

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



Facultad de ingeniería

SPMP

Proyecto de grado

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

Ingeniería de sistemas

Bogotá D.C.
Octubre 2024

Prefacio

El desarrollo de la investigación científica se encuentra estrechamente ligado a la integridad ética con la que se llevan a cabo los proyectos. Los comités de ética juegan un papel crucial al garantizar que cada investigación cumpla con los estándares morales y regulatorios necesarios para proteger a los participantes, y a su vez asegurar la validez de los resultados. Sin embargo, el volumen de investigaciones y la complejidad del proceso de evaluación de las mismas, han superado la capacidad tradicional de estos comités, que deben enfrentarse a procesos tediosos y en gran medida manuales para revisar cada proyecto.

El contenido de este documento presenta el desarrollo de una plataforma de soporte para la verificación de criterios éticos en los consentimientos informados de proyectos de investigación, implementada para el Hospital Universitario San Ignacio. El propósito de este proyecto es agilizar esta etapa específica del proceso de revisión ética de los proyectos de investigación, mediante el uso de inteligencia artificial y procesamiento de lenguaje natural, abordando uno de los principales desafíos que enfrentan los comités: la creciente carga operativa y la complejidad en la evaluación de los proyectos.

A través de esta plataforma, se pretende facilitar las evaluaciones, reduciendo los tiempos de respuesta y minimizando los errores humanos. El alcance del proyecto incluye tanto el desarrollo de una plataforma, como la integración de un modelo avanzado de inteligencia artificial ya existente, con la finalidad de que asista al comité en su proceso de evaluación (más específicamente al evaluador), ayudando a garantizar el cumplimiento de las normativas éticas y regulatorias vigentes.

Adicionalmente, de forma complementaria, se desarrollará una parte de la plataforma para asistir al investigador en la creación y redacción del consentimiento informado. Esto permitirá que disminuyan los errores por falta de completitud en los documentos, y que, como consecuencia, se reduzcan también así los tiempos de evaluación, ya que el evaluador no tendría que enfocarse en esto, sino únicamente en el contenido de ellos.

Este documento está dirigido a comités de ética, investigadores, y gestores de proyectos de investigación en el ámbito de la salud que deseen mejorar la eficiencia de sus procesos de evaluación ética. Asimismo, es de interés para académicos y profesionales del área de la inteligencia artificial aplicada, quienes podrán encontrar en este proyecto una propuesta de solución para la automatización de procesos institucionales.

Confiamos en que la plataforma propuesta contribuirá a fortalecer la calidad de los proyectos de investigación, no solo en el Hospital Universitario San Ignacio, sino también como una referencia para otras instituciones interesadas en implementar herramientas tecnológicas avanzadas.

Tabla de contenidos

1. Vista General del Proyecto.....	7
1.1. Visión del producto.....	7
1.2. Propósito, alcance y objetivos.....	7
■ Propósito.....	7
■ Alcance.....	13
■ Objetivos.....	14
1.3. Supuestos y Restricciones.....	14
■ Supuestos.....	14
■ Restricciones.....	15
1.4. Entregables.....	15
1.5. Resumen de Calendarización y presupuesto.....	16
1.6. Evolución del plan.....	17
1.7. Glosario.....	19
2. Contexto del proyecto.....	20
2.1. Modelo de ciclo de vida.....	20
■ Análisis de alternativas y Justificación.....	28
2.2. Lenguajes y herramientas.....	30
■ Análisis de Alternativas y justificación.....	32
2.3. Plan de aceptación de producto.....	34
1. Levantamiento y Validación de Requerimientos (Sprint 1 y 2).....	35
2.4. Organización del proyecto y comunicación.....	37
■ Interfaces Externas.....	37
■ Organigrama y descripción de roles.....	37
3. Administración del proyecto.....	39
3.1. Métodos y herramientas de estimación.....	39
3.2. Inicio del proyecto.....	41
■ Entrenamiento del personal.....	41
■ Infraestructura.....	42
3.3. Planes de trabajo del proyecto.....	43
■ Descomposición de actividades.....	43
■ Calendarización.....	45
4. Monitoreo y control del proyecto.....	46
4.1. Administración de requerimientos.....	46
4.2. Monitoreo y control del progreso.....	48
4.3. Cierre del proyecto.....	50
5. Entrega del Producto.....	51
6. Procesos de soporte.....	51
6.1. Ambiente de trabajo.....	51
6.2. Análisis y Administración de Riesgos.....	52
6.3. Administración de configuración y Documentación.....	55
6.4. Métricas y proceso de Medición.....	57
6.5. Control de calidad.....	59

7. Anexos..... 61

8. Referencias..... 62

Lista de figuras

Figura 1 - BPMN Ciclo de vida

Figura 2 - BPMN Levantamiento de requerimientos

Figura 3 - BPMN Desarrollo Iterativo

Figura 4 - BPMN Integración y Ajuste del Modelo de IA

Figura 5 - BPMN Pruebas funcionales y preparación de entrega

Figura 6 - BPMN Cierre del Proyecto

Figura 7 - WBS Descomposición de actividades

Figura 8 - BPMN Scrum de tareas iterativas

Figura 9 - BPMN Sprints

Figura 10 - BPMN Administración de Requerimientos

Figura 11 - BPMN Administración de Riesgos

Figura 12 - BPMN Proceso de control de cambios

Lista de tablas

Tabla 1. Formato que los evaluadores

Tabla 2. Etapas del proyecto

Tabla 3. Glosario

Tabla 4. Interfaces Externas

Tabla 5. Riesgos identificados

Tabla 6. Riesgos más importantes

Tabla 7. Artefactos

1. Vista General del Proyecto

1.1. Visión del producto

Se propone el desarrollo de una plataforma de software que asista al Comité de Ética del Hospital Universitario San Ignacio en la revisión de los documentos de consentimiento informado de proyectos de investigación. Esta herramienta utilizará un modelo de lenguaje de gran escala (LLM) existente para analizar de manera eficiente y precisa dichos documentos, identificando el cumplimiento de las normativas éticas internacionales vigentes y entregando como respuesta una retroalimentación específica para cada norma. [1][2][20]

La plataforma también contará con una sección enfocada en los investigadores, ayudándolos en la generación del documento de consentimiento informado, a través de un formulario el cual está basado en una plantilla preestablecida por el hospital. El objetivo de esta sección es reducir los errores por falta de completitud, que actualmente corresponden a una gran parte de los errores encontrados en los consentimientos, y por ende, se agilicen los tiempos de evaluación.

De esta manera, la visión del producto que se propone es transformar el proceso de evaluación ética, actualmente laborioso y propenso a errores, en una experiencia más ágil y confiable. Esto permitirá a los evaluadores enfocarse en decisiones más complejas que requieren juicio y experiencia humana, potenciando sus capacidades sin reemplazarlos.

Además, la plataforma contará con una interfaz intuitiva que presentará claramente las áreas que cumplen o no con las normativas, señalando dónde se requiere atención adicional. Con esta implementación, se espera revolucionar la manera en que el comité de ética gestiona la evaluación de consentimientos informados, estableciendo un nuevo estándar en la gestión ética de la investigación. [12]

En última instancia, esta herramienta no solo mejorará el proceso interno del comité, sino que también tendrá un impacto positivo en el avance de la investigación ética a nivel institucional. Al establecer un modelo eficiente y éticamente sólido, la plataforma podría servir de referencia para otros comités de ética, promoviendo prácticas de evaluación más efectivas y consistentes en el ámbito de la investigación científica.

1.2. Propósito, alcance y objetivos

■ Propósito

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 más alta para sus miembros, en especial en la gestión y revisión de dichos proyectos.

Este aumento en la demanda ha derivado en significativas demoras en la aprobación de estudios no solo en el HUSI, sino también en muchos otros centros de investigación en Colombia. Dicha situación afecta de manera negativa el avance de los proyectos y, en consecuencia, dificulta el cumplimiento de los cronogramas establecidos por los investigadores, reduciendo la capacidad de los equipos de investigación para progresar de manera eficiente y oportuna. [13]

El comité de ética del HUSI recibe más de 200 solicitudes de revisión al año, y cada solicitud requiere una revisión ética detallada y rigurosa. Este proceso se realiza manualmente, lo que lo convierte en una tarea altamente laboriosa y propensa a errores humanos.

De esta manera, la falta de un sistema ágil para asistir la verificación de los documentos implica que la evaluación de cada proyecto puede tomar más tiempo del necesario, afectando tanto la eficiencia como la calidad del proceso.

