RF029 - Prompts efectivos**

- Descripción

La plataforma debe ser capaz de identificar las leyes que no se están cumpliendo, ya sea por falta de información o incumplimiento a la normativa.

- Justificación

Los prompts deben ser lo suficientemente claros y efectivos para que el modelo seleccionado pueda identificar correctamente las leyes que se están cumpliendo y las que no.

Funcionalidad de Gestión del consentimiento (Investigador)

Esta funcionalidad permite al investigador crear, editar y gestionar consentimientos informados mediante un formulario. El sistema guía paso a paso el llenado de los campos requeridos, valida su completitud, permite conservar la información diligenciada, generar documentos en formato PDF y consultar el historial de casos creados. Su propósito principal es facilitar la construcción de documentos éticos completos, normalizados y trazables, alineados con los requisitos establecidos por el comité de ética institucional.

- **RF023 – Crear caso de consentimiento informado**

- Descripción:

El sistema debe permitir al investigador iniciar la creación de un nuevo caso, habilitando el formulario correspondiente para su diligenciamiento estructurado.

- Justificación:

Este paso permite gestionar cada consentimiento como un caso independiente, facilitando su trazabilidad, edición y redacción dentro de un flujo estructurado y controlado.

- **RF024 – Completar formulario de consentimiento**

- Descripción:

El sistema debe mostrar un formulario con los campos requeridos para el consentimiento informado: introducción, info general, riesgos, etc.

- Justificación:

Esta estructura garantiza que el investigador no omita secciones fundamentales y que la documentación final cumpla con los estándares normativos establecidos.

- **RF025 – Mostrar errores de completitud**

- Descripción:

El sistema debe notificar al investigador de manera clara si algún campo obligatorio está incompleto o contiene errores antes de avanzar o guardar.

- Justificación:

Esta validación previene la generación de documentos deficientes, reduce devoluciones por parte del comité y mejora la calidad del proceso de creación.

- **RF026 – Validar campos obligatorios**

- Descripción:

El sistema debe impedir la generación del documento final si no se han diligenciado todos los campos obligatorios establecidos por el comité.

- Justificación:

Evita que se generen documentos incompletos, garantizando el cumplimiento normativo y de completitud antes de ser enviados a evaluación.

- **RF027 – Guardar borrador del consentimiento**

- Descripción:

La plataforma debe permitir guardar el avance del formulario en *localstorage* en cualquier momento sin requerir completitud total.

- Justificación:

Facilita el trabajo en jornadas no continuas y reduce el riesgo de pérdida de información durante la construcción del documento.

- **RF028 – Generar PDF del consentimiento informado**

- Descripción:

El sistema debe permitir generar un archivo PDF con los datos ingresados en el formulario, siguiendo el formato oficial aprobado por el comité de ética.

- Justificación:

Establece un formato uniforme de presentación que mejora la legibilidad, validación y archivo de los consentimientos informados.

- **RF029 – Consultar historial de casos generados**

- Descripción:

El investigador debe poder visualizar todos los casos de consentimiento creados, con sus fechas y la opción de poder descargarlos nuevamente, en caso de ser necesario.

- Justificación:

Garantiza trazabilidad documental, facilita el seguimiento de evaluaciones previas y permite acceder a registros archivados cuando sea necesario.

2. Restricciones

El desarrollo de la plataforma web destinada a apoyar el proceso de evaluación ética del Comité de Ética del Hospital Universitario San Ignacio (HUSI) está sujeto a múltiples restricciones técnicas, tecnológicas, normativas e institucionales. Estas restricciones garantizan tanto la viabilidad del proyecto como su correcta operación, alineándose con las prácticas institucionales y las normativas éticas aplicables. A

continuación, se detallan estas restricciones clasificadas en restricciones de hardware, software, impuestas por el cliente y generales.

2.1. Restricciones de Hardware

Para asegurar un desempeño adecuado del sistema, se establecen las siguientes limitaciones técnicas mínimas a nivel de hardware, tanto para el entorno de desarrollo como para el despliegue en producción:

- *Procesador:*
Mínimo de 4 núcleos para garantizar eficiencia en el procesamiento simultáneo de múltiples contenedores Docker.
- *Memoria RAM:*
Mínimo de 8 GB de RAM debido a los requisitos de ejecución concurrente del frontend, backend, base de datos MongoDB y el sistema de almacenamiento MinIO.
- *Espacio en disco:*
Al menos 10 GB de espacio libre para dependencias, volúmenes de Docker, almacenamiento de datos persistentes y archivos generados durante el uso del sistema.
- *Conectividad de Red:*
Tarjeta de red que permita conexión estable a internet para acceder a APIs externas (Gemini, Groq, LanguageTool) y operar adecuadamente la aplicación web.

2.2. Restricciones de Software

Las siguientes restricciones están relacionadas con las tecnologías y herramientas que serán empleadas para el desarrollo y despliegue de la plataforma, garantizando su compatibilidad, rendimiento y escalabilidad:

- *Contenedores y Virtualización:*
Docker y Docker Compose para despliegue y orquestación de todos los servicios, garantizando portabilidad entre entornos.
- *Sistemas Operativos Compatibles:*
Cualquier sistema operativo compatible con Docker (Linux, Windows con Docker Desktop o macOS con Docker instalado).
- *Base de Datos:*
MongoDB como base de datos NoSQL, seleccionada por su capacidad para gestionar eficientemente documentos JSON complejos y escalabilidad ante crecimiento en volumen de datos.
- *Almacenamiento de Archivos:*
MinIO como sistema auto alojado compatible con el protocolo S3, para almacenamiento robusto, escalable y económico de documentos PDF y evaluaciones generadas.

- *Tecnologías para Desarrollo:*
 - *Backend:* TypeScript con Bun como entorno de ejecución y empaquetador, Express.js para la implementación del servidor y organización basada en Clean Architecture.
 - *Frontend:* React con TypeScript, Vite como herramienta de compilación, patrón Atomic Design y librerías complementarias como shadcn/ui y TanStack Query para gestión eficiente de interfaces.
- *Herramientas CASE:*

LucidChart y Visual Paradigm como herramientas principales para análisis, modelado UML y documentación consistente de los requerimientos y diseños técnicos del sistema.
- *Navegadores Compatibles:*

Las últimas versiones de Google Chrome y Microsoft Edge.

2.3. Restricciones impuestas por el cliente

El cliente establece ciertas restricciones clave para asegurar la alineación de la plataforma con sus objetivos institucionales, éticos y operativos:

- *Confidencialidad y seguridad de los datos:*
 - Las credenciales de acceso a servidores, bases de datos y otros servicios deben mantenerse seguras, privadas y restringidas únicamente al personal autorizado.
- *Uso complementario plataforma fundanet:*
 - La plataforma debe ser diseñada para funcionar como un complemento estratégico de Fundanet. Aunque no es posible una integración directa entre ambos sistemas, ambas plataformas deben mantener una relación funcional que permita optimizar los procesos y maximizar el valor para los usuarios.
- *Idioma del sistema:*
 - Español, sin soporte multilingüe en esta versión inicial, debido al contexto específico, institucional y operativo.
- *Seguimiento continuo:*
 - Obligación de realizar reuniones periódicas con los miembros del Comité de Ética, para revisar avances, validar funcionalidades implementadas e incorporar ajustes necesarios según retroalimentación continua, durante la etapa de desarrollo. Una vez se realice la entrega oficial del desarrollo, este seguimiento no será requerido.

2.4. Restricciones de diseño

La plataforma está sujeta a las siguientes decisiones arquitectónicas y restricciones de diseño, que influyen directamente sobre el desarrollo, mantenimiento y evolución futura del sistema:

- *Arquitectura N-Tier:*
El sistema debe tener separación clara de la lógica en capas: Presentación, lógica de negocio y persistencia.
- *Clean Architecture:*
El backend debe estructurarse en capas independientes, desacoplando la lógica de negocio del framework utilizado y de la capa de persistencia, garantizando alta cohesión, bajo acoplamiento y fácil mantenimiento.
- *Atomic Design (Frontend):*
El diseño visual e interacción de usuario sigue estrictamente principios de Atomic Design, permitiendo interfaces modulares, escalables y mantenibles.
- *Control de versiones:*
Uso exclusivo de GitHub para la gestión, colaboración y versionado del código fuente del proyecto.

2.5. Restricciones generales

Además de las restricciones específicas de hardware, software, diseño y aquellas impuestas por el cliente, existen condiciones generales adicionales que afectan directamente las decisiones operativas y estratégicas de la plataforma web, particularmente en términos de seguridad, operatividad continua, cumplimiento normativo y adaptación institucional. Estas restricciones generales establecen un marco fundamental para asegurar la alineación total del proyecto con los estándares éticos, técnicos y organizacionales requeridos por el Hospital Universitario San Ignacio (HUSI):

- *Ausencia de mecanismos de tolerancia a fallos:*
Actualmente, la plataforma no dispone de redundancia ni reinicio automático en caso de fallos críticos. La interrupción de un servicio esencial (backend, base de datos, almacenamiento) implica que la plataforma deja de estar operativa hasta su restablecimiento manual, lo cual puede afectar la continuidad operativa. Este punto deberá abordarse explícitamente en futuras etapas del proyecto.
- *Consideraciones críticas de seguridad digital:*
Aunque la plataforma implementa controles básicos de seguridad (autenticación JWT y control de acceso por roles), no cuenta actualmente con cifrado HTTPS en entornos de desarrollo. Por lo tanto, para el despliegue en producción, será obligatorio implementar comunicaciones cifradas mediante HTTPS. Además, se recomienda evaluar e implementar otras medidas avanzadas de seguridad digital

(firewalls) en futuras fases para fortalecer la seguridad global del sistema.

- *Cumplimiento estricto con políticas institucionales del HUSI:*

El sistema debe cumplir en todo momento con las políticas internas del Hospital Universitario San Ignacio relacionadas con confidencialidad, gestión segura de datos personales, y las buenas prácticas establecidas para la evaluación ética de proyectos. Cualquier modificación futura en estas políticas institucionales deberá reflejarse inmediatamente en ajustes del sistema, tanto funcionales como operativos.

- *Normativas éticas nacionales e internacionales:*

La plataforma está obligada a cumplir estrictamente con normativas éticas internacionales (Declaración de Helsinki y directrices CIOMS) y nacionales (Ley 1581 de 2012). Las funcionalidades del sistema deberán reflejar este marco normativo, y cualquier desarrollo o ajuste deberá estar alineado con dichos estándares éticos.

- *Flexibilidad limitada en la infraestructura:*

Aunque el sistema está diseñado para ser modular y adaptable, su correcto funcionamiento queda condicionado a una disponibilidad mínima de recursos computacionales especificados claramente en las restricciones de hardware. Desplegar la plataforma por debajo de esos requerimientos mínimos podría afectar su desempeño.

3. Análisis Costo Operacional

La implementación de una plataforma digital inteligente para apoyar el proceso de revisión de consentimientos informados en el Comité de Ética del HUSI implica un costo operativo bajo moderado, completamente justificable frente a los beneficios obtenidos en términos de eficiencia, trazabilidad y reducción de errores. El uso de tecnologías open source como Bun o Node.js, React, MongoDB y MinIO permite eliminar los gastos asociados a licencias comerciales, lo que disminuye significativamente el costo del sistema.

Los costos operativos principales se concentran en tres áreas específicas:

- *Soporte técnico mínimo:* Incluyendo tareas periódicas de monitoreo, respaldo y solución de incidencias menores.
- *Consumo de modelos de lenguaje:* Si se supera el límite de tokens gratuitos de los modelos LLM utilizados (como Gemini o Groq), el modelo podría quedar fuera de servicio, por lo cual sería necesario pagar una suscripción la cual extienda los límites de uso de cada proveedor.

Con todos estos factores en cuenta, el costo operativo de la plataforma se convierte en una inversión estratégica. En comparación con los recursos que actualmente se destinan a revisiones manuales, correcciones y retrabajos, esta solución no solo reduce de manera significativa los tiempos de evaluación, sino que también optimiza el uso del personal del Comité, lo que fortalece su sostenibilidad y capacidad de crecimiento a mediano y largo plazo.