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 los consentimientos siguen una estructura fija y están basados en normativas específicas y plantillas que facilitan la evaluación, el proceso sigue siendo altamente demandante desde el punto de vista administrativo. [12]

Un ejemplo del formato que los evaluadores deben llenar se muestra a continuación:

Característica		Si cumple	No cumple	No aplica
1	Contiene versión y fecha del FCI			
2	Contiene título del proyecto e instituciones participantes			
3	Describe la justificación y los objetivos de la investigación. Problema, pertinencia del estudio, objetivo general			
4	Describe los procedimientos del estudio y su propósito, incluyendo la identificación de aquellos que son experimentales (aleatorización, cegamiento, uso de placebo, etc.).			
5	Determina las molestias o los riesgos esperados. Si no existen informar.			
6	Determina los beneficios que puedan obtenerse. Si no existen informar.			

7	<p>Indica la seguridad que no se identificará al sujeto y que se mantendrá la confidencialidad de la información relacionada con su privacidad. Se acoge a la Ley 1581 de 2012 (Habeas Data) e indica cómo se manejarán los datos:</p> <ul style="list-style-type: none"> • Dónde se almacenarán los datos. • Mecanismos de custodia y seguridad de los mismos. • Tiempo de custodia. • Quiénes tendrán acceso a la información y bajo qué parámetros de seguridad se accederá a ellos. • Tratamiento de datos: codificación o anonimización tanto para los análisis como para la publicación de los resultados. • Posibilidad de conocer sus datos personales registrados en la base de datos del estudio, solicitar rectificación de los mismos y de retirar su consentimiento para el tratamiento de los datos en cualquier momento del estudio, excepto a partir de la anonimización. • Describe los procesos de transferencia de datos a terceros, en caso de estudios colaborativos, y la garantía de mantener la privacidad, confidencialidad y seguridad en el tratamiento por parte del tercero. 			
8	Indica los procedimientos alternativos que pudieran ser ventajosos para el sujeto.			
9	<p>Expresa el compromiso de proporcionar información actualizada obtenida durante el estudio, aunque ésta pudiera afectar la voluntad del sujeto para continuar participando.</p> <p>En caso de realización de estudios que requieran entrega de resultados de procedimiento o consejería (genética, por ejemplo), se debe explicar el proceso.</p>			
10	<p>Expresa la garantía de que, si existen gastos adicionales, éstos serán cubiertos por el presupuesto de la investigación o de la institución responsable de la misma.</p> <p>Enunciar costos generados por participar en el estudio, ej.: transporte, alimentación etc. y la forma cómo serán reembolsados (bonos de alimentación o transporte, dinero).</p>			
11	Indica tiempo de permanencia en el estudio: Tiempo de participación, frecuencia de las intervenciones si aplica.			
12	Indica la libertad de retirar su consentimiento en cualquier momento y dejar de participar en el estudio sin que por ello se creen perjuicios para continuar su cuidado y tratamiento.			

Tabla 1. Formato que los evaluadores

Como se puede observar, la tarea de verificación de estos consentimientos se basa en que los evaluadores completen formatos en los que se clasifique cada criterio como "cumple", "no cumple" o "no aplica" para cada normativa aplicable. Este proceso requiere una revisión detallada y un análisis exhaustivo, ya que se deben buscar y validar todos los elementos normativos relevantes según el tipo de estudio.

Adicional a esto, el formato de documento que se tiene a evaluar es el siguiente:

INTRODUCCIÓN

En verde encontrará instrucciones que deben ser eliminadas.

En rojo encontrará los campos que debe reemplazar con la información correspondiente con su proyecto.

Código del sujeto: _____

Este documento de consentimiento es para uso en un proyecto de investigación que involucrará a sujetos que quizá tengan o no la capacidad de dar consentimiento para su participación. En este documento de consentimiento, “usted” hace referencia al participante de la investigación. Si usted es un representante legal, recuerde que “usted” hace referencia al participante de investigación.

Usted está siendo invitado a participar en el proyecto de investigación “nombre del proyecto”. Este documento le proporciona la información necesaria para que usted participe voluntaria y libremente. El proyecto es realizado por nombre de la(s) institución(es) participante(s). Antes de dar su consentimiento, usted necesita entender plenamente el propósito de su decisión. Este proceso se denomina consentimiento informado. Una vez que haya leído este documento y resuelto con el investigador las dudas, se le pedirá que firme este formato en señal de aceptación de participar.

INFORMACIÓN GENERAL

1. ¿Por qué se debe realizar este estudio? Describa brevemente el problema y la pertinencia del estudio.
2. ¿Cuál es el objetivo de este estudio? Enuncie el objetivo general.
3. ¿En qué consiste el estudio? Describa de manera clara y sencilla los procedimientos, intervenciones y su propósito, incluyendo los experimentales (aleatorización, cegamiento, uso de placebo, etc.).
4. ¿Cuáles son las molestias o los riesgos esperados? Describir molestias y riesgos. En caso de que no existan molestias o riesgos igualmente se debe informar.
5. ¿Cuáles son los beneficios que puedo obtener por participar? Enunciar beneficios. En caso de que no existan beneficios igualmente se debe informar.
6. ¿Existe confidencialidad en el manejo de mis datos? Este proyecto se acoge a la ley 1581 de 2012 (Hábeas Data) que aplica para el tratamiento de datos personales.

Indique brevemente cómo se manejarán los datos:

- Describa en dónde se almacenarán los datos e información, los mecanismos de custodia y seguridad de los mismos y el tiempo de custodia.

- Describa quiénes tendrán acceso a la información y bajo qué parámetros de seguridad se accederá a ellos

- Describa cómo se llevará a cabo la anonimización de los datos tanto para los análisis como para la publicación de los resultados.

- Describa la posibilidad de conocer los datos personales registrados en la base de datos del estudio, solicitar rectificación de los mismos y de retirar su consentimiento para el tratamiento de los datos en cualquier momento del estudio, excepto a partir de la anonimización. - Describa los procesos de transferencia de datos a terceros, en

caso de estudios colaborativos, y la garantía de mantener la privacidad, confidencialidad y seguridad en el tratamiento por parte del tercero.

7. ¿Existen procedimientos alternativos que pudieran ser ventajosos para mí?

En caso de que se realicen intervenciones con dispositivos, procedimiento médico-quirúrgico o medicamentos explicar si existen otras intervenciones que puedan realizarse para la patología del paciente.

8. Expresar el compromiso de proporcionar información actualizada obtenida durante el estudio, aunque ésta pudiera afectar la voluntad del sujeto para continuar participando.

En caso de realización de estudios que requieran entrega de resultados de procedimiento o consejería (genética, por ejemplo), explicar el proceso.

9. ¿Existe alguna obligación financiera? Participar en este estudio no tiene ningún costo económico para usted. En caso contrario, enunciar costos generados por participar en el estudio, describir cuáles: transporte, alimentación etc. y la forma cómo serán asumidos con cargo al presupuesto del proyecto.
10. ¿Cuánto tiempo durará mi participación en el estudio? Indique el tiempo de participación y en caso de ser necesario la frecuencia de las intervenciones.
11. ¿Qué sucede si no deseo participar o me retiro del estudio? Usted puede decidir no participar o retirarse en cualquier momento del estudio, sin que esto afecte de manera alguna el tratamiento médico que necesita.

En estudios en los que se involucran menores de edad o discapacitados físicos y mentales adultos, que propongan intervenciones o procedimientos que superan el riesgo mínimo (no hacen parte del estándar de manejo), se debe evaluar la capacidad de entendimiento de acuerdo a la Res. 8430 de 1993. (Mantener o retirar el numeral 12 de acuerdo a la naturaleza del estudio)

12. Al menor de edad o la persona que usted representa se le solicitará informar su aceptación para ingresar al estudio a través de un Asentimiento Informado. De acuerdo a la Resolución 8430 de 1993, el proceso de asentimiento debe estar precedido de una valoración de razonamiento entendimiento y lógica realizada por un psicólogo, neurólogo o psiquiatra. A través de este documento se solicita su autorización para realizar la valoración.

Tabla 2. Formato Consentimiento Informado

En caso de estudios con riesgo mayor al mínimo (mantener o retirar el numeral 13 de acuerdo a la naturaleza del estudio):

Como se puede observar en el formato, si bien gran parte se deja de manera predeterminada, hay muchos datos que el investigador tiene que completar manualmente. Estos datos muchas veces no son llenados por parte del investigador, lo que genera retrasos adicionales en el proceso de evaluación, ya que el evaluador debe devolver el documento para que los campos sean llenados, y luego revisarlo nuevamente. Actualmente, en el comité, una gran parte de los documentos son devueltos por esta razón, lo cual ha impactado directamente en el retraso de la evaluación. Esto se debe a que, además de tratarse de documentos extensos para evaluar, el hecho de tener que realizar el trabajo dos veces incrementa los retrasos.