3.1. Costo de Infraestructura

La plataforma fue diseñada con una arquitectura modular y basada en contenedores, lo que permite su despliegue tanto en entornos locales como institucionales sin incurrir en costos elevados desde las fases iniciales. A continuación, se detallan los principales componentes de la infraestructura:

- *Contenedores Docker*: Toda la solución (frontend, backend, base de datos, almacenamiento y servicios auxiliares) se ejecuta en contenedores orquestados mediante Docker Compose. Esta configuración garantiza portabilidad, aislamiento entre servicios y facilidad de despliegue, sin generar costos adicionales, ya que se trata de herramientas de código abierto.
- *Frontend y Backend*: Desarrollados en React con Vite y Express.js con Node, respectivamente, ambos pueden ser desplegados en servidores estándar sin necesidad de licencias comerciales ni hardware especializado.
- *Base de datos y almacenamiento*: Se utilizan MongoDB y MinIO, ambos ejecutados en contenedores propios. MinIO actúa como sistema de almacenamiento de objetos compatible con el protocolo S3, ofreciendo una alternativa completa y escalable sin requerimientos de licenciamiento.
- *Modelo de IA*: Para entornos de prueba o simulación, el sistema puede integrarse con modelos de lenguaje accesibles mediante APIs gratuitas, evitando así la necesidad de infraestructura local avanzada como servidores con GPU.

Esta estrategia permite aprovechar la infraestructura existente de manera eficiente y garantiza que la plataforma pueda operar desde sus primeras etapas sin requerir inversiones adicionales en hardware o servicios en la nube.

3.2. Costo de Mantenimiento

El mantenimiento del sistema se proyecta como una tarea de bajo costo y fácil gestión, que puede ser asumida por el equipo técnico institucional del HUSI. Las actividades contempladas incluyen:

- *Mantenimiento preventivo mensual*: Comprende la revisión periódica de registros de actividad (logs), respaldo de la base de datos, verificación del funcionamiento de los contenedores, actualización de dependencias y control de disponibilidad de los servicios. Estas tareas se ejecutarán únicamente en caso de ser necesarias y bajo solicitud del Comité de Ética.

- Gestión de claves de API y ajustes de prompts: Se recomienda realizar una revisión semestral para garantizar la vigencia de las claves de acceso a los modelos de lenguaje utilizados y ajustar los prompts si el Comité de Ética actualiza sus criterios de evaluación.
- Actualización de paquetes y dependencias: Dado que la plataforma utiliza tecnologías ampliamente soportadas y de código abierto (como React, Node.js, MongoDB, entre otras), no se requiere suscripción a servicios externos ni pago de licencias comerciales para realizar estas actualizaciones.

Estas acciones permiten mantener la plataforma operativa y actualizada con un esfuerzo técnico razonable, alineado con los recursos disponibles. Además, contribuyen a asegurar la continuidad del sistema y a mantener altos estándares de calidad y eficiencia en el proceso de evaluación ética.

4. Especificación Funcional

La plataforma está diseñada para asistir al Comité de Ética del HUSI en la revisión de consentimientos informados mediante un flujo digital que combina formularios, validaciones automáticas y un modelo de inteligencia artificial que analiza el cumplimiento de criterios normativos.

El sistema contempla funcionalidades para la autenticación y gestión de usuarios con control de acceso basado en roles (evaluador, investigador), la carga y descarga de documentos, la consulta del historial de evaluaciones y la generación de respuestas automáticas por parte del modelo.

Adicionalmente, se proporciona una interfaz que permite visualizar los criterios evaluados por la IA junto con la justificación textual del modelo y los fragmentos del documento asociados, promoviendo la transparencia y la trazabilidad de cada revisión. Las funcionalidades están organizadas en módulos independientes bajo una arquitectura N-Tier, lo cual permite su escalabilidad futura y su integración con otros sistemas si así se requiere.

4.1. Actores

El sistema contempla los siguientes actores principales:

- Investigador: Es el usuario que cuenta con la funcionalidad de crear los documentos de consentimiento informado a través de un formulario que le proporciona una guía para desarrollarlo, incluyendo validaciones ortográficas y de completitud.
- Evaluador del Comité de Ética: Es el responsable de subir los archivos

de consentimiento informado a la plataforma, realizar su evaluación asistida a través del modelo de IA, modificar dicha evaluación de acuerdo a su criterio y emitir una aprobación o devolución del documento, por medio de un correo electrónico. Adicionalmente este actor también podrá visualizar estadísticas relacionadas con la información de las evaluaciones realizadas para evidenciar métricas de la plataforma..

4.2. Historias de usuario

A continuación, se presentan las historias de usuario más representativas del sistema, definidas en función de las necesidades del Comité de Ética del HUSI y de los flujos críticos para investigadores y evaluadores. Estas historias fueron redactadas siguiendo el formato: “Como [rol], quiero [acción] para [objetivo]”, y guían el desarrollo iterativo de la solución en el marco de trabajo Scrum.

- ***HU-001 - Registro de Investigador:***
Como Investigador,
Quiero registrarme en la plataforma,
Para poder crear, y visualizar los formularios de consentimiento informado.
- ***HU-002 - Cambio de contraseña***
Como usuario,
Quiero cambiar mi contraseña,
Para poder establecer una nueva contraseña que sea más segura o fácil de recordar para mí.
- ***HU-003 - Inicio de sesión (Evaluador)***
Como evaluador,
Quiero iniciar sesión en la plataforma,
Para acceder a mi cuenta y utilizar los servicios disponibles.
- ***HU-004 - Subida de documentos para evaluación***
Como evaluador del comité de ética,
Quiero cargar un archivo con el formato específico de consentimiento informado en la plataforma,
Para que pueda ser analizado por la IA y verificar si cumple con las normas éticas establecidas.
- ***HU-005 - Generación del consentimiento (Investigador)***
Como investigador,

Quiero diligenciar un formulario estructurado para generar un consentimiento informado,

Para asegurarme de que el documento cumple con el formato exigido por el comité.

- ***HU-006 - Evaluación del contenido del documento***

Como evaluador del comité de ética,

Quiero que la plataforma analice el contenido del documento cargado,

Para verificar si cumple con las normas éticas establecidas.

- ***HU-007 - Identificación de normas cumplidas e incumplidas***

Como evaluador,

Quiero que la plataforma me indique qué normas éticas aprueba o no el documento,

Para agilizar y facilitar mi proceso de revisión y toma de decisiones.

- ***HU-008 - Justificación de cumplimientos***

Como evaluador,

Quiero saber por qué la IA clasificó el cumplimiento de ciertas normas,

Para poder confirmar o corregir el resultado de la evaluación.

- ***HU-009 - Generación de informe de evaluación***

Como evaluador,

Quiero recibir un informe detallado con el análisis del documento,

Para tener un resumen claro de su cumplimiento ético.

- ***HU-010 - Visualización del historial de casos (Investigador)***

Como investigador,

Quiero ver los consentimientos que he creado,

Para tener un historial de los casos, en caso de que necesite consultar uno en específico.

- ***HU-011 - Envío de evaluación por correo (Evaluador)***

Como evaluador,

Quiero enviar el resultado de la evaluación al investigador o al comité,

Para notificar los resultados formales del proceso.

- ***HU-012 - Validación de correo institucional***

Como evaluador,

Quiero que la plataforma valide que el correo destinatario sea institucional,

Para asegurar que solo personal autorizado reciba la información enviada.

- ***HU-013 - Visualización de estadísticas del sistema***

Como evaluador,
Quiero acceder a gráficos y métricas del desempeño de las evaluaciones,
Para monitorear el estado y eficacia del proceso de revisión.

- ***HU-014 - Edición de prompts de evaluación ética***

Como evaluador,
Quiero editar los prompts utilizados por la IA,
Para ajustar los criterios de evaluación a los lineamientos del comité.

- ***HU-015 - Inicio de sesión (Investigador)***

Como Investigador,
Quiero iniciar sesión en la plataforma,
Para acceder a mi cuenta y utilizar los servicios disponibles.

Estas historias guían el desarrollo iterativo del sistema dentro del marco de trabajo Scrum, asegurando que cada funcionalidad implementada responda a una necesidad concreta y prioritaria del Comité de Ética.

DISEÑO DE LA SOLUCIÓN

1. **Arquitectura del Sistema**

La plataforma se fundamenta en un modelo *N-Tier*, que distribuye la aplicación en tres capas diferenciadas: presentación, lógica de negocio y acceso a datos. Esta división modular asegura que cada responsabilidad permanezca encapsulada, facilita el mantenimiento evolutivo y refuerza la escalabilidad del sistema. Asimismo, se adopta el patrón *cliente-servidor*, de modo que los usuarios interactúan con una interfaz web que comunica sus peticiones al servidor backend a través de HTTP, delegando en éste la orquestación de la lógica de la solución.

La capa de presentación se ha implementado con React y TypeScript, siguiendo el enfoque de *Atomic Design* para organizar los componentes en niveles que van desde átomos, hasta organismos y plantillas que combinan múltiples piezas reutilizables. Gracias a esta jerarquía, se logra coherencia visual, escalabilidad, mantenibilidad y uniformidad en la experiencia del usuario, ya sea investigador o evaluador. La comunicación con el backend se efectúa mediante Axios y un esquema RESTful, mientras que la autenticación y autorización se gestionan con JWT. Además, durante la creación de casos, la capa de presentación se integra directamente con LanguageTool, lo que permite realizar correcciones ortográficas en tiempo real y mejorar la calidad de los textos ingresados.

Por su parte, la lógica de negocio reside en un servidor construido con Express.js y TypeScript, organizado de conformidad con los principios de *Clean Architecture*. En el centro del diseño se ubican las entidades de dominio, que representan las reglas esenciales del proyecto. A su alrededor, los casos de uso encapsulan la aplicación de dichas reglas para operaciones concretas, mientras que la capa de adaptadores y controladores expone los puntos de entrada desde peticiones HTTP y traduce las respuestas para la capa de infraestructura. En el anillo más externo se encuentran las dependencias específicas, como el envío de correos, el almacenamiento o la conexión con proveedores de IA, lo que permite reemplazar o actualizar cualquier componente externo sin afectar la lógica central. Este esquema de capas concéntricas facilita las pruebas unitarias y promueve la mantenibilidad a largo plazo.

La capa de acceso a datos combina dos servicios especializados:

- *MongoDB*, como almacén NoSQL, custodia la información estructurada (usuarios, evaluaciones y prompts).
- *MinIO*, compatible con el protocolo S3, almacena los objetos binarios, especialmente los PDF de consentimientos y los informes generados por el modelo de lenguaje.

Ambos sistemas se integran mediante controladores en la capa de infraestructura, garantizando que las operaciones de persistencia permanezcan desacopladas de las reglas de negocio.

El sistema, además, se conecta con proveedores externos de modelos de lenguaje, como Groq y Gemini, responsables de analizar los documentos cargados y devolver evaluaciones basadas en prompts configurables. Estas integraciones se realizan por canales HTTPS seguros, invocando los servicios desde la capa de casos de uso para mantener la coherencia con la arquitectura establecida.

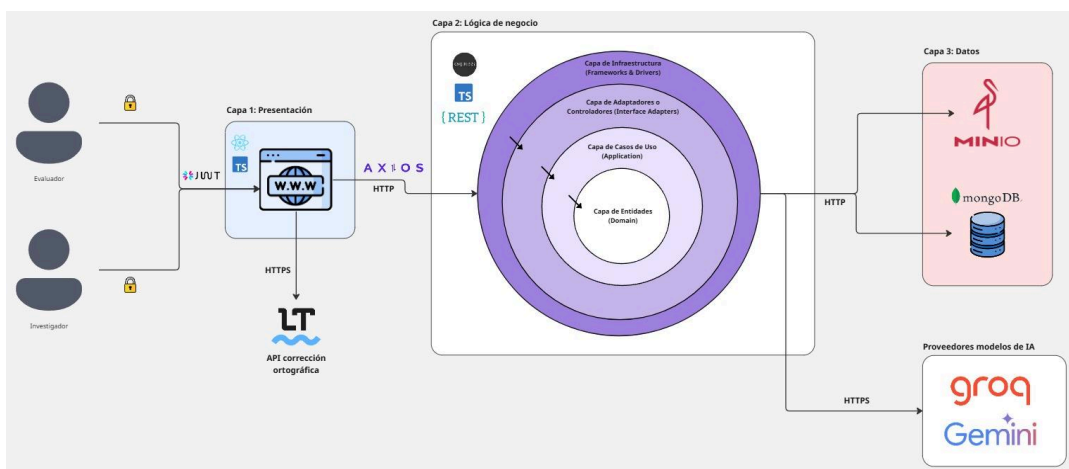


Figura 1. Diagrama de arquitectura de alto nivel (Anexo 16)

Como se puede observar, la combinación de N-Tier, Clean Architecture y Atomic Design dota al proyecto de una estructura robusta, extensible y fácilmente testeable. Este diseño no solo optimiza el desarrollo colaborativo, sino que también garantiza una trazabilidad adecuada desde la interfaz de usuario hasta la persistencia de datos y los servicios externos, alineándose con los objetivos de calidad y sostenibilidad a largo plazo de la memoria de grado. En caso de requerir mayor detalle sobre el diseño de la solución, se puede visualizar en el documento de descripción del diseño de software, en el *Anexo 11*.