Debido a esto, se resalta la necesidad de agilizar el proceso actual de verificación manual, que es un proceso lento y propenso a errores, al implementar la verificación automática de los datos ingresados, lo cual también incrementa la probabilidad de errores e inconsistencias con una lesión directamente causada por los medicamentos administrados/los procedimientos realizados, en esta investigación al centro hospitalario del estudio, y el centro del estudio le proporcionará el tratamiento necesario, situación que limita el desarrollo de investigaciones médicas de calidad en el tiempo necesario y afecta la capacidad de respuesta del comité frente a las solicitudes.

De esta manera, se propone como propósito del proyecto implementar soluciones tecnológicas que permitan asistir gran parte del proceso de revisión ética,

específicamente en lo relacionado con la verificación de los consentimientos informados, sin comprometer la rigurosidad y calidad de las evaluaciones. Esto no solo permitiría reducir los tiempos del proceso, sino que también contribuiría a minimizar errores, mejorar la consistencia en las revisiones y liberar tiempo para que los miembros del comité puedan concentrarse en tareas de mayor complejidad.

■ Alcance

El proyecto se enfocará en el desarrollo de una plataforma de software para el Hospital Universitario San Ignacio, destinada a asistir al Comité de Ética e investigadores en el proceso de evaluación y formulación de consentimientos informados, para proyectos de investigación que involucren a seres humanos.

El alcance específico del proyecto incluye:

- Utilización de un modelo de inteligencia artificial (IA) existente: Se ajustará un modelo de lenguaje grande (LLM) ya disponible mediante técnicas de *prompt engineering* para analizar documentos de consentimiento informado, identificando si cumplen o no cumplen las normativas éticas internacionales aplicables. [7][10][14]
- Integración del modelo de IA en una plataforma web: Se creará una interfaz de usuario intuitiva que permitirá a los evaluadores cargar documentos, visualizar los resultados del análisis y recibir retroalimentación específica sobre el cumplimiento normativo. [3]
- Asistencia en la verificación de criterios éticos: La plataforma ayudará en la identificación de omisiones en los documentos, facilitando la revisión y reduciendo el tiempo dedicado al análisis manual. [4][5]
- Soporte a los investigadores en la preparación de los consentimientos informados: La plataforma guiará a los investigadores en la elaboración de los documentos, asegurándose de que no falte ningún requisito antes de enviarlos a evaluación. De esta manera, se evitarán devoluciones innecesarias y se acelerará el proceso de aprobación.

Adicionalmente, el proyecto no abarca:

- Limitaciones de evaluación: En caso de que la opción a seleccionar para evaluar una norma sea “No aplica”, el evaluador será quien debe establecer esto, el LLM no lo hará. [8] [17]
- La evaluación completa de todos los aspectos éticos de un proyecto de investigación: La plataforma se centrará exclusivamente en el análisis de los documentos de consentimiento informado y no evaluará otros componentes del proyecto.
- Reemplazo del evaluador humano: El sistema está diseñado para asistir y complementar la labor del evaluador, no para sustituir su criterio y experiencia. Las decisiones que requieran juicio humano seguirán siendo responsabilidad del evaluador.
- Integración con sistemas existentes del hospital: No se contempla la integración con otros sistemas o bases de datos del hospital más allá de lo necesario para el funcionamiento autónomo de la plataforma.
- Soporte multilingüe o adaptación a normativas de otros países: El proyecto se enfocará en las normativas éticas internacionales aplicables al contexto del Hospital Universitario San Ignacio y no considerará adaptaciones a otros idiomas o regulaciones extranjeras.

■ Objetivos

Objetivo General

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

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 usado en proyectos de investigación presentados al Comité de Ética del Hospital Universitario San Ignacio.
- 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.

1.3. Supuestos y Restricciones

■ Supuestos

La plataforma está diseñada para ser utilizada principalmente por personal administrativo y miembros del comité de ética. Se presume que, si bien poseen conocimientos básicos de tecnología, no cuentan con una formación técnica avanzada. Se anticipa que tanto los investigadores como los miembros del comité tendrán acceso completo y adecuado a los datos de los proyectos de investigación dentro de la plataforma, lo que facilitará su evaluación. Para su operación, se requiere que los usuarios dispongan de la infraestructura tecnológica mínima necesaria, como computadoras actualizadas y acceso a internet estable.

Partiendo de la premisa de que tanto el comité de ética como los investigadores estarán dispuestos a adoptar un sistema digitalizado, se confía en que la automatización mejorará la eficiencia de los procesos. Asimismo, se espera que el uso de inteligencia artificial (IA) sea comprendido y aceptado por los usuarios sin generar desconfianza.

Otro supuesto clave es que el personal recibirá una capacitación adecuada para utilizar el sistema de manera eficiente. Finalmente, se espera una acogida positiva por parte de los usuarios, quienes apreciarán las mejoras en eficiencia y la reducción de los tiempos en los procesos de evaluación ética.

■ Restricciones

La plataforma deberá cumplir estrictamente con las normativas éticas y regulatorias tanto a nivel nacional como internacional, incluyendo leyes de protección de datos y privacidad en el sector de la salud, garantizando la confidencialidad e integridad de los datos procesados. Dada la importancia de los procesos que gestionará, la plataforma también deberá asegurar una alta disponibilidad.

Otra restricción clave es que el sistema también debe ser capaz de escalar eficientemente a medida que aumente el volumen de proyectos de investigación, conforme crezca su adopción. Además, la implementación de inteligencia artificial deberá cumplir con estándares técnicos que garanticen precisión en la detección de incumplimientos éticos. Todos los cambios y decisiones durante el desarrollo deberán seguir un enfoque iterativo, bajo la metodología Scrum, respetando los plazos y recursos disponibles del proyecto.

1.4. Entregables

El desarrollo del proyecto se gestionará bajo el marco de trabajo Scrum, el cual promueve la entrega incremental de valor mediante iteraciones cortas y controladas, denominadas sprints. Este enfoque permitirá gestionar la complejidad del proyecto a través de la colaboración continua con los actores involucrados, la validación temprana de los avances y la adaptación constante a los requerimientos del comité de ética.

El proyecto se dividirá en las siguientes etapas, cada una con entregables específicos que se revisarán y aprobarán al finalizar cada sprint:

- *Levantamiento y validación de requerimientos*

En esta etapa se realizará el trabajo conjunto con el comité de ética para identificar las necesidades, los desafíos actuales y las funcionalidades esperadas. El principal entregable serán los mockups validados por el cliente, los cuales servirán como referencia para el desarrollo posterior.

- *Diseño y definición funcional del sistema*

A partir de los requerimientos levantados, se definirán las funcionalidades clave, las reglas del negocio y los flujos de interacción del usuario. El entregable de esta etapa será el documento de definición de requerimientos, junto con el product backlog del proyecto.

- *Desarrollo iterativo de la solución*

En esta fase se realizarán varios sprints orientados a la construcción de las funcionalidades priorizadas, incluyendo la integración del modelo de inteligencia artificial (IA) para el análisis de documentos. En cada sprint se entregarán versiones incrementales del sistema, que incluirán las mejoras acordadas en las reuniones de revisión, con el director.

- *Pruebas y validación de la plataforma*

Se llevarán a cabo pruebas funcionales, técnicas y de usabilidad con los usuarios clave para validar el correcto funcionamiento del sistema y su alineación con los objetivos del comité de ética. El entregable será un informe de validación con los resultados y las recomendaciones para los ajustes finales.

- *Entrega final y despliegue (dockerizado)*

La última etapa contempla la entrega de la plataforma web completamente operativa, con el modelo de inteligencia artificial integrado y validado. Es importante aclarar que esta entrega no corresponde a un despliegue en un servidor productivo, sino a la entrega del sistema empaquetado en contenedores, de manera que los médicos del hospital y los miembros del comité de ética puedan realizar el despliegue cuando lo consideren necesario. Esta modalidad de entrega responde a un requerimiento específico del comité de ética, quienes solicitaron recibir el sistema en este formato.

El entregable incluirá, además del sistema en contenedores, la documentación técnica y funcional necesaria para su correcta implementación y mantenimiento, así como los documentos teóricos y de contextualización que respaldan el desarrollo del proyecto de grado.

De esta manera, el marco Scrum permitirá mantener un ciclo de mejora continua, garantizando que los resultados obtenidos respondan efectivamente a las necesidades planteadas por el comité de ética y demás partes interesadas.

1.5. Resumen de Calendarización y presupuesto

El desarrollo del proyecto se llevará a cabo en un periodo estimado de cuatro meses, utilizando el marco de trabajo ágil Scrum, con sprints quincenales que permitirán entregar avances incrementales y realizar revisiones periódicas con los actores clave. En cada sprint se priorizará la entrega de valor y se validarán los resultados con los integrantes del equipo y el Comité de Ética del Hospital Universitario San Ignacio.

Es importante aclarar que este proyecto no cuenta con un presupuesto asignado, por lo tanto, no se incluyen aspectos financieros en esta planificación.