1.1. Vista lógica

La plataforma adopta una arquitectura cliente-servidor basada en el modelo N-Tier, con una organización interna inspirada en los principios de Clean Architecture. Esta combinación garantiza una separación clara entre la interfaz de usuario, la lógica de negocio y el acceso a datos, favoreciendo al mismo tiempo la cohesión de cada módulo y minimizando el acoplamiento entre ellos.

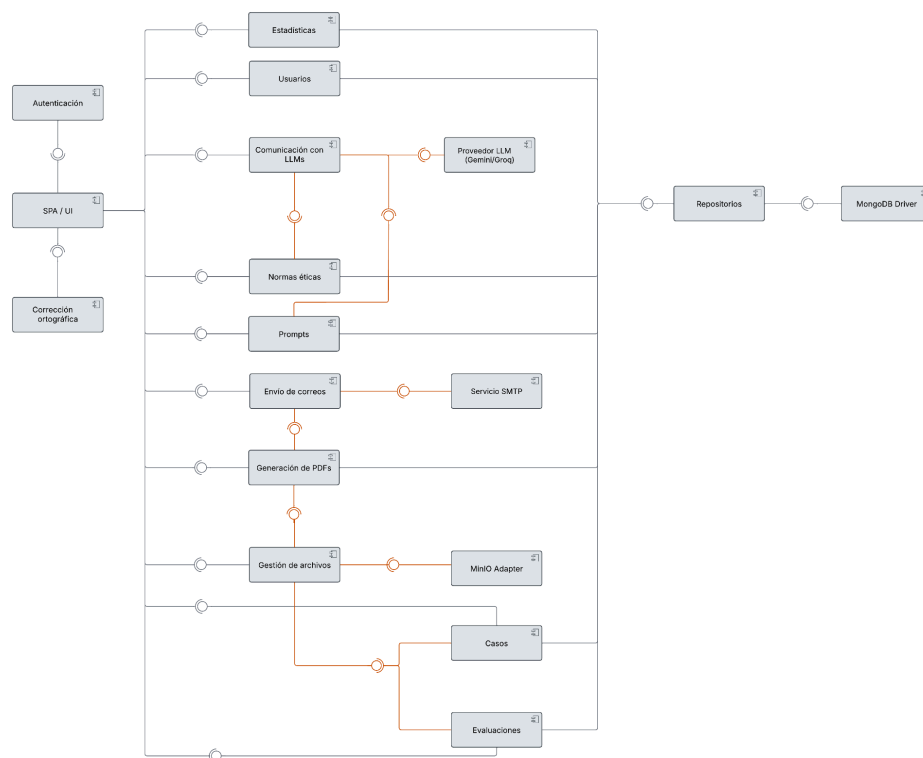


Figura 2. Diagrama de componentes (Anexo 17)

En el extremo izquierdo del diagrama se representa la SPA/UI, construida con React y TypeScript. Esta capa de presentación canaliza todas las interacciones de evaluadores e investigadores hacia el backend mediante peticiones HTTP. Además, incluye dos servicios de cliente específicos:

- *Corrección ortográfica* (LanguageTool), que ofrece validación y sugerencias en tiempo real mientras se redactan los casos.
- *Autenticación*, encargado de gestionar el ciclo de inicio de sesión, emisión y renovación de tokens JWT, y control de acceso basado en roles.

En el núcleo de la lógica de negocio se ubican los módulos funcionales más relevantes, cada uno responsable de una tarea específica dentro del flujo de trabajo de la plataforma:

- *Estadísticas*: recopila y procesa métricas de uso, generación de informes y visualizaciones de rendimiento.
- *Usuarios*: administra la información de evaluadores e investigadores, sus perfiles y roles.
- *Comunicación con LLMs*: gestiona el envío de prompts y documentos a proveedores externos (Groq/Gemini) y recibe sus respuestas, encapsulando la lógica de interacción con el modelo.
- *Normas éticas*: define, almacena y mantiene actualizadas las reglas que rigen la evaluación de los consentimientos.
- *Prompts*: gestiona los distintos textos parametrizables que alimentan al LLM según cada criterio ético.

A estos se suman otros componentes operativos fundamentales para el correcto funcionamiento de la plataforma:

- *Envío de correos*: utiliza un servicio SMTP para notificar a los usuarios, adjuntando los resultados de las evaluaciones en formato PDF.
- *Generación de PDFs*: transforma plantillas EJS en documentos PDF mediante Puppeteer, permitiendo la descarga y previsualización de informes.
- *Gestión de archivos*: interactúa con MinIO a través de un adaptador S3-compatible para almacenar y recuperar los archivos PDF de casos y resultados.
- *Casos y Evaluaciones*: implica el flujo funcional principal, desde la carga de documentos hasta la creación, revisión, re-evaluación y registro de resultados.

Todos estos módulos internos se conectan a un sistema unificado de repositorios, que abstrae las operaciones de persistencia y ofrece una interfaz consistente para el MongoDB Driver. Gracias a este diseño, el acceso a datos permanece desacoplado de la lógica de negocio, lo que facilita la realización de pruebas, el mantenimiento evolutivo y la incorporación de nuevas funcionalidades con un impacto mínimo en el sistema existente.

1.2. Vista de procesos

Esta sección describe los principales procesos funcionales del sistema desde la perspectiva del usuario, con el fin de ilustrar cómo interactúan los distintos actores, investigadores y evaluadores, con la plataforma. El objetivo es facilitar la comprensión del flujo de actividades desde la interfaz gráfica hasta la lógica interna del sistema, abarcando tanto operaciones simples como aquellas más complejas que involucran múltiples componentes.

Para representar dichos procesos, se utilizan diagramas de actividad, los cuales permiten modelar de manera estructurada y visual las secuencias de decisiones, bifurcaciones y tareas involucradas en cada interacción. Estos diagramas están alineados con el modelo de navegación de la plataforma, lo que garantiza la coherencia entre la experiencia de usuario y la arquitectura funcional del sistema.

A continuación, se presentan las descripciones de los principales procesos de la plataforma:

Proceso 1: Crear nuevo caso

Este flujo inicia cuando un usuario con rol de Investigador accede a la opción "*Crear nuevo caso*". El sistema despliega un formulario en el que se debe ingresar la información correspondiente al consentimiento informado. A medida que el usuario completa los campos, el sistema ejecuta automáticamente una verificación ortográfica utilizando la API de LanguageTool, lo que permite identificar y corregir errores en tiempo real antes de continuar con el proceso.

Una vez diligenciados todos los campos requeridos, el sistema valida que la información esté completa y que los textos no contengan errores ortográficos pendientes. Tras esta validación, se genera una vista previa del documento en formato PDF, la cual el usuario puede revisar para verificar que el contenido se ha renderizado correctamente.

Finalmente, al hacer clic en "*Guardar caso*", el sistema guarda la información ingresada en la base de datos (MongoDB) y almacena el archivo PDF generado en el sistema de archivos (MinIO). Una vez completada esta operación, el investigador es redirigido automáticamente a la sección "*Historial de casos*", donde podrá consultar el nuevo caso registrado junto con los documentos creados previamente. Para una mejor comprensión de este flujo, puede consultarse el diagrama correspondiente en el *Anexo 1*.

Proceso 2: Previsualizar o eliminar un caso

Desde la pantalla "*Historial de casos*", el investigador puede visualizar o eliminar los documentos previamente registrados. En esta vista, se presenta una tabla con los diferentes casos creados por el usuario. Al seleccionar un caso específico, se despliega un menú de opciones que permite visualizar el archivo PDF asociado o iniciar el proceso de eliminación.

Si se opta por visualizar el documento, el sistema recupera el archivo correspondiente desde MinIO y lo presenta en un visor integrado. Desde allí, el investigador puede descargar nuevamente el documento en caso de ser necesario, aunque este no puede ser modificado.

En caso de seleccionar la opción "*Eliminar*", el sistema solicita una confirmación explícita por parte del usuario. Si la acción es confirmada, se elimina tanto el archivo PDF del sistema de almacenamiento como el registro correspondiente en la base de datos (MongoDB). Si la eliminación es cancelada, el sistema no realiza ninguna modificación.

Para una mejor comprensión de este flujo, puede consultarse el diagrama correspondiente en el **Anexo 2**.

Proceso 3: Subida de archivos y creación de evaluaciones

El evaluador tiene la posibilidad de subir hasta diez archivos en formato PDF mediante una interfaz de tipo drag-and-drop. Una vez seleccionados, los archivos se listan en pantalla para su revisión, permitiendo al usuario eliminar individualmente aquellos que no desee incluir o continuar con la carga completa.

Al confirmar la carga, los archivos son almacenados en el sistema de archivos MinIO, y de forma paralela se generan evaluaciones automáticas con estado inicial "PENDIENTE" en la base de datos MongoDB. Una vez finalizado el proceso, el evaluador es redirigido automáticamente a la sección de "Historial de archivos", donde podrá gestionar, visualizar o evaluar los documentos recién subidos.

Para una mejor comprensión de este flujo, puede consultarse el diagrama correspondiente en el **Anexo 3**.

Proceso 4: Gestión de evaluaciones

Desde la sección "Historial de archivos", el evaluador puede aplicar filtros por estado de evaluación y acceder a diferentes acciones disponibles para cada caso registrado. Si una evaluación aún no ha sido procesada, se habilita la

opción "Evaluar". En cambio, si ya ha sido evaluada, el sistema presenta las opciones de "Reevaluar", "Editar", "Ver detalles", "Ver PDF" o "Eliminar".

Al seleccionar "Evaluar" o "Reevaluar", el sistema envía el archivo correspondiente, junto con los prompts configurados, al modelo de lenguaje seleccionado (Groq o Gemini). Una vez procesada la respuesta, se actualizan automáticamente las normas éticas asociadas a la evaluación.

La opción "Ver detalles" permite visualizar una tabla con el desglose de cada norma evaluada, incluyendo su estado, justificación y cita correspondiente. Por su parte, "Editar" permite modificar la información registrada de la evaluación, como el ID, estado o tipo de error. En caso de seleccionar "Eliminar", el sistema solicita confirmación y, una vez aceptada, elimina tanto el archivo del sistema de almacenamiento (MinIO) como el registro en la base de datos (MongoDB).

Para una mejor comprensión de este flujo, puede consultarse el diagrama correspondiente en el **Anexo 4**.

Proceso 5: Visualización del resultado y envío por correo

Una vez finalizada una evaluación, el evaluador puede visualizar los resultados a través de una tabla editable que muestra todas las normas éticas evaluadas. Cada norma puede ser modificada manualmente, permitiendo ajustar su estado, justificación o cita antes de formalizar el envío del informe.

Al hacer clic en "Enviar resultado", el sistema despliega un visor con el PDF generado automáticamente a partir de la evaluación, junto con un formulario que permite ingresar los datos del destinatario, seleccionar el motivo del envío y redactar un mensaje personalizado. Tras confirmar el envío, la plataforma utiliza el servicio SMTP para remitir el correo electrónico con el documento adjunto. Finalmente, se muestra una notificación que confirma el envío exitoso.

Para una mejor comprensión de este flujo, puede consultarse el diagrama correspondiente en el **Anexo 5**.

Proceso 6: Configuración de proveedor de IA, modelo, API Key y prompts

Este proceso permite a los evaluadores configurar los parámetros necesarios para la interacción con los modelos de lenguaje (LLMs) utilizados durante la evaluación automatizada de consentimientos informados. Desde la pestaña "Inteligencia artificial", el usuario puede realizar las siguientes acciones:

- Seleccionar el proveedor de LLM (actualmente, Groq o Gemini).
- Ingresar y actualizar su clave de acceso (API Key).

- Elegir el modelo específico asociado al proveedor seleccionado.
- Editar los prompts personalizados que orientan el comportamiento del modelo durante la evaluación automática.

Una vez que se guardan los cambios, el sistema valida la consistencia de los datos ingresados y almacena de forma segura la configuración actualizada en la base de datos.

Para una mejor comprensión de este flujo, puede consultarse el diagrama correspondiente en el *Anexo 6*.

Proceso 7: Visualización de estadísticas de evaluación

Desde la sección "Estadísticas", el evaluador puede seleccionar un rango de fechas para consultar las métricas clave del sistema, entre las que se incluyen:

- Total de evaluaciones realizadas
- Cantidad de consentimientos aprobados y rechazados
- Tasa de devolución
- Tiempos promedio de evaluación

Una vez aplicado el filtro temporal, el sistema presenta tarjetas resumen con los principales indicadores y genera visualizaciones gráficas, como líneas de tiempo y gráficos de torta, que facilitan el análisis del desempeño y la identificación de tendencias en el proceso de evaluación.

Para una mejor comprensión de este flujo, puede consultarse el diagrama correspondiente en el *Anexo 7*.

1.3. Vista física

La arquitectura física del sistema está diseñada para operar, en su versión final, dentro de la nube privada del Hospital Universitario San Ignacio y la Pontificia Universidad Javeriana, garantizando un entorno controlado, seguro y alineado con los estándares institucionales. Actualmente, el sistema se encuentra estructurado bajo un modelo cliente-servidor, donde cada componente lógico se ejecuta dentro de contenedores Docker, facilitando su portabilidad, aislamiento y administración en entornos distribuidos.

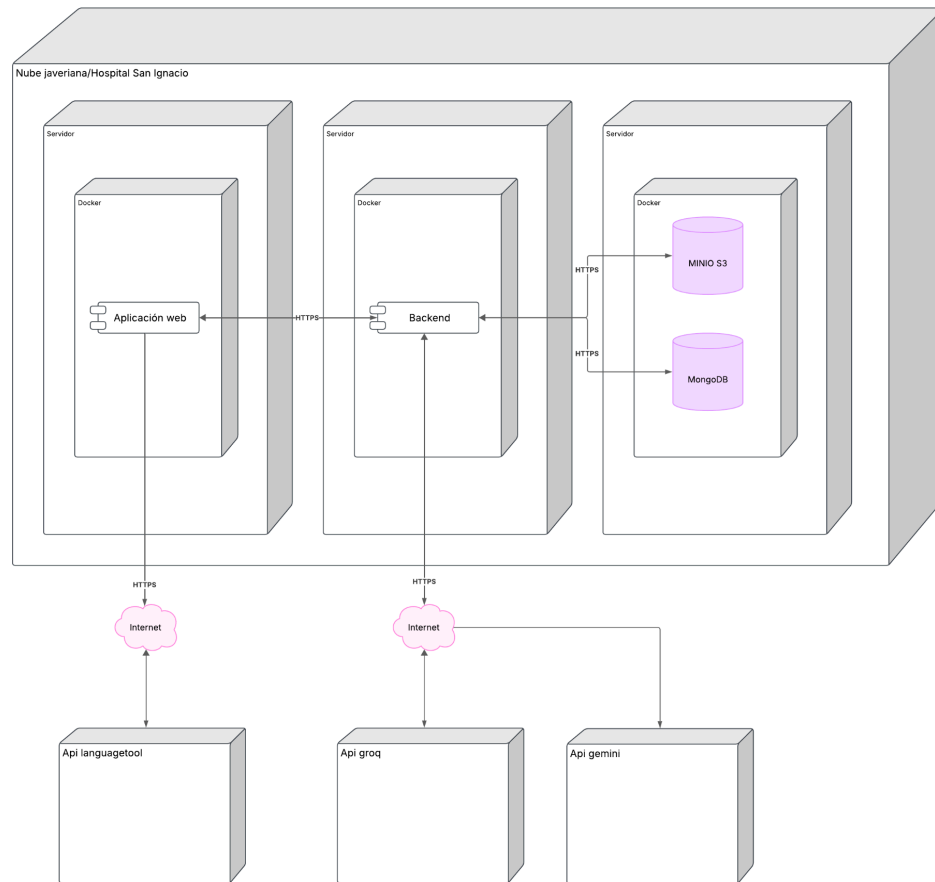


Figura 3. Diagrama de despliegue (Anexo 18)

A continuación, se describen los tres servidores principales que conforman la infraestructura física:

- *Aplicación web*

Este servidor contiene un contenedor Docker que ejecuta la interfaz de usuario (SPA), desarrollada con React y TypeScript. La aplicación es accesible por medio del protocolo HTTPS y está diseñada para ser utilizada por los perfiles de investigador y evaluador.

Desde esta capa de presentación se realizan llamadas HTTP seguras al backend, así como integraciones directas con servicios externos como la API de LanguageTool para realizar corrección ortográfica en tiempo real durante la creación de casos.

- *Backend*

Este servidor aloja un contenedor que ejecuta el backend construido en Express.js, organizado bajo los principios de Clean Architecture. Este

componente expone una API REST consumida por la interfaz web y es responsable de toda la lógica de negocio del sistema:

- Gestión de autenticación y control de acceso
- Administración de casos y evaluaciones
- Generación y envío de PDFs
- Comunicación con servicios de IA como Groq o Gemini para la evaluación automática de documentos
- Envío de notificaciones a través de SMTP

- *Persistencia*

Este servidor contiene dos contenedores que se encargan del almacenamiento estructurado y no estructurado del sistema:

- *MongoDB*: almacena los datos estructurados como usuarios, evaluaciones, prompts, configuraciones y resultados.
- *MinIO*: actúa como un sistema de almacenamiento de objetos compatible con el estándar Amazon S3, utilizado para guardar los archivos PDF de consentimientos informados y reportes generados.

- *Servicios externos (APIs públicas)*

El sistema interactúa con dos servicios externos a través de conexiones HTTPS:

- *API LanguageTool*: accedida directamente por el frontend para ejecutar tareas de corrección ortográfica sobre los textos ingresados por el usuario.
- *API Groq / Gemini*: consumida desde el backend para enviar documentos y prompts personalizados que serán procesados por los modelos de lenguaje y devueltos como evaluaciones automáticas.

Todos los servicios están contenidos de forma aislada mediante Docker, lo que permite desplegar múltiples réplicas en caso de requerir escalabilidad horizontal. La comunicación entre contenedores se realiza a través de redes internas seguras, mientras que toda interacción con usuarios o servicios externos se lleva a cabo exclusivamente vía HTTPS, garantizando así la confidencialidad e integridad de los datos.

Este modelo de despliegue asegura un entorno robusto, escalable y alineado con los requerimientos técnicos y éticos del Hospital Universitario San Ignacio, y permite al equipo de desarrollo comprender de manera precisa la ubicación y operación física de cada componente del sistema.

DESARROLLO DE LA SOLUCIÓN

1. Implementación

La implementación de la plataforma se llevó a cabo siguiendo el marco metodológico ágil Scrum, permitiendo el desarrollo incremental de funcionalidades mediante iteraciones quincenales (sprints). Durante cada sprint se construyeron módulos clave del sistema, incluyendo autenticación, gestión de documentos, análisis automatizado de los consentimientos informados y visualización de resultados. Se emplearon tecnologías modernas como Node.js con TypeScript para el backend, React con TypeScript bajo principios de Atomic Design para el frontend, MongoDB como base de datos documental, y MinIO como solución de almacenamiento de archivos compatible con S3.

El desarrollo se organizó en cinco fases: levantamiento de requerimientos, desarrollo iterativo de funcionalidades básicas, integración del modelo de IA, pruebas funcionales y cierre del proyecto. En cada fase se documentaron avances mediante revisiones de sprint y retrospectivas, asegurando la trazabilidad del proceso. El control de versiones se manejó a través de Git, mientras que los entornos de desarrollo y pruebas fueron orquestados con Docker y docker-compose para garantizar replicabilidad y portabilidad.

2. Producto Final

El producto final es una plataforma web orientada a asistir al Comité de Ética del Hospital Universitario San Ignacio (HUSI) en la verificación de documentos de consentimiento informado. Esta herramienta permite al evaluador cargar un documento, visualizarlo, analizarlo automáticamente mediante un modelo de lenguaje ajustado con prompt engineering, y recibir un resumen estructurado que indica si se cumple o no con los criterios éticos definidos.

Entre sus características destacan: carga y almacenamiento seguro de documentos en formato PDF, visualización integrada del documento original junto a los hallazgos del modelo, edición de criterios manuales, historial de casos evaluados, control de acceso por roles (evaluador e investigador) y una interfaz sencilla adaptada al flujo de trabajo real del comité. Además, la solución puede ser desplegada fácilmente en entornos locales o en la nube, sin depender de licencias comerciales, lo que facilita su sostenibilidad y escalabilidad. A continuación, se presentan las interfaces del producto final. En caso de ser necesario, se pueden ver las imágenes de las interfaces más a detalle en el *Anexo 19*.

2.1. Interfaces del Login

Estas interfaces son comunes en toda la plataforma, por lo que no varían entre investigadores y evaluadores. Sin embargo, es importante destacar que cualquier persona que se registre en la plataforma será asignada automáticamente con el rol de investigador, ya que los evaluadores solo pueden ser creados en este momento por el personal de soporte.

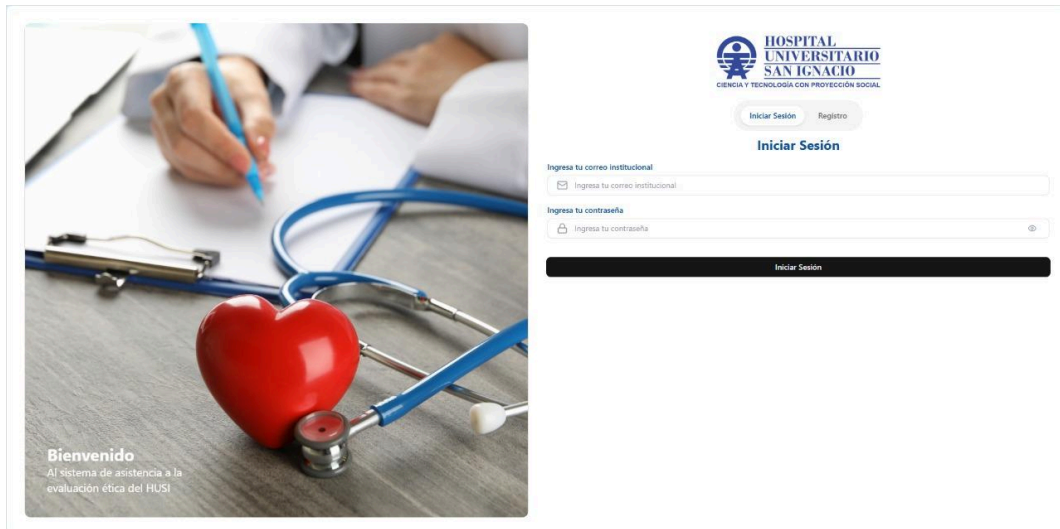


Figura 4. Pantalla de Login

En esta interfaz se brinda a los usuarios la posibilidad de acceder al sistema mediante sus credenciales institucionales, asegurando un inicio de sesión seguro y controlado. El diseño está dividido en dos secciones: una visual con elementos representativos del entorno clínico y otra funcional con el formulario de autenticación. Esta organización no solo refuerza la identidad institucional del HUSI, sino que también facilita una experiencia de usuario clara e intuitiva desde el primer ingreso.