Etapas	Descripción	Duración estimada
Levantamiento y validación de requerimientos	Se trabajará de manera colaborativa con el Comité de Ética para identificar los requisitos funcionales y no funcionales, así como los flujos de uso esperados. Se recopilarán documentos reales que servirán de referencia para el ajuste del modelo de inteligencia artificial.	4 semanas
Desarrollo iterativo de la plataforma	A través de sprints quincenales, se construirá progresivamente una interfaz accesible y funcional, que permita a los usuarios cargar documentos de consentimiento informado y visualizar los resultados del análisis realizado por el modelo de IA. Cada sprint	8 semanas

	incluirá entregables revisables y retroalimentación por parte del comité.	
Integración y ajuste del modelo de IA	Se integrará un modelo de lenguaje previamente existente, ajustándolo mediante técnicas de <i>prompt engineering</i> para que pueda aplicar los criterios éticos definidos y evaluar correctamente los documentos. Esta integración se desarrollará en paralelo a la construcción de la plataforma.	2 semanas
Pruebas funcionales, validación con usuarios y preparación para la entrega	Se realizarán pruebas funcionales y de usabilidad con los usuarios del Comité de Ética para asegurar que la plataforma cumple con los objetivos del proyecto. La entrega final incluirá el sistema empaquetado en contenedores Docker, permitiendo su despliegue en cualquier momento y entorno requerido por el comité.	2 semanas
Cierre del proyecto	En esta etapa se entregan todos los productos y documentación finales, se presenta la plataforma validada al Director de Tesis, al Comité de Ética y a los respectivos jurados del Proyecto de Grado para su aprobación formal, y se archivan los entregables en el repositorio institucional, asegurando la transferencia de conocimientos para su mantenimiento futuro.	2 semanas

Tabla 2. Etapas del proyecto

1.6. Evolución del plan

El Plan de Administración del Proyecto es un documento vivo que puede requerir ajustes y actualizaciones durante el ciclo de vida del proyecto. El proceso para realizar estos cambios sigue un enfoque iterativo y ágil, garantizando que los ajustes se realicen de manera controlada y con la aprobación de los stakeholders clave (que serían en nuestro caso los miembros del comité de ética, evaluadores, administrativos y demás).

Proceso General de Cambios al Plan

1. Detección de la Necesidad de Cambio

Los cambios al plan pueden originarse de diversas fuentes, como cambios en los requerimientos del Comité de Ética, nuevos descubrimientos técnicos, limitaciones de recursos, o retroalimentación durante las fases de desarrollo. En cualquier caso, la necesidad de un cambio será identificada y registrada.

2. Propuesta de Cambio

Una vez detectada la necesidad de un cambio, se deberá formular una propuesta formal que describa el ajuste necesario y su justificación. La propuesta incluirá el impacto potencial en el cronograma, los entregables y los recursos. Esta propuesta será documentada y evaluada por el equipo y se decidirá si el cambio es viable y necesario.

3. Revisión y Aprobación

La propuesta de cambio será revisada por los stakeholders relevantes, como el equipo de desarrollo y el comité de ética (en caso de ser necesario). En una reunión de revisión de sprint (Scrum), se discutirá el impacto del cambio en los objetivos del proyecto y se decidirá si se aprueba, rechaza o aplaza la implementación del cambio.

- Sección Referente: Para más detalles sobre la administración de requerimientos y cómo se manejan los cambios, consultar la sección 4.1 "Administración de Requerimientos".

4. Actualización del Plan

Una vez aprobado, el cambio se formalizará en el Plan de Administración del Proyecto. Las secciones afectadas del plan serán revisadas y actualizadas para reflejar el nuevo alcance, cronograma, entregables o asignación de recursos. Esta actualización será registrada y compartida con todos los miembros del equipo y stakeholders.

- Sección Referente: Para más información sobre cómo se gestionan los entregables y el cronograma, consultar la sección 1.5 "Resumen de Calendarización y Presupuesto".

5. Implementación del Cambio

El cambio será incorporado en el backlog del proyecto y priorizado para ser desarrollado en los próximos sprints, según su importancia y urgencia. El equipo de desarrollo implementará los cambios conforme a las directrices del plan actualizado.

- Sección Referente: La gestión de tareas y la implementación de cambios están cubiertas en detalle en la sección 3.3.2 "Calendarización".

6. Monitoreo del Cambio

Después de implementar el cambio, se monitorizará su impacto en el proyecto para asegurarse de que no haya problemas imprevistos. Cualquier ajuste adicional será tratado de acuerdo con este mismo proceso iterativo de cambios.

Sección Referente: Para detalles sobre el monitoreo de cambios y su impacto en el proyecto, consultar la sección 4.2 "Monitoreo y Control del Progreso".

1.7. Glosario

Término	Definición
IA	Inteligencia Artificial
LLM	Large Language Model, modelo de lenguaje a gran escala para procesamiento de lenguaje natural
NLP	Natural Language Processing, procesamiento de lenguaje natural
HUSI	Hospital Universitario San Ignacio
Mockup	Modelo visual o esquema del diseño de una interfaz
Sprint	Periodo de tiempo en la metodología ágil donde se completan tareas específicas
TDD	Test-Driven Development, desarrollo guiado por pruebas
API	Application Programming Interface, interfaz de programación de aplicaciones
Git	Sistema de control de versiones distribuido para gestionar cambios en el código fuente
GitHub	Plataforma de colaboración y alojamiento de repositorios Git
Node.js	Entorno de ejecución de JavaScript que permite crear aplicaciones escalables en backend
TypeScript	Superconjunto de JavaScript que añade tipado estático y otras características avanzadas
MongoDB	Base de datos NoSQL ideal para datos no estructurados o semiestructurados
Scrum	Metodología ágil de gestión de proyectos enfocada en entregas incrementales mediante sprints
JAVA	Lenguaje de programación de propósito general, orientado a objetos, utilizado para desarrollo en backend
SpringBoot	Framework de desarrollo para Java que simplifica la creación de aplicaciones web y servicios backend
Angular	Framework de desarrollo de aplicaciones web basado en JavaScript desarrollado por Google, enfocado en el desarrollo de aplicaciones de una sola página (SPA).
React	Biblioteca de JavaScript desarrollada por Facebook para construir interfaces de usuario, especialmente útiles para crear aplicaciones de una sola página.

Express.js	Framework de servidor web para Node.js que facilita la creación de aplicaciones web y APIs RESTful, proporcionando herramientas y funcionalidades para el backend.
Prompt Engineering	Técnica en Inteligencia Artificial para diseñar instrucciones (prompts) que optimicen las respuestas de modelos de lenguaje como los LLMs.
JavaScript	Lenguaje de programación de propósito general que se utiliza principalmente en desarrollo web para mejorar la interactividad y la dinámica de las páginas web.
TypeScript	Superconjunto de JavaScript que añade tipado estático y características avanzadas, lo que facilita el mantenimiento y la escalabilidad de aplicaciones grandes.
Jest	Framework de pruebas en JavaScript, popular para probar aplicaciones React y JavaScript en general, ofreciendo funcionalidades de prueba unitarias y de integración.
Frontend	Parte de una aplicación que interactúa directamente con el usuario.
HTTPS	Protocolo de Transferencia de Hipertexto
TCP/IP	Protocolo de comunicación estándar para redes
RESTful	Arquitectura que sigue el modelo REST (Representational State Transfer)
Docker	Plataforma que permite crear, desplegar y ejecutar aplicaciones en contenedores
Backend	Parte de una aplicación que se ejecuta en el servidor y se encarga del procesamiento de datos y la lógica de negocio, conectando con bases de datos y gestionando las solicitudes de los clientes.

Tabla 3. Glosario

2. Contexto del proyecto

2.1. Modelo de ciclo de vida

El propósito de esta sección es definir el modelo de ciclo de vida seleccionado para el desarrollo de la plataforma, destacando las prácticas y métodos que orientarán el trabajo del equipo. Para este proyecto, se ha optado por utilizar exclusivamente el marco de trabajo Scrum, el cual permite gestionar el desarrollo de manera ágil, incremental y adaptativa.

El ciclo de vida del proyecto se gestionará a través de sprints quincenales, en los cuales el equipo trabajará de forma iterativa para entregar funcionalidades incrementales que serán revisadas y validadas por los actores involucrados, especialmente el Comité de Ética del Hospital Universitario San Ignacio.

El desarrollo inicia con un primer sprint orientado a la comprensión del contexto y levantamiento de requisitos, donde se identificarán las necesidades, objetivos y desafíos que enfrenta el comité en el proceso de evaluación de documentos de consentimiento

informado. Esta etapa permitirá clarificar las expectativas y definir las funcionalidades clave de la plataforma.