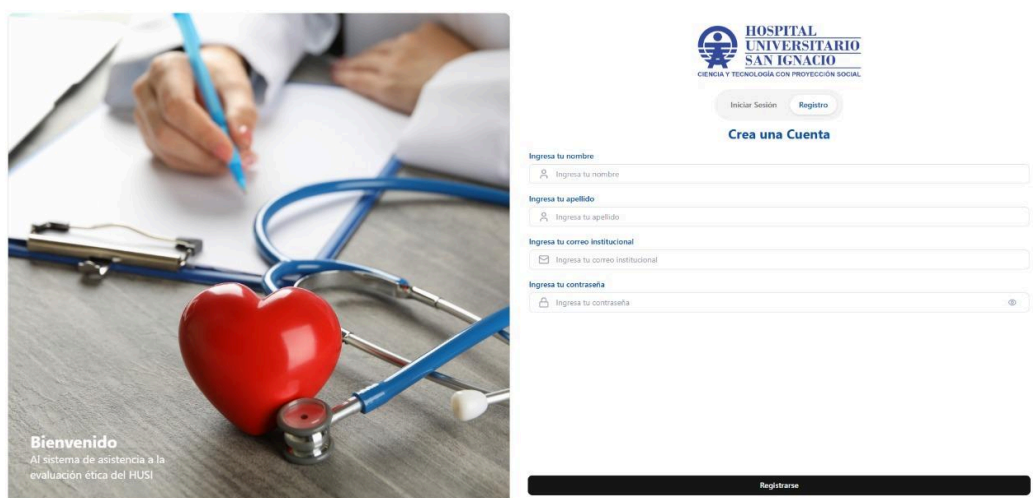


Figura 5. Pantalla de Registro

En esta interfaz se ofrece a los nuevos usuarios la posibilidad de crear una cuenta para acceder al sistema. El formulario de registro solicita información básica como nombre, apellido, correo institucional y contraseña, asegurando que el acceso esté restringido a miembros autorizados. Al igual que en la pantalla de inicio de sesión, se mantiene la identidad visual institucional del HUSI, lo que aporta coherencia y confianza. Esta funcionalidad es esencial para permitir que los investigadores se integren de manera autónoma a la plataforma, mientras que los evaluadores son creados exclusivamente por el equipo administrador.

2.2. Interfaces Evaluador

Para el evaluador, las interfaces fueron diseñadas para facilitar una navegación intuitiva y eficiente, centrada en el flujo de revisión documental. A continuación, se presentan sus respectivas pantallas con la explicación de las funcionalidades que corresponden a ella.

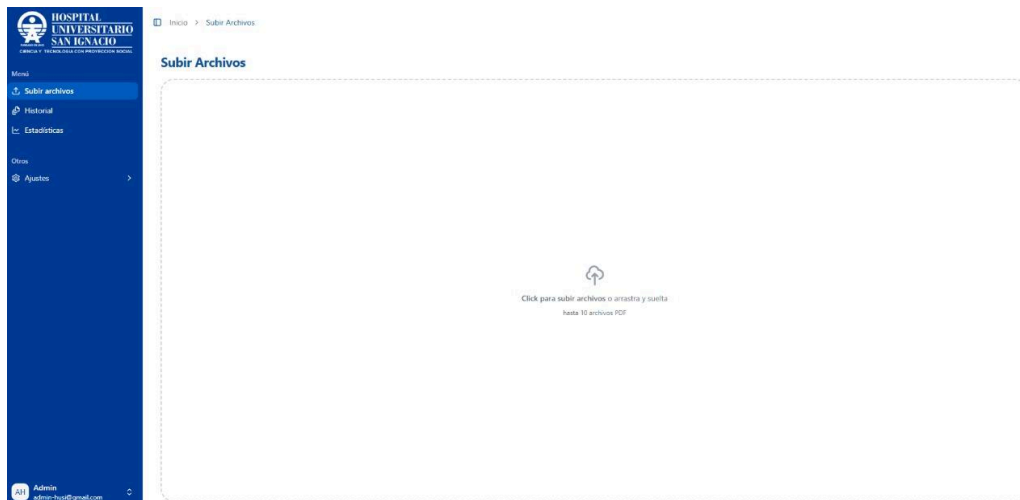


Figura 6. Pantalla para subir archivos

Esta interfaz corresponde a la sección *"Subir Archivos"*, donde el evaluador puede cargar hasta 10 documentos en formato PDF para su posterior evaluación. La plataforma permite arrastrar y soltar los archivos o seleccionarlos manualmente, facilitando un proceso ágil e intuitivo de carga.

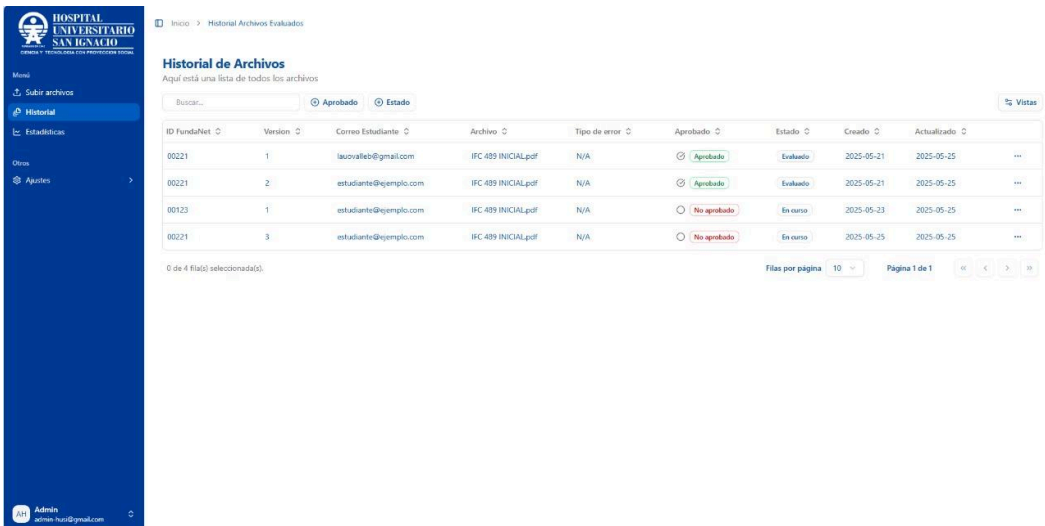


Figura 7. Pantalla Historial de Archivos Evaluador

Esta interfaz corresponde a la sección *"Historial de Archivos"*, del evaluador, donde se presenta una lista detallada de todos los documentos evaluados en la plataforma. Muestra información como el ID Fundanet, correo del evaluador, nombre del archivo, estado de evaluación, aprobación y fechas de creación y actualización. Permite hacer seguimiento al proceso de revisión y consultar rápidamente el estado de cada caso.

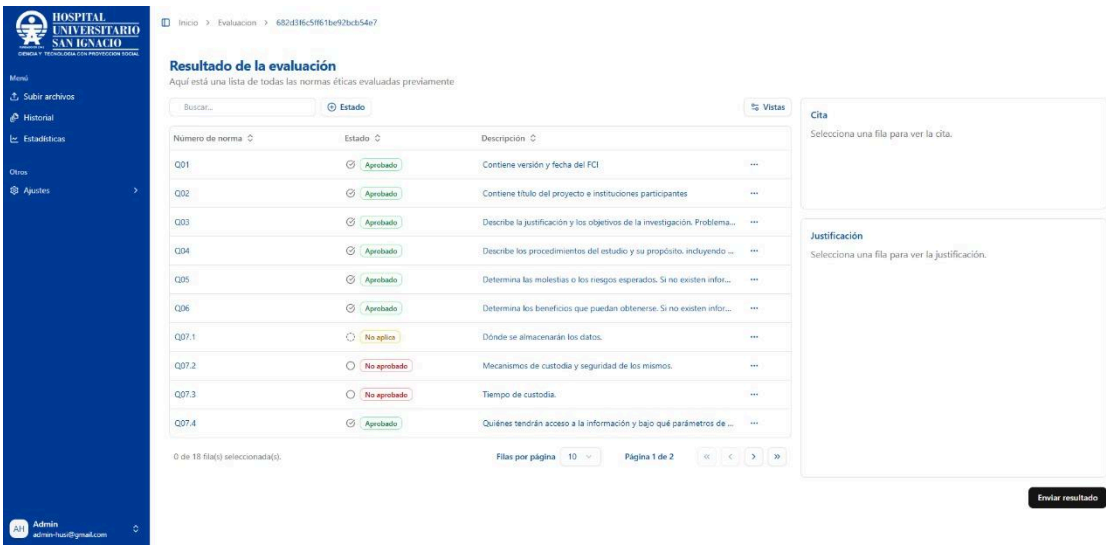


Figura 8. Pantalla de resultados de evaluación

Esta interfaz muestra el resultado detallado de una evaluación ética. Cada fila corresponde a una norma evaluada por el modelo, indicando su estado, descripción, y permitiendo al evaluador consultar o editar su justificación y cita en los paneles laterales. Facilita la revisión manual y el ajuste de los resultados antes de su envío final.

Enviar resultado de la evaluación

Correo de destino:

Motivo del correo:

Mensaje adicional:

Vista previa PDF: 1 / 3, 100%

GESTIÓN DE INVESTIGACIÓN CLÍNICA
RESULTADOS DE LA EVALUACIÓN AUTOMÁTICA

Número de Norma	Descripción	Justificación	Estado
Q01	Contiene versión y fecha del PCI	La versión del documento es 1.0 y la fecha es 26 de octubre de 2023.	APROBADO
Q02	Contiene título del proyecto e instituciones participantes	El título del proyecto es "Hallazgos en videocapsuloscopia en población con enfermedad reumática autoinmune sistémica". El proyecto es realizado por el Hospital Universitario San Ignacio y la Pontificia Universidad Javeriana.	APROBADO
Q03	Describe la justificación y los objetivos de la investigación, problema, pertinencia del estudio, objetivo general	Existen diversas técnicas para valorar la microcirculación a nivel del lecho ungueal. La videocapsuloscopia es una de ellas y se ha convertido en una herramienta diagnóstica rutinaria en el campo de la reumatología para la valoración del fenómeno de Raynaud. (...) Establecen las características de las videocapsuloscopias en adultos sin antecedente de enfermedad reumática autoinmune sistémica en fenómeno de Raynaud, que residen en Colombia en una ciudad con altura mayor a 2500 metros sobre el nivel del mar, de acuerdo con el consenso internacional de reporte establecido por la Alianza Europea de Asociaciones de Reumatología (EULAR). Para este estudio se realizó videocapsuloscopia, la cual consta de un microscopio digital portátil, un	APROBADO

Figura 9. Pantalla envío de correo

En esta interfaz se le permite al evaluador enviar por correo electrónico los resultados de una evaluación ética. Incluye una vista previa del informe en PDF, campos para ingresar el correo del destinatario, seleccionar el motivo del envío y añadir un mensaje adicional. Facilita la formalización y trazabilidad del proceso de evaluación mediante un canal institucional.

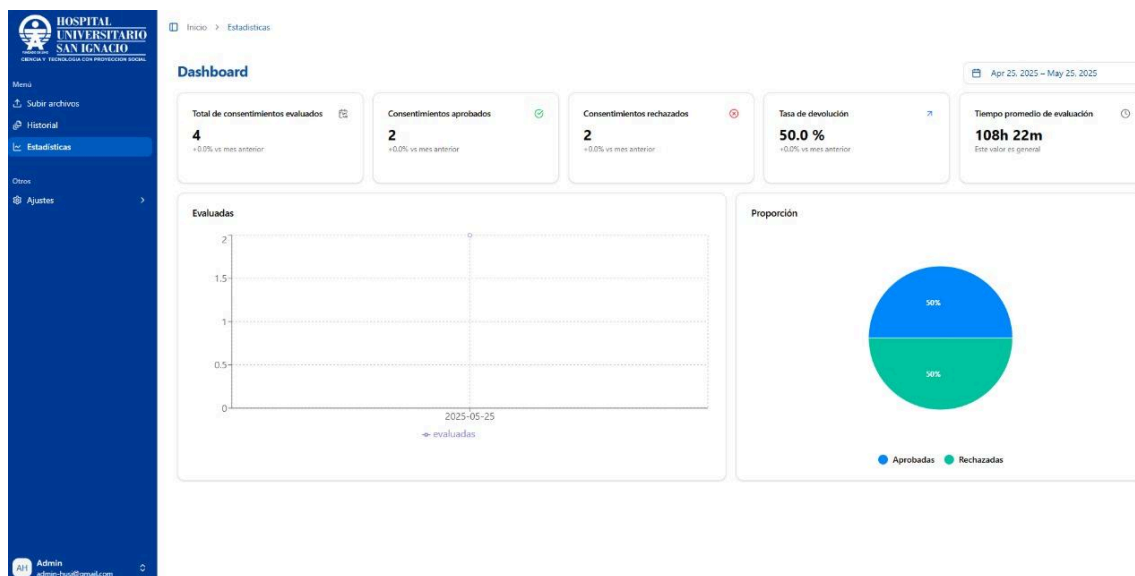


Figura 10. Pantalla de Estadísticas

Esta interfaz corresponde a la sección "Estadísticas", donde el evaluador puede visualizar indicadores clave del sistema, como el número total de consentimientos evaluados, aprobados y rechazados, la tasa de devolución y el tiempo promedio de evaluación. Incluye también gráficas dinámicas que permiten analizar el desempeño del proceso de revisión en un periodo determinado.

HOSPITAL UNIVERSITARIO SAN IGNACIO

Inicio > Ajustes > Cuenta

Ajustes
Administre la configuración de su cuenta y establezca sus preferencias.

Cuenta

Mi Cuenta
Actualiza la información de la cuenta.

Nombre
Admin

Apellido
Husi

Correo electrónico
admin-husi@gmail.com

Actualizar Cuenta

Contraseña
Actualiza tu contraseña.

Contraseña actual
Contraseña actual

Nueva contraseña
Nueva contraseña

Figura 11. Pantalla de Ajustes → Cuenta

Esta interfaz corresponde a la sección "*Cuenta*" dentro del módulo de Ajustes, donde el usuario puede actualizar su información personal, como nombre, apellido y correo electrónico, así como cambiar su contraseña de acceso a la plataforma. Facilita la gestión de datos de perfil de forma segura y centralizada.

HOSPITAL UNIVERSITARIO SAN IGNACIO

Inicio > Ajustes > IA

Ajustes
Administre la configuración de su cuenta y establezca sus preferencias.

Inteligencia artificial

Proveedor actual: gemini
Selecciona el proveedor de IA que utilizará la plataforma.

Proveedor
geminí

Seleccionar proveedor

Proveedor y API Key de gemini
Configura la clave para el proveedor de IA seleccionado.

Nueva API Key
Nueva API Key

Actualizar API Key

Modelo por defecto de gemini
Selecciona el modelo que usará la plataforma.

Modelo
geminí-2.0-flash

Cambiar Modelo de IA

Prompts
Modifica los prompts de la plataforma.

Prompt 1
Contiene versión y fecha del FCI

Prompt 2
Contiene título del proyecto e instituciones participantes

Figura 12. Pantalla de Ajustes → Inteligencia artificial

Esta interfaz corresponde a la sección de configuración de inteligencia artificial, donde el evaluador puede seleccionar el proveedor de modelo de lenguaje (por ejemplo, Gemini), actualizar su API Key, elegir el modelo predeterminado y editar los *prompts* personalizados que guían la evaluación automática de los documentos. Esta funcionalidad permite adaptar el comportamiento del sistema de IA a los criterios específicos del Comité de Ética.

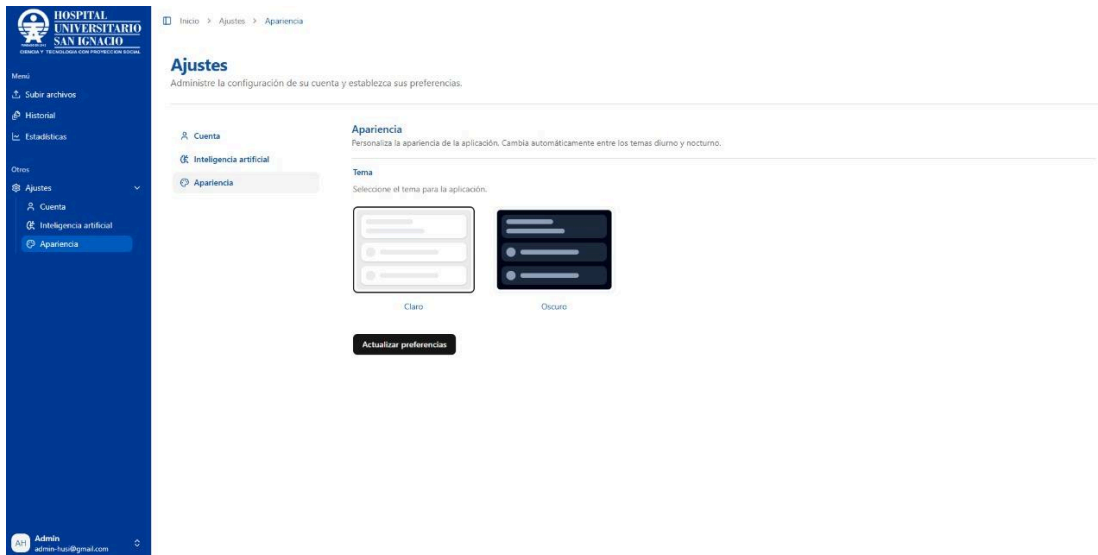


Figura 13. Pantalla de Ajustes → Apariencia

En esta interfaz se ofrece la posibilidad de cambiar la apariencia de la plataforma, permitiendo al usuario alternar entre modo claro y modo oscuro, según sus preferencias. Esta funcionalidad resulta especialmente útil para quienes trabajan en horarios nocturnos, ya que el modo oscuro reduce la fatiga visual y mejora la experiencia de uso en ambientes con poca iluminación.

2.3. Interfaces Investigador

La interfaz del investigador fue diseñada para facilitar la creación y carga de nuevos documentos de consentimiento informado. Desde su panel principal, el investigador puede iniciar la elaboración del documento mediante un formulario dinámico, estructurado conforme a los requisitos éticos definidos por el HUSI. Este formulario incluye secciones prediseñadas y campos obligatorios que ayudan a prevenir omisiones comunes, lo que contribuye a reducir significativamente la tasa de devoluciones.

Una vez finalizado el formulario, el investigador puede exportar el documento en formato PDF y enviarlo al comité para su evaluación. Además, cuenta con un historial de envíos, consulta del estado de revisión y acceso a los comentarios emitidos por el evaluador.

A continuación, se presentan las principales interfaces disponibles para el investigador:



Inicio > Crear Nuevo Caso

Crear nuevo caso

Llena cada uno de los campos requeridos del consentimiento informado

Cabecera

Versión

Ingresar la versión del FCI

Código

Ingresar el código del FCI

Fecha

Ingresar una fecha

Introducción

Información general

Autorización

Previsualizar PDF

Figura 14. Pantalla de Crear nuevo Caso

Esta interfaz corresponde al formulario de creación de un nuevo caso por parte del investigador. Está organizado en secciones desplegadas que agrupan los campos requeridos del consentimiento informado, como *Cabecera*, *Introducción*, *Información general* y *Autorización*. Una vez completado el formulario, el usuario puede generar una previsualización en PDF del documento antes de guardarlo o enviarlo para evaluación.



Inicio > Historial Casos

Historial de Archivos

Aquí está una lista de todos los archivos

Buscar...

Vistas

Nombre Proyecto	Versión	Código	Creado	Actualizado	
2	2	2	2025-05-23	2025-05-23	...
Proyecto 1	1	111	2025-05-25	2025-05-25	...

0 de 2 fila(s) seleccionada(s).

Filas por página: 10

Página 1 de 1

< > >>

Figura 15. Pantalla Historial Archivos del Investigador

Esta interfaz corresponde al *"Historial de archivos"* del investigador, donde se listan todos los documentos de consentimiento informado previamente creados. Muestra información clave como el nombre del proyecto, versión, código, y fechas de creación y actualización, permitiendo al usuario realizar seguimiento y gestionar fácilmente sus casos registrados.

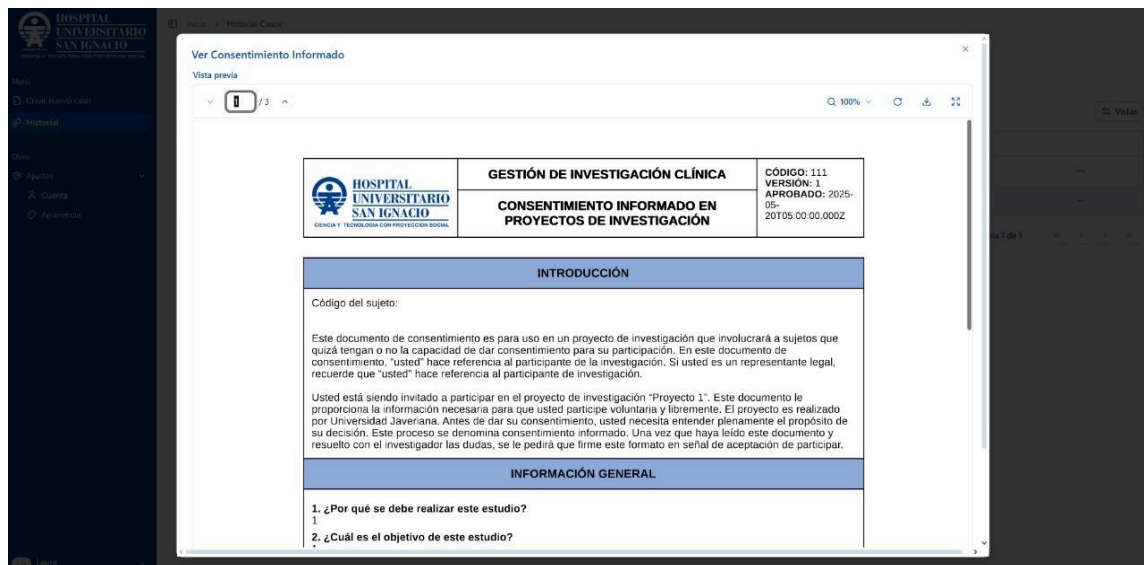


Figura 16. Pantalla de Previsualización de PDF

Esta interfaz permite al investigador visualizar en detalle el documento de consentimiento informado antes de su envío o evaluación. A través de un visor PDF integrado, el usuario puede revisar la estructura, el contenido y el formato del documento, asegurándose de que cumple con los requisitos éticos establecidos.

VALIDACIÓN Y RESULTADOS

1. Pruebas de Software

La plataforma desarrollada fue sometida a un proceso exhaustivo de pruebas para garantizar su funcionamiento adecuado, rendimiento, bajo carga, fiabilidad de sus componentes, y aceptación por parte de los usuarios finales. Las pruebas se diseñaron e implementaron conforme a buenas prácticas de aseguramiento de calidad, alineadas con la norma ISO 25010. Para obtener una información más detallada sobre las pruebas realizadas, recomendamos mirar el documento de Pruebas, el cual se encuentra en el *Anexo 12*.

1.1. Metodología de evaluación

Se adoptó una estrategia de pruebas escalonada que incluye pruebas de integración, pruebas de carga y estrés, pruebas end-to-end (E2E) y pruebas de aceptación por parte de usuarios reales (UAT). Estas se ejecutaron sobre los principales módulos funcionales y flujos críticos del sistema, utilizando un entorno local basado en Docker y herramientas especializadas como Jest, Artillery y Playwright.

2. Resultados Pruebas

2.1. Pruebas de Integración

Las pruebas de integración se llevaron a cabo utilizando las herramientas Jest y Supertest, las cuales permitieron simular peticiones HTTP de forma automatizada sin necesidad de desplegar el servidor en un entorno productivo. Estas pruebas se enfocaron en verificar la correcta interacción entre los distintos controladores y servicios del backend, evaluando cómo se comportan en conjunto al procesar datos, invocar lógica de negocio y acceder a la capa de persistencia.

Se alcanzó una cobertura del 90% en los endpoints de los módulos principales: autenticación, gestión de casos, normas éticas, evaluaciones, integración con modelos de inteligencia artificial (IA), generación de documentos PDF, administración de prompts, envío de correos electrónicos, estadísticas del sistema y administración de usuarios. Cada prueba fue diseñada para validar operaciones CRUD, flujos de datos, validaciones y manejo de errores, garantizando que cada controlador respondiera adecuadamente bajo múltiples escenarios funcionales.