Posteriormente, se avanzará a sprints dedicados al diseño y desarrollo iterativo de la solución, que incluyen la construcción de una interfaz accesible, el ajuste del modelo de inteligencia artificial mediante técnicas de prompt engineering, y la integración de capacidades de análisis para evaluar los documentos según los criterios éticos establecidos.

Cada sprint incluirá actividades de:

- Planificación, para definir los objetivos y tareas a desarrollar.
- Desarrollo y pruebas continuas, para asegurar la calidad y funcionalidad de cada entrega.
- Revisión y demostración, donde el Comité de Ética y el Director podrán validar los avances.
- Retrospectiva, para identificar oportunidades de mejora y ajustar la estrategia del equipo.

El proyecto contempla también prácticas complementarias que refuerzan el control y la calidad del desarrollo, tales como:

- Gráficos de burndown, para visualizar el avance del trabajo frente a lo planificado.
- Trabajo colaborativo, combinando desarrollo individual y en pareja para abordar tareas de mayor complejidad.
- Desarrollo guiado por pruebas (TDD), para garantizar que las funcionalidades cumplan con los criterios definidos desde el inicio.

Finalmente, la entrega de la plataforma se realizará en contenedores Docker, permitiendo a los usuarios del comité desplegar la solución cuando lo consideren necesario, en el entorno de su preferencia.

A continuación, se presentan los diagramas BPMN que describen el flujo general del ciclo de vida del proyecto y los subprocesos asociados a cada fase gestionada mediante Scrum.

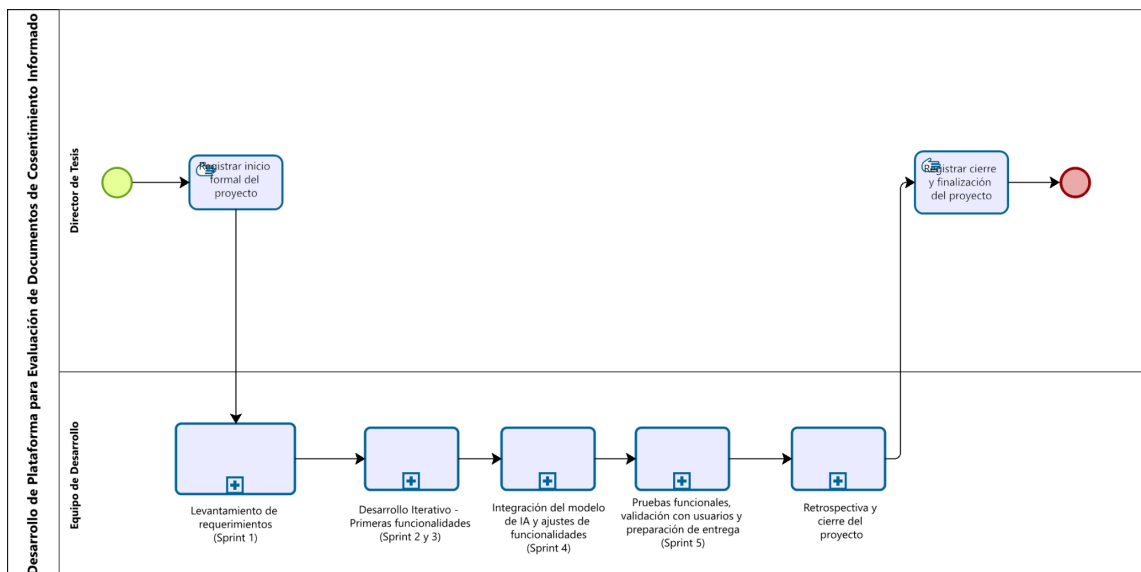


Figura 1 - BPMN Ciclo de vida

Este diagrama BPMN representa el ciclo de vida del proyecto (ver anexo 3) para el desarrollo de una plataforma de evaluación de documentos de consentimiento informado en el Hospital Universitario San Ignacio. En este diagrama se detallan los roles, actividades y entregables clave, con el fin de proporcionar una visión clara y estructurada del desarrollo bajo el marco de trabajo Scrum.

Se identifican dos actores principales:

- *El Director de Tesis*, responsable de formalizar el inicio y cierre del proyecto desde el ámbito académico.
- *El Equipo de Desarrollo*, encargado de ejecutar las actividades técnicas y metodológicas en sprints iterativos.

El flujo inicia con el levantamiento de requerimientos (Sprint 1), donde se recopilan y validan las necesidades y expectativas de los usuarios. Posteriormente, en los sprints 2 y 3, se lleva a cabo el desarrollo iterativo de las primeras funcionalidades, permitiendo entregar versiones funcionales tempranas que puedan ser revisadas y ajustadas.

En el Sprint 4, se realiza la integración del modelo de inteligencia artificial y los ajustes funcionales necesarios para su correcta operación en la plataforma. Luego, en el Sprint 5, se ejecutan las pruebas funcionales y la validación con los usuarios, dejando lista la solución para su entrega final.

Finalmente, el equipo lleva a cabo una retrospectiva y cierre del proyecto, donde se consolidan los resultados, se generan los entregables finales y se formaliza el cierre ante el Director de Tesis.

Los artefactos generados durante el proceso incluyen:

- Mockups y documentación de requisitos en las etapas iniciales.
- Versiones funcionales del sistema entregadas en cada sprint.
- Documentación técnica y de usuario junto con el sistema empaquetado en contenedores Docker para facilitar su despliegue por parte del comité.
- Reportes de prueba y validación, que evidencian el cumplimiento de los criterios funcionales y de calidad.

Este BPMN muestra un flujo secuencial, donde cada etapa representa una entrega incremental que es revisada y validada con los actores involucrados. La descripción detallada de cada sprint está respaldada por artefactos y herramientas que garantizan que las actividades se alineen con las expectativas y necesidades del Comité de Ética.

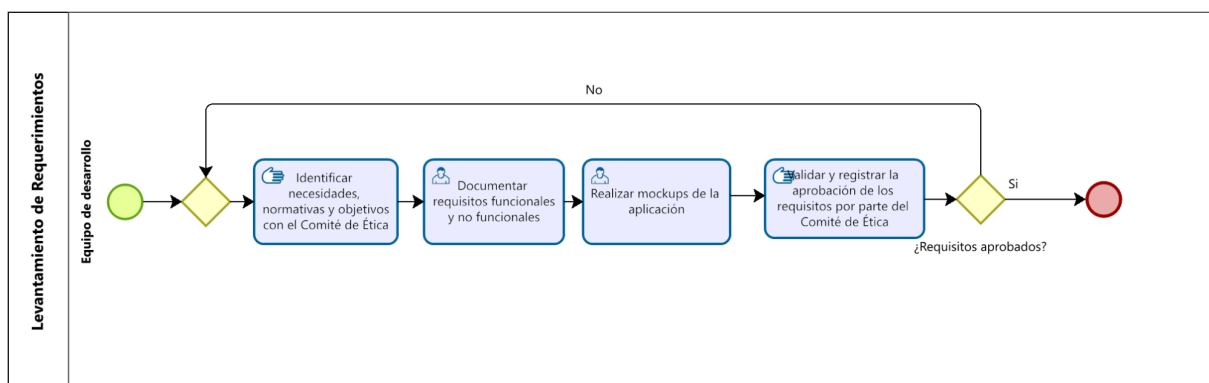


Figura 2 - BPMN Levantamiento de requerimientos

Este diagrama BPMN ilustra la fase de Levantamiento de Requerimientos (Sprint 1) (ver anexo 6) en el desarrollo ágil de la plataforma para el Comité de Ética del Hospital Universitario San Ignacio. En esta etapa, el Equipo de Desarrollo ejecuta las siguientes tareas clave:

1. *Identificar necesidades, normativas y objetivos con el Comité de Ética*
Se definen y clarifican los lineamientos éticos y los objetivos específicos del sistema que guiarán todo el proyecto.
2. *Documentar requisitos funcionales y no funcionales*
Se detallan las especificaciones técnicas y operativas de la plataforma: funcionalidades, rendimiento, usabilidad y seguridad.
3. *Realizar mockups de la aplicación*
Se crean prototipos de la interfaz para que el Comité comprenda cómo será la interacción y la apariencia de la plataforma.
4. *Validar y registrar la aprobación de los requisitos y mockups por parte del Comité de Ética*
El Comité revisa el documento de especificaciones y los prototipos.
 - Si los requisitos y mockups son aprobados, la fase concluye y el proyecto avanza al siguiente sprint.
 - Si no, el flujo retorna al primer paso para ajustar los requerimientos según las observaciones recibidas.

Los artefactos generados en esta fase, el documento de requisitos funcionales y no funcionales y los mockups aprobados, servirán como base sólida para el resto del desarrollo, asegurando el alineamiento con las expectativas y regulaciones del Comité antes de iniciar el diseño y la codificación.

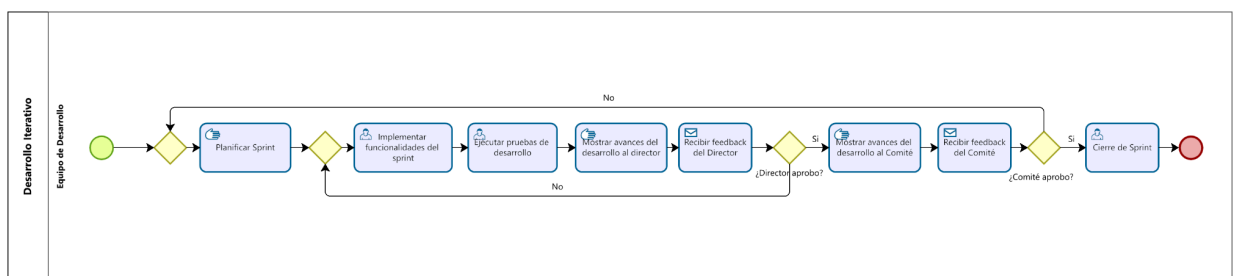


Figura 3 - BPMN Desarrollo Iterativo

En la fase de Desarrollo Iterativo, el Equipo de Desarrollo transforma los requisitos validados en *funcionalidades operativas*, avanzando desde la planificación hasta la entrega de un incremento de producto listo para revisión. A continuación, se detalla paso a paso lo que ocurre en esta etapa, como se muestra en el diagrama BPMN (ver anexo 5):

1. *Sprint Planning*

- El equipo revisa el Product Backlog y selecciona las historias de usuario prioritarias (por ejemplo, carga de documentos de consentimiento, visualización de resultados preliminares del modelo de IA, gestión de errores).
- Para cada historia se definen criterios de aceptación, estimaciones de esfuerzo y tareas concretas. El resultado es el Sprint Backlog, la lista de ítems que se desarrollarán en la iteración.

2. *Implementación de funcionalidades*

- En los primeros días de cada sprint, el equipo de desarrollo implementa las tareas previamente acordadas, las cuales varían según los objetivos de la iteración. Estas labores abarcan tanto el frontend como el backend, y se asignan en función de las necesidades específicas de la fase del proyecto.
- Entre las actividades principales destacan:
 - Desarrollo de la interfaz de usuario de la plataforma.
 - Implementación del envío automático de correos electrónicos.
 - Cálculo de estadísticas y generación de métricas.
 - Visualización de datos y creación de documentos PDF.

De este modo, cada sprint avanza de forma equilibrada en todos los frentes técnicos, garantizando entregas coherentes con los requisitos definidos

3. *Pruebas del desarrollo*

- Durante el sprint se realizaron únicamente pruebas de desarrollo: los desarrolladores evaluaron cada nueva funcionalidad tanto de forma aislada como integrada en la aplicación. De este modo, se confirmó que cada módulo funcionara correctamente por sí mismo y, al mismo tiempo, encajara sin conflictos con el resto del sistema.

4. *Sprint Review con el Director de Tesis*

- Al cierre de la última semana del sprint, el equipo prepara una demo interna para el Director de Tesis. Durante la reunión correspondiente, se presentan los flujos de trabajo implementados hasta ese momento y se muestran los avances obtenidos con respecto al sprint anterior.
- El Director de Tesis ofrece su retroalimentación, enfocándose en aspectos de usabilidad, cumplimiento de los criterios definidos y posibles ajustes. Si no se identifican observaciones, otorga la aprobación para continuar con el desarrollo de las siguientes funcionalidades y coordina la revisión con el Comité de Ética.

5. *Sprint Review con el Comité de Ética*

- Una vez aprobado por el Director de Tesis, se presenta al Comité de Ética la versión refinada para verificar que las funcionalidades iniciales satisfacen los requisitos definidos.
- El Comité de Ética emite sus observaciones sobre la usabilidad de la interfaz, la coherencia del flujo de trabajo y el cumplimiento de las normativas aplicables.

6. Inspección y adaptación (bucle de feedback)

- Si alguna funcionalidad no cumple los criterios de aceptación, el flujo regresa a “Implementación de funcionalidades” para ajustar el código.
- Este bucle se repite hasta que tanto el Director como el Comité validan las funcionalidades desarrolladas.

7. Cierre del Sprint y Retrospectiva

- Una vez aprobadas todas las historias, el equipo da como finalizadas esas tareas, registra en el repositorio el código final, actualiza el Product Backlog con ítems nuevos o pendientes, y elabora una reunión para evaluar el progreso del equipo, y definir las tareas faltantes para realizar en el siguiente sprint.
- En la retrospectiva, el equipo reflexiona sobre lo que funcionó bien y define acciones de mejora para el siguiente sprint.

Al concluir estos Sprints, se obtiene un incremento de software plenamente funcional y alineado con las expectativas del Comité de Ética, listo para integrarse con módulos posteriores y garantizar un desarrollo ágil, transparente y centrado en el valor real para el cliente.

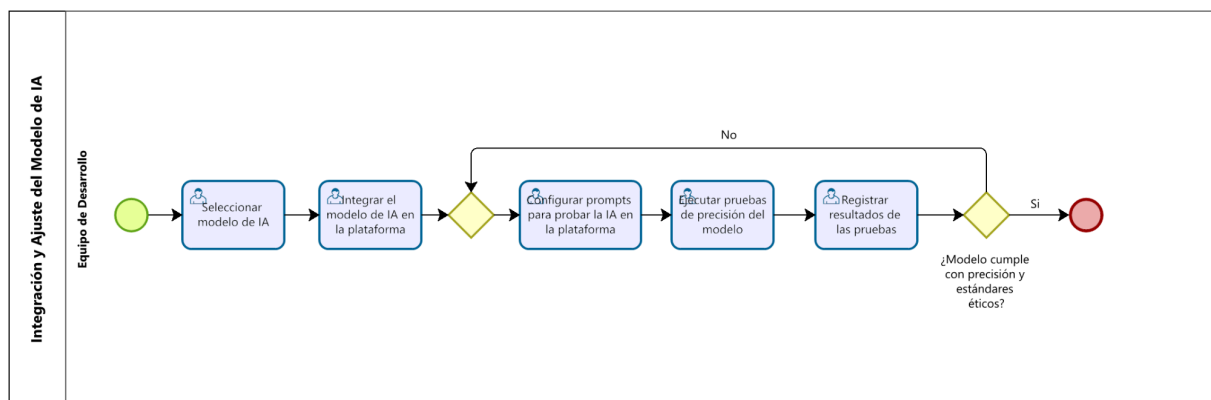


Figura 4 - BPMN Integración y Ajuste del Modelo de IA

En el Sprint 4, el equipo se enfocó en integrar y ajustar el modelo de IA dentro de la plataforma, asegurando su funcionamiento con la precisión y los estándares éticos definidos. En este diagrama BPMN (ver anexo 9) se muestra que el primer paso fue seleccionar el modelo de IA más adecuado, para lo cual se evaluaron aspectos como la calidad y la velocidad de las respuestas, así como los costos asociados. Posteriormente, se incorporó el servicio seleccionado en el backend de la aplicación, exponiendo sus resultados en la interfaz de usuario.

Para respaldar esta decisión, se realizó un benchmark comparativo entre diferentes modelos de IA. Los resultados fueron presentados a la evaluadora principal del Comité de Ética, quien brindó su opinión sobre cuál modelo ofrecía una mejor precisión. Finalmente, se integró el modelo seleccionado como base, aunque la plataforma permite cambiar entre diferentes modelos, de acuerdo con la preferencia del evaluador.

Con la integración técnica completada, el equipo avanzó a la fase de *prompt engineering*, en la que se diseñaron y ajustaron las instrucciones que el modelo recibe antes de procesar los documentos. Este proceso se llevó a cabo mediante iteraciones cortas: cada conjunto de prompts se probaba en un entorno controlado, para identificar los casos de error o ambigüedad.

Al finalizar cada ciclo de pruebas, el equipo analizaba la precisión del modelo frente a los prompts diseñados. Si los resultados no alcanzaban los niveles esperados o se identificaban comportamientos inesperados, se realizaban nuevos ajustes tanto en los prompts como en los parámetros del modelo, hasta lograr un rendimiento adecuado. Solo cuando el modelo superaba estas pruebas se consideraba listo para la demostración interna y la presentación al Director de Tesis.

Este enfoque iterativo de inspección y adaptación, representado en el diagrama BPMN, permitió refinar el modelo de IA de manera controlada, garantizando su correcta integración en la plataforma y su alineación con los criterios éticos definidos por el Comité de Ética. Con el modelo validado y documentado, se cerró el sprint actualizando el Product Backlog y dejando al equipo preparado para avanzar a la siguiente fase de pruebas.