Además, se realizaron mockeos controlados en los módulos que dependían de servicios externos, como el renderizado de PDFs y el envío de correos SMTP, con el fin de asegurar la estabilidad de las pruebas y evitar efectos colaterales. Esta estrategia permitió identificar y corregir errores lógicos, inconsistencias entre módulos y omisiones en la validación de datos.

2.2. Pruebas de Carga y estrés

Las pruebas de carga y estrés se llevaron a cabo utilizando la herramienta Artillery, con el objetivo de evaluar la estabilidad, el rendimiento y la capacidad de respuesta de la plataforma ante escenarios de alta concurrencia. Los requerimientos establecidos por el HUSI fueron hasta 25 usuarios virtuales concurrentes. Sin embargo, se simularon múltiples escenarios con hasta 50 usuarios virtuales concurrentes para asegurar su funcionamiento, reproduciendo flujos completos de interacción con los distintos módulos de la aplicación.

Los resultados obtenidos evidenciaron un comportamiento estable del sistema, sin errores y con tiempos de respuesta dentro de los márgenes aceptables para una plataforma hospitalaria en producción:

- *Evaluación con LLM:* A pesar de tratarse del flujo más intensivo computacionalmente, presentó una latencia alta esperada (~10 s en p99) debido a la interacción con modelos de lenguaje externos. Sin

embargo, logró una tasa de éxito del 100 %, sin bloquear otros procesos ni generar errores en las respuestas.

- *Gestión de usuarios, casos, evaluaciones y prompts*: Estos módulos mantuvieron un throughput sostenido de hasta 12 solicitudes por segundo, con latencias promedio por debajo de 2 segundos, incluso durante operaciones complejas como el cambio de contraseña o la eliminación de evaluaciones. Se comprobó que las tareas más exigentes en procesamiento no afectaron la fluidez del sistema.
- *Autenticación y módulo de estadísticas*: Ambos mantuvieron un desempeño sub-segundo, con $p95 < 130$ ms en todos los casos, incluso bajo carga máxima. Esto confirma la eficiencia del módulo de autenticación y generación de vistas estadísticas.

En todos los escenarios, el sistema demostró ser capaz de manejar adecuadamente la carga objetivo sin comprometer la experiencia del usuario. La constancia de los códigos y la ausencia total de errores críticos respaldan la solidez del backend en condiciones de uso intensivo.

Estas pruebas no solo validan la arquitectura y la implementación técnica, sino que también aportan confianza sobre la viabilidad de la plataforma en entornos reales, donde múltiples investigadores y evaluadores pueden estar activos de manera simultánea.

2.3. Pruebas End-to-end

Las pruebas end-to-end (E2E) se implementaron utilizando Playwright, con el propósito de validar el comportamiento completo del sistema desde la perspectiva de los usuarios finales. Estas pruebas automatizadas reprodujeron los flujos principales de los dos perfiles clave de la plataforma: evaluador e investigador.

El objetivo fue verificar que los distintos componentes del sistema funcionaran correctamente en conjunto, garantizando la coherencia funcional, la integridad de los datos y la fluidez en la experiencia del usuario a través de distintos escenarios reales de uso.

Los resultados obtenidos fueron satisfactorios:

- Todos los flujos críticos se completaron sin errores, confirmando que el sistema permite desde el inicio de sesión hasta la generación y evaluación de documentos, sin interrupciones ni pérdidas de información.

- Se validó la correcta interacción entre el frontend, backend y servicios externos (como modelos de IA y servidores SMTP), confirmando la robustez de la plataforma bajo operaciones complejas y encadenadas.
- Se verificó la compatibilidad multinavegador en Chrome, Safari, WebKit y Edge, sin presentar diferencias funcionales o visuales significativas.
- Las pruebas demostraron una alta estabilidad en formularios dinámicos, carga y manipulación de documentos, y operaciones CRUD, incluso tras múltiples iteraciones.
- Se integraron métricas de trazabilidad (capturas, videos, logs y trazas) que facilitaron la validación de cada ejecución y permitirán futuras tareas de mantenimiento o regresión.

Las pruebas E2E confirmaron que la plataforma cumple con los requisitos funcionales y de usabilidad esperados por el Comité de Ética del HUSI, asegurando una experiencia de usuario fluida, predecible y libre de fallos en los flujos más importantes del sistema.

2.4. Pruebas de aceptación del usuario (UAT)

Las pruebas UAT Se realizaron con 20 participantes sin experiencia técnica previa. Tras interactuar con la plataforma, completaron una encuesta de 15 ítems bajo escala Likert (1 a 5). Los resultados promedio por módulo superaron el valor de 4,2 en todas las dimensiones evaluadas (facilidad y claridad), destacando los módulos de creación de casos (4,65), autenticación (4,4) y subida de archivos (4,55). Lo que confirma una experiencia de usuario intuitiva y amigable.

Además, la consistencia en las valoraciones (con desviaciones estándar generalmente inferiores a 0,7) sugiere que la percepción positiva fue compartida por la mayoría de los participantes, sin grandes dispersiones ni puntos críticos. Los comentarios cualitativos recolectados reforzaron la impresión general de que la plataforma es clara, fácil de navegar y adecuada para usuarios con conocimientos limitados en tecnología.

Los resultados de esta prueba de aceptación validan que la solución propuesta cumple con los requisitos de usabilidad y accesibilidad esperados, y puede ser adoptada por investigadores y evaluadores del Hospital Universitario San Ignacio sin necesidad de capacitación técnica adicional.

3. Validación Arquitectónica

La arquitectura basada en contenedores (Docker) demostró un alto grado de aislamiento, escalabilidad y confiabilidad. Los módulos frontend (React), backend (Express con Bun) y servicios de soporte (MongoDB, MinIO) interactuaron

correctamente bajo condiciones de uso reales y simuladas. Las pruebas demostraron que la separación por capas (presentación, lógica de negocio, persistencia) contribuyó a la mantenibilidad y robustez del sistema. Se validó también la correcta gestión de dependencias externas (LLM, SMTP) sin interrupciones ni pérdidas de datos, confirmando la viabilidad de la arquitectura N-Tier planteada para el entorno hospitalario del HUSI.

Adicionalmente, los resultados obtenidos en las pruebas de integración, carga y extremo a extremo permitieron verificar que la arquitectura permite una escalabilidad horizontal eficiente, así como una adecuada distribución de la carga entre módulos. La implementación en contenedores favoreció la reproducción del entorno en diferentes equipos de desarrollo y validación, lo que facilitó la colaboración del equipo y la consistencia en los resultados obtenidos.

El uso de herramientas de automatización (Jest, Artillery, Playwright) integradas directamente en el ecosistema de desarrollo permitió validar la interoperabilidad de los servicios, la tolerancia a fallos y la independencia de los módulos. La arquitectura también se mostró resiliente ante picos de tráfico, manteniendo la calidad del servicio y sin afectar la experiencia del usuario final.

La arquitectura propuesta no solo cumple con los requerimientos técnicos del sistema, sino que además se alinea con los principios de escalabilidad, mantenibilidad y modularidad necesarios para futuras ampliaciones o integraciones con otros sistemas del Hospital Universitario San Ignacio.

4. Validación de Modelo IA

La selección de los proveedores de modelos de Inteligencia Artificial (IA) se fundamentó en un análisis comparativo, orientado a identificar las soluciones que mejor se adaptaran a los requerimientos de diversidad, calidad y accesibilidad del sistema de evaluación ética. Este proceso condujo a la elección estratégica de Gemini y Groq como plataformas complementarias, asegurando una base tecnológica versátil y robusta para el procesamiento y análisis de lenguaje natural en el contexto de los consentimientos informados.

La plataforma Groq fue seleccionada primordialmente por su extensa oferta de modelos de IA, incluyendo variantes de alto rendimiento como DeepSeek y Llama. Esta diversidad es fundamental, ya que permite al sistema acceder a modelos con distintas arquitecturas y capacidades de "potencia" o especialización. Aunque su capa de acceso gratuito presenta ciertas limitaciones en cuanto a volumen de uso, la amplitud de su catálogo de modelos ofrece una ventaja significativa para realizar validaciones, enriqueciendo la fiabilidad de los análisis.

Por su parte, Gemini se destacó por ofrecer una generosa y extensa capa gratuita, acompañada de modelos de IA de alta calidad y eficiencia. Esta característica es crucial para garantizar la sostenibilidad operativa y la usabilidad del sistema, permitiendo procesar un volumen considerable de solicitudes sin incurrir en costos prohibitivos por consumo de tokens. La calidad intrínseca de los modelos de Gemini asegura que la accesibilidad no comprometa la precisión ni la profundidad de las evaluaciones éticas generadas.

La decisión de integrar y conservar ambos proveedores, Gemini y Groq, responde a una estrategia deliberada de complementariedad. Esta aproximación dual permite al sistema beneficiarse de la amplitud de modelos de Groq para escenarios de validación específicos y de la eficiencia costo-beneficio de Gemini para las operaciones regulares. De esta manera, se garantiza que el personal evaluador disponga de una variedad de herramientas y modelos de IA, permitiéndoles seleccionar el recurso más adecuado según las necesidades particulares de cada evaluación, optimizando tanto la calidad del análisis como la eficiencia del proceso global.

Si se desea conocer más a detalle las métricas o información específica de cada uno de los modelos implementados en la plataforma ver *Anexo 20*.

CONCLUSIONES

1. Conclusiones

La implementación de una plataforma digital para asistir al Comité de Ética del Hospital Universitario San Ignacio en la verificación de consentimientos informados representa un avance significativo en la modernización de los procesos de evaluación ética. A lo largo del desarrollo, se logró identificar los principales cuellos de botella que enfrentaba el comité, como la alta carga operativa, la subjetividad en la revisión manual y la alta tasa de devoluciones por incompletitud, y se diseñó una solución tecnológica adaptada a estas necesidades reales.

El uso de tecnologías abiertas y el aprovechamiento de modelos de lenguaje mediante ingeniería de prompts permitieron construir un sistema funcional, sostenible y alineado tanto con los estándares éticos como con los recursos disponibles. Esta solución no solo agiliza los tiempos de revisión, sino que también mejora la trazabilidad y consistencia en las decisiones del comité, sin desplazar el criterio humano, sino fortaleciéndolo.

Además, el enfoque metodológico ágil facilitó la validación continua de funcionalidades con los actores involucrados, asegurando que el producto final

respondiera verdaderamente a los problemas identificados. En términos generales, el proyecto demostró que es posible integrar inteligencia artificial en entornos clínicos-académicos de forma responsable y útil, generando valor tangible para la institución y, por extensión, para la investigación biomédica en Colombia.

2. Análisis del impacto

El impacto de esta solución se refleja tanto a nivel operativo como institucional. En el plano operativo, la automatización parcial del proceso de revisión ética permite reducir tiempos, minimizar errores comunes y evitar retrabajos, lo cual libera tiempo para que los evaluadores puedan concentrarse en casos complejos que sí requieren mayor juicio profesional.

Desde el punto de vista institucional, la plataforma contribuye al fortalecimiento del HUSI como referente nacional en investigación clínica con enfoque ético, al adoptar herramientas tecnológicas que garantizan un control riguroso, trazable y estandarizado del proceso. Además, promueve una cultura de mejora continua en la gestión ética de proyectos de investigación, algo fundamental en entornos donde el volumen y la complejidad de los estudios seguirá en aumento.

Finalmente, la solución propuesta tiene el potencial de ser replicable en otros comités de ética dentro del país o la región, permitiendo así escalar su impacto y contribuir a la transformación digital de la evaluación ética en contextos hospitalarios similares.

3. Trabajos Futuros

Aunque la plataforma desarrollada cumple con los objetivos establecidos y representa un avance significativo en la optimización del análisis ético de documentos de consentimiento informado, existen diversas líneas de trabajo que podrían abordarse en futuras iteraciones del proyecto.

En primer lugar, sería valioso ampliar la cobertura de la plataforma para soportar otros formatos de consentimiento informado, como los dirigidos a menores de edad o poblaciones con requerimientos especiales. Esto permitiría adaptar la herramienta a una mayor diversidad de casos evaluados por el Comité de Ética.

En segundo lugar, se propone implementar un protocolo de comunicación seguro mediante HTTPS, garantizando así una mayor protección de los datos sensibles manejados por el sistema y cumpliendo con estándares de seguridad requeridos en entornos clínicos y hospitalarios.