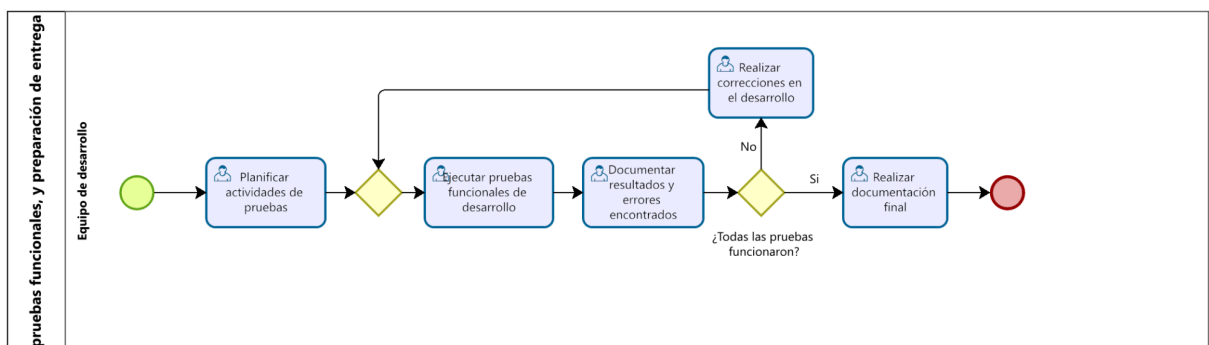


Figura 5 - BPMN Pruebas funcionales y preparación de entrega

Este diagrama BPMN representa la fase de Pruebas funcionales, validación con usuarios y preparación para la entrega (ver Anexo 7) dentro del ciclo de vida del proyecto de la plataforma de evaluación de documentos de consentimiento informado, desarrollado por el Equipo de Desarrollo bajo la metodología Scrum. En esta etapa, el equipo lleva a cabo dos sprints consecutivos (cuatro semanas), durante los cuales se planifican y ejecutan las actividades de prueba.

El proceso inicia con la elaboración y aprobación del Plan de Pruebas, donde se definen los escenarios, los datos de prueba y las métricas de éxito para tres tipos de pruebas:

- *Pruebas de Integración*, ejecutadas con Jest, que validan la correcta interacción entre los módulos del sistema, incluyendo la generación de documentos PDF, la gestión de datos del investigador y la integración con los componentes de inteligencia artificial.
- *Pruebas End-to-End (E2E)*, ejecutadas con Playwright, que simulan los flujos críticos de usuario, desde la carga del formulario por parte del investigador hasta la decisión final del Comité de Ética, que es notificada mediante correo electrónico a los actores correspondientes.

- *Pruebas de Carga*, realizadas con Artillery, que permiten medir la capacidad de respuesta y el rendimiento del sistema bajo condiciones controladas de concurrencia, garantizando que los tiempos de respuesta se mantengan dentro de los umbrales definidos.

Los resultados de las pruebas son gestionados manualmente, registrando los hallazgos, defectos y evidencias en el repositorio del proyecto. Cada defecto identificado se prioriza y se corrige antes de re-ejecutar las pruebas, hasta que se cumplan los criterios de aceptación establecidos.

A continuación, se verifica que todas las pruebas planificadas hayan sido exitosas y que el sistema cumpla con los criterios de aceptación definidos, los cuales son objetivos y medibles. En caso de encontrar fallos, el equipo de desarrollo realiza las correcciones necesarias y repite el ciclo de pruebas hasta alcanzar la estabilidad requerida. Una vez que todas las pruebas se superan satisfactoriamente, el producto queda listo para su entrega y se procede a elaborar la documentación final, que incluye el Informe de Pruebas, el Informe de Seguridad, el Manual de Usuario, la Guía de Despliegue y la Memoria Técnica. Finalmente, con la documentación completa, se avanza a la etapa de cierre del proyecto.

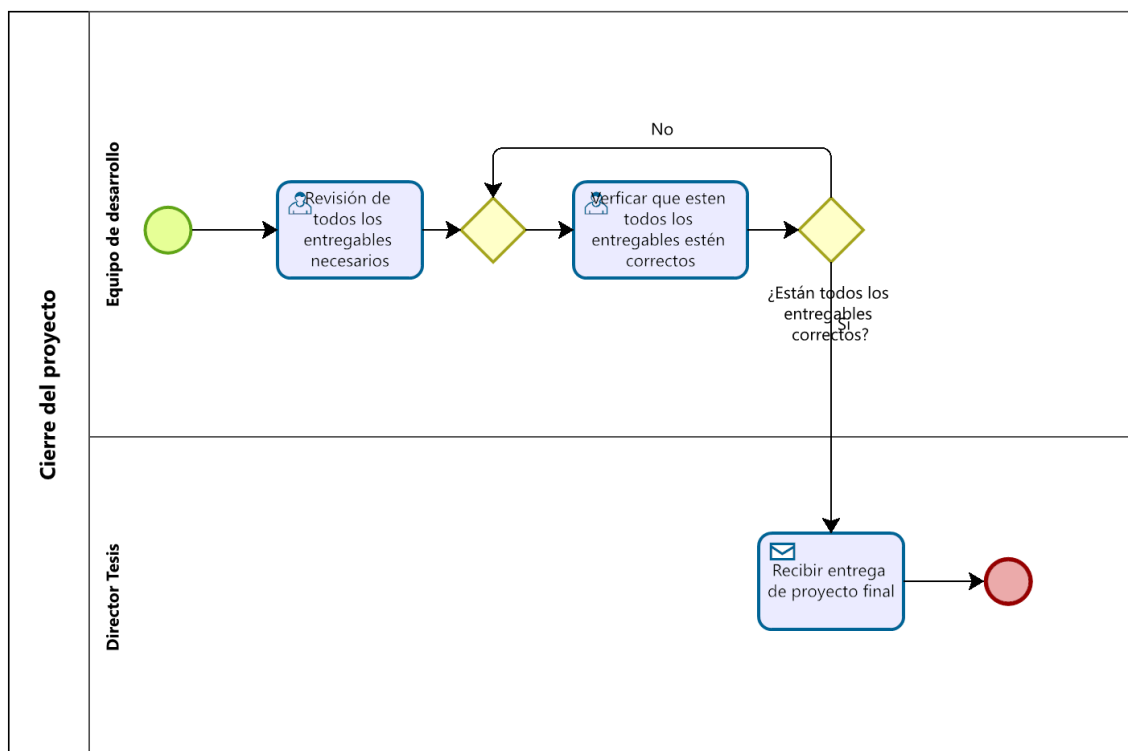


Figura 6 - BPMN Cierre del Proyecto

El diagrama BPMN para el Cierre del proyecto (anexo 4) describe la fase de cierre del proyecto dentro del ciclo de vida de la plataforma de evaluación de documentos de consentimiento informado, desarrollada por el Equipo de Desarrollo y coordinada con el Director de Tesis. En esta etapa, el equipo inicia con la revisión exhaustiva de todos los entregables requeridos para la entrega final del proyecto de grado, incluyendo la documentación técnica, los informes de prueba y otros documentos relevantes. Posteriormente, se verifica que cada entregable cumpla con los estándares de contenido,

ortografía, gramática y demás aspectos formales, contrastándolos con la lista de requisitos y criterios de aceptación previamente definidos. Esta evaluación contempla la funcionalidad, seguridad y conformidad con los atributos de calidad estipulados para el proyecto.

Una vez concluida la revisión, y si todos los entregables cumplen con los criterios establecidos, se gestiona su envío al Director de Tesis para formalizar la entrega y aceptación final de la plataforma, marcando así la conclusión exitosa del proyecto. En caso contrario, el proceso retorna a la etapa de revisión y corrección de los entregables, aplicando los ajustes necesarios y realizando una nueva validación.

Este procedimiento garantiza un cierre ordenado y riguroso, asegurando que el producto final cumple con todos los requisitos antes de su entrega definitiva.

■ **Análisis de alternativas y Justificación**

Para satisfacer las necesidades del Comité de Ética del Hospital Universitario San Ignacio, se evaluaron distintas metodologías de desarrollo de software, desde enfoques predictivos hasta marcos ágiles, y se optó por aplicar Scrum en cinco etapas. Este modelo integra planificación iterativa, desarrollo guiado por pruebas y entregas incrementales de valor, lo que permite mantener un control tanto de los plazos como del progreso, abarcando desde la captura de requisitos hasta el despliegue en contenedores y la entrega definitiva de la plataforma.