Adicionalmente, se propone implementar un proceso de creación de cuentas para usuarios de tipo Evaluador, el cual estará sujeto a validaciones rigurosas. Estas se realizarán primordialmente mediante la verificación de certificados oficiales emitidos

por el HUSI (Hospital Universitario San Ignacio). Como alternativa, se contemplará la presentación de otros documentos oficiales o acreditaciones profesionales expedidas por entidades competentes y reconocidas (tales como otras instituciones de salud, universidades o colegios profesionales) que permitan constatar de manera fehaciente la idoneidad y afiliación del solicitante como evaluador.

Otra línea de trabajo relevante consiste en el desarrollo y entrenamiento de un modelo de lenguaje propio (LLM) ajustado al dominio bioético y al contexto institucional. Un modelo personalizado podría mejorar aún más la precisión de las evaluaciones, optimizar el rendimiento en español y reducir la dependencia de servicios externos.

Finalmente, se considera como una evolución natural de la plataforma su expansión para cubrir todo el proceso de evaluación de proyectos de investigación, no solo la revisión del documento de consentimiento informado. Esto incluiría funcionalidades para la gestión de protocolos completos, seguimiento de observaciones, validación de anexos y coordinación entre investigadores y evaluadores dentro de un entorno centralizado.





















REFERENCIAS

-
- [1] J. Yuan, R. Tang, X. Jiang, y X. Hu, «Large Language Models for Healthcare Data Augmentation: An Example on Patient-Trial Matching», 2023, arXiv. doi: 10.48550/ARXIV.2303.16756.
 - [2] K. Sridharan y G. Sivaramakrishnan, «Assessing the Decision-Making Capabilities of Artificial Intelligence Platforms as Institutional Review Board Members», J. Empir. Res. Hum. Res. Ethics, vol. 19, n.o 3, pp. 83-91, jul. 2024, doi: 10.1177/15562646241263200.
 - [3] K. Sridharan y G. Sivaramakrishnan, «Leveraging artificial intelligence to detect ethical concerns in medical research: a case study», J. Med. Ethics, p. jme-2023-109767, 2024, doi: 10.1136/jme-2023-109767.
 - [4] G. Singh, A. Mishra, C. Pattanayak, A. Priyadarshini, y R. Das, «Artificial intelligence and the Institutional Ethics Committee: A balanced insight into pros and cons, challenges, and future directions in ethical review of clinical research», J. Integr. Med. Res., vol. 1, n.o 4, p. 164, 2023, doi: 10.4103/jimr.jimr_30_23.
 - [5] S. Harrer, «Attention is not all you need: the complicated case of ethically using large language models in healthcare and medicine», eBioMedicine, vol. 90, p. 104512, abr. 2023, doi: 10.1016/j.ebiom.2023.104512.
 - [6] M. Balas et al., «Exploring the potential utility of AI large language models for medical ethics: an expert panel evaluation of GPT-4», J. Med. Ethics, vol. 50, n.o 2, pp. 90-96, feb. 2024, doi: 10.1136/jme-2023-109549.
 - [7] P. Yu, H. Xu, X. Hu, y C. Deng, «Leveraging Generative AI and Large Language Models: A Comprehensive Roadmap for Healthcare Integration», Healthcare, vol. 11, n.o 20, p. 2776, oct. 2023, doi: 10.3390/healthcare11202776.

- [8] R. Watkins, «Guidance for researchers and peer-reviewers on the ethical use of Large Language Models (LLMs) in scientific research workflows», *AI Ethics*, may 2023, doi: 10.1007/s43681-023-00294-5.
- [9] M. Sallam, «The Utility of ChatGPT as an Example of Large Language Models in Healthcare Education, Research and Practice: Systematic Review on the Future Perspectives and Potential Limitations», 21 de febrero de 2023, Cold Spring Harbor Laboratory. doi: 10.1101/2023.02.19.23286155.
- [10] M. Pflanzner, Z. Traylor, J. B. Lyons, V. Dubljević, y C. S. Nam, «Ethics in human–AI teaming: principles and perspectives», *AI Ethics*, vol. 3, n.o 3, pp. 917-935, ago. 2023, doi: 10.1007/s43681-022-00214-z.
- [11] X. Meng et al., «The application of large language models in medicine: A scoping review», *iScience*, vol. 27, n.o 5, p. 109713, may 2024, doi: 10.1016/j.isci.2024.109713.
- [12] F. S. S. Lopes, C. Trindade, T. Carvalho, M. S. Almeida, y A. S. Carvalho, «Developing Project-Specific Informed Consent Forms: A Multi-Step Approach within the REPO4EU Framework», *REPO4EU*, may 2024. doi: 10.58647/rexpo.24000050.v1.
- [13] C. Grady, «Institutional Review Boards: Purpose and Challenges», *Chest*, vol. 148, n.o 5, pp. 1148-1155, nov. 2015, doi: 10.1378/chest.15-0706.
- [14] C. González-Gonzalo et al., «Trustworthy AI: Closing the gap between development and integration of AI systems in ophthalmic practice», *Prog. Retin. Eye Res.*, vol. 90, p. 101034, sep. 2022, doi: 10.1016/j.preteyeres.2021.101034.
- [15] J.-L. Ghim y S. Ahn, «Transforming clinical trials: the emerging roles of large language models», *Transl. Clin. Pharmacol.*, vol. 31, n.o 3, p. 131, 2023, doi: 10.12793/tcp.2023.31.e16.
- [16] P. Friesen et al., «Governing AI-Driven Health Research: Are IRBs Up to the Task?», *Ethics Hum. Res.*, vol. 43, n.o 2, pp. 35-42, mar. 2021, doi: 10.1002/eahr.500085.
- [17] A. Ferrario y N. Biller-Andorno, «Large language models in medical ethics: useful but not expert», *J. Med. Ethics*, vol. 50, n.o 9, pp. 653-654, sep. 2024, doi: 10.1136/jme-2023-109770.
- [18] A. Fedyk, J. Hodson, N. Khimich, y T. Fedyk, «Is artificial intelligence improving the audit process?», *Rev. Account. Stud.*, vol. 27, n.o 3, pp. 938-985, sep. 2022, doi: 10.1007/s11142-022-09697-x.
- [19] T. Davenport y R. Kalakota, «The potential for artificial intelligence in healthcare», *Future Healthc. J.*, vol. 6, n.o 2, pp. 94-98, jun. 2019, doi: 10.7861/futurehosp.6-2-94.
- [20] S. Bouhouita-Guermech, P. Gogognon, y J.-C. Bélisle-Pipon, «Specific challenges posed by artificial intelligence in research ethics», *Front. Artif. Intell.*, vol. 6, p. 1149082, jul. 2023, doi: 10.3389/frai.2023.1149082.
- [21] A. S. Ahuja, «The impact of artificial intelligence in medicine on the future role of the physician», *PeerJ*, vol. 7, p. e7702, oct. 2019, doi: 10.7717/peerj.7702.
- [22] M. Abu-Jeyyab, S. Alrosan, y I. Alkhawaldeh, «Harnessing Large Language Models in Medical Research and Scientific Writing: A Closer Look to The Future: LLMs in Medical Research and Scientific Writing», *High Yield Med. Rev.*, vol. 1, n.o 2, dic. 2023, doi: 10.59707/hymrfbya5348.
- [23] Node.js — Run JavaScript Everywhere. Recuperado de: <https://nodejs.org/>

-
- [24] TypeScript: Typed JavaScript at Any Scale. Recuperado de: <https://www.typescriptlang.org/>
- [25] Python Software Foundation. Python. Recuperado de: <https://www.python.org/>
- [26] Angular. One framework. Mobile & desktop. Recuperado de: <https://angular.io/>
- [27] React. A JavaScript library for building user interfaces. Recuperado de: <https://reactjs.org/>
- [28] Express.js. Express - Node.js web application framework. Recuperado de: <https://expressjs.com/>
- [29] Jest. Delightful JavaScript Testing. Recuperado de: <https://jestjs.io/>
- [30] MongoDB: The application data platform. Recuperado de: <https://www.mongodb.com/>
- [31] Git. Recuperado de: <https://git-scm.com/>
- [32] GitHub: Where the world builds software. Recuperado de: <https://github.com/>
- [33] Visual Studio Code. Code editing. Redefined. Recuperado de: <https://code.visualstudio.com/>
- [34] Visual Paradigm - UML, Agile, PMBOK, TOGAF, BPMN and More. Recuperado de: <https://www.visual-paradigm.com/>
- [35] Microsoft Teams. Recuperado de: <https://www.microsoft.com/es-co/microsoft-teams/group-chat-software>
- [36] Microsoft Outlook. Recuperado de: <https://outlook.live.com/>
- [37] Discord | Your Place to Talk and Hang Out. Recuperado de: <https://discord.com/>
- [38] WhatsApp. Recuperado de: <https://www.whatsapp.com/>
- [39] Google Docs: Free Online Documents for Personal Use. Recuperado de: <https://www.google.com/docs/about/>
- [40] ClickUp. One app to replace them all. Recuperado de: <https://clickup.com/>
- [41] T. Mikolov, K. Chen, G. Corrado, y J. Dean, «Efficient Estimation of Word Representations in Vector Space», arXiv preprint arXiv:1301.3781, 2013.
- [42] Q. Le y T. Mikolov, «Distributed Representations of Sentences and Documents», Proceedings of the 31st International Conference on Machine Learning, pp. 1188–1196, 2014.
- [43] J. E. F. Friedl, «Mastering Regular Expressions» (3rd ed.), O'Reilly Media, 2006.
- [44] D. Jurafsky y J. H. Martin, «Speech and Language Processing» (2nd ed.), Prentice Hall, 2009.
- [45] Google Cloud. Document AI. Recuperado de: <https://cloud.google.com/document-ai>
- [46] Amazon Web Services. Amazon Textract. Recuperado de: <https://aws.amazon.com/textract/>
- [47] IBM. Watson Discovery. Recuperado de: <https://www.ibm.com/cloud/watson-discovery>
- [48] I. Goodfellow, Y. Bengio y A. Courville, «Deep Learning», MIT Press, 2016.
- [49] F. Chollet, «Deep Learning with Python», Manning Publications, 2018.
- [50] Artillery: Modern load testing and functional testing. Recuperado de: <https://artillery.io/docs/>
- [51] Microsoft. Playwright: Fast and reliable end-to-end testing for modern web apps. Recuperado de: <https://playwright.dev/docs/intro>

ANEXOS

1. Diagrama de Actividad Crear Caso
 DiagramaActividad_CrearCaso_Investigador.png
2. Diagrama Actividad Previsualizar Eliminar Casos
 DiagramaActividad_PrevisualizarEliminar_Investigador.png
3. Diagrama Actividad Crear Evaluación
 DiagramaActividad_CrearEvaluacion_Evaluador.png
4. Diagrama Actividad Gestión de Evaluaciones
 DiagramaActividad_GestionEvaluaciones_Evaluador.png
5. Diagrama Actividad Envío de Correo
 DiagramaActividad_ResultadoCorreo_Evaluador.png
6. Diagrama de Actividad LLM
 DiagramaActividad_LLM_Evaluador.png
7. Diagrama Actividad Estadísticas
 DiagramaActividad_Estadisticas_Evaluador.png
8. Documento Propuesta de Proyecto
 Propuesta Proyecto.docx
9. Documento Plan de Administración de Proyecto
 1_planproyecto_PLANADMINISTRACIONPROYECTO_V2FINAL.docx
10. Documento Especificación de Requerimientos de Software
 2_especificacion_requerimientos_SRS_V2FINAL
11. Documento Descripción del Diseño de Software
 3_disenosolucion_SDD
12. Documento de Pruebas
 Documento de Pruebas
13. Documento de requerimientos
 Requerimientos-finales.xlsx
14. Plantilla de Consentimientos Informados
 PLANTILLA_CONSENTIMIENTO.pdf
15. Lista de Chequeo de Contenidos de protocolos de investigación
 ListaChequeo.pdf
16. Diagrama de arquitectura de alto nivel
 ArquitecturaAltoNivel.jpg
17. Diagrama de componentes
 Componentes.png
18. Diagrama de despliegue
 Despliegue.png
19. Carpeta con imágenes de interfaces
 Interfaces
20. Documento Benchmark
 Extra_BenchMark IAs