- *Levantamiento de requisitos y planificación inicial*

Al inicio del proyecto, el equipo lleva a cabo una reunión inicial con el Director de Tesis, con el fin de definir claramente el alcance, los criterios de aceptación y los atributos de calidad esperados. Posteriormente, se organizan encuentros específicos con los evaluadores y directivos del Comité de Ética para identificar y establecer en detalle los requerimientos y funcionalidades esenciales de la plataforma. Con base en esta información, se crean mockups que serán presentados al Comité para validar que estos cumplen plenamente con sus expectativas y requerimientos iniciales. Al finalizar esta etapa, quedarán definidos el Product Backlog, las historias de usuario, así como los requisitos funcionales y no funcionales. Además, se configurará el repositorio de Git necesario para gestionar de manera ordenada y eficiente las siguientes fases del desarrollo del proyecto.

- *Desarrollo iterativo de funcionalidades iniciales*

En esta etapa, que tendrá una duración aproximada de ocho semanas, el equipo realizará reuniones semanales para verificar avances, evaluar posibles impedimentos y reajustar tareas según sea necesario. El objetivo principal es lograr que la plataforma alcance un estado funcional y estable, con la integración completa entre backend y frontend, de modo que únicamente quede pendiente la incorporación del componente de inteligencia artificial. Adicionalmente, durante esta fase se llevará a cabo la integración con Minio-S3 y se dockerizarán tanto la base de datos como el backend, facilitando así el despliegue en contenedores en las etapas posteriores del proyecto.

- *Integración del modelo de IA y ajustes de funcionalidades*

Con el entorno de desarrollo ya preparado, se integrará el componente de inteligencia artificial. Dicho componente fue seleccionado mediante un benchmark en el que se evaluaron criterios como precisión, tiempos de respuesta y desempeño general de distintos modelos. Los resultados de esta evaluación fueron presentados al evaluador principal para que seleccionara el modelo que mejor se ajusta a las necesidades específicas del proyecto.

Posteriormente, se diseñarán prompts específicos para analizar los documentos de consentimiento informado y verificar si estos cumplen con las normas éticas establecidas por el Comité. La integración conectará el frontend con el servicio de IA, gestionando adecuadamente los errores y tiempos de respuesta. Las respuestas generadas por el modelo serán analizadas mediante un archivo JSON, que posteriormente se traducirá y mostrará directamente en la interfaz, facilitando así que el evaluador pueda revisar fácilmente los resultados proporcionados por la IA. Finalmente, se llevarán a cabo pruebas exhaustivas sobre todas las funcionalidades de la plataforma, con el propósito de asegurar que la incorporación del modelo no haya interferido con otras características ya implementadas.

- *Pruebas funcionales, validación con usuarios y preparación para la entrega*

En esta etapa, se planifican y preparan las pruebas de integración, end-to-end (E2E) y de carga, de acuerdo con el Plan de Pruebas aprobado y alineado con la norma ISO/IEC 29119. El equipo de desarrollo ejecuta las pruebas de integración utilizando Jest para validar la correcta interacción entre los módulos del sistema, así como las pruebas E2E con Playwright para simular los flujos críticos de usuario desde la carga del formulario hasta la decisión final del Comité de Ética. Además, se realizan pruebas de carga ligera con Artillery, que permiten medir la latencia y el rendimiento del sistema bajo condiciones controladas de concurrencia y tiempos de respuesta.

Durante la ejecución, se identifican y corrigen oportunamente los errores detectados, repitiendo las pruebas hasta que se cumplan los criterios de aceptación definidos. Una vez que todas las pruebas son satisfactorias, se documentan los resultados, incluyendo hallazgos, defectos y evidencias relevantes. Finalmente, el equipo revisa y ajusta la documentación general del proyecto, como el Informe de Pruebas, el Informe de Seguridad, el Manual de Usuario, la Guía de Despliegue y la Memoria Técnica, con el propósito de preparar la entrega formal al Director de Tesis y avanzar hacia la etapa de cierre del proyecto.

- *Cierre del proyecto*

La fase de cierre del proyecto representa la etapa final dentro del ciclo de vida de la plataforma de evaluación de documentos de consentimiento informado, la cual es desarrollada por el Equipo de Desarrollo en coordinación con el Director de Tesis. En esta fase, el equipo lleva a cabo una revisión exhaustiva de todos los entregables previstos para la entrega final del proyecto de grado, incluyendo documentación técnica, informes de pruebas y otros documentos relacionados. Cada uno de estos entregables es evaluado cuidadosamente para verificar que cumpla con los criterios establecidos. Si todos los elementos superan satisfactoriamente esta validación, se procede formalmente a gestionar su entrega al Director de Tesis, quien se encargará de realizar la aceptación

definitiva del producto. En caso de detectarse inconsistencias o errores, será necesario regresar a la etapa previa de revisión y corrección, efectuando los ajustes correspondientes y realizando nuevamente una validación completa.

Finalmente, tras la entrega al director, se inicia la preparación formal de la presentación del proyecto, actividad que culmina con el cierre definitivo del mismo.

2.2. Lenguajes y herramientas

Para el desarrollo de este proyecto se planea utilizar un conjunto de tecnologías que permitan implementar de manera efectiva las funcionalidades requeridas. Entre estas, se incluyen:

BackEnd

- JavaScript (Node.js) se utilizará para el desarrollo y la lógica de negocio. Node.js es un entorno de ejecución para JavaScript que permite crear aplicaciones escalables y de alto rendimiento en el backend [23]. Sin embargo, también se contempla la posibilidad de utilizar Java y Spring Boot u otros lenguajes para el desarrollo del backend, si así lo requiere el Hospital.
- TypeScript: Es un superconjunto de JavaScript que agrega tipado estático y otras características avanzadas, facilitando la detección de errores durante el desarrollo y mejorando la mantenibilidad del código [24]. Se empleará tanto en el frontend como en el backend.

Frameworks y FrontEnd

- React: Se seleccionará esta librería para el desarrollo del frontend de la aplicación web, ya que permite crear interfaces de usuario dinámicas y responsivas, mejorando la experiencia del usuario [26][27].
- Express.js: Framework web para Node.js que simplifica la creación de aplicaciones web y APIs RESTful, agilizando el desarrollo del backend [28].

Bases de Datos

- MongoDB: Base de datos NoSQL, ideal para almacenar metadatos y referencias relacionadas con los documentos de consentimiento informado [30]. Facilita el manejo y consulta de datos en formato JSON.
- Minio S3: Sistema de almacenamiento de objetos compatible con Amazon S3, utilizado para guardar los documentos de consentimiento informado en su formato original. Ofrece alta disponibilidad y escalabilidad para el almacenamiento seguro de documentos, permitiendo una recuperación eficiente cuando sean necesarios para su evaluación.

Control de Versiones y Gestión de Código

- Git: Sistema de control de versiones distribuido que permite gestionar los cambios en el código fuente de manera eficiente y colaborar con múltiples desarrolladores [31].

- GitHub: Plataforma de alojamiento de repositorios Git que facilita la colaboración, integración continua y revisión de código entre los miembros del equipo [32].

Modelo LLM

- Modelo de lenguaje grande (LLM): Su propósito es analizar y procesar el documento de consentimiento informado, asistiendo en la evaluación ética. El modelo LLM será afinado mediante el uso de prompts personalizados que se adapten a las consultas del sistema. Las herramientas a utilizar serán Gemini, seleccionada por su extensa capa gratuita y alta precisión en el análisis de documentos, y Groq, elegida por la amplia variedad de modelos que ofrece a un económico costo por token, lo que permitirá optimizar recursos mientras se mantiene la calidad de las evaluaciones.[6][9][16][17]

Entornos de Desarrollo y Herramientas de Modelado

- Visual Studio Code: Editor de código fuente ligero y potente que soporta múltiples lenguajes y cuenta con extensiones útiles para el desarrollo, depuración y colaboración [33].
- Visual Paradigm: Herramienta de modelado UML y diseño de software que ayuda en la planificación y documentación de la arquitectura del sistema [34].
- Lucidchart: Plataforma colaborativa de diagramación basada en la nube que permite crear fácilmente diagramas de flujo, mapas de procesos y modelos visuales para documentar la arquitectura del sistema y los flujos de trabajo del comité de ética

Comunicación y Colaboración

- Microsoft Teams: Plataforma que integra chat, videollamadas y almacenamiento de archivos, facilitando la comunicación y coordinación entre los miembros del equipo y con los stakeholders [35].
- Outlook: Cliente de correo electrónico utilizado para la gestión de comunicaciones formales, reuniones y coordinación con el equipo y el comité de ética [36].
- Discord / WhatsApp: Aplicaciones de mensajería instantánea utilizadas para comunicaciones rápidas y menos formales entre los miembros del equipo, fomentando la colaboración en tiempo real [37][38].
- Google Docs: Herramienta para la creación y edición colaborativa de documentos en línea, utilizada para la redacción y revisión de la documentación del proyecto [39].

Gestión de Proyectos

- ClickUp: Plataforma de gestión de proyectos que permite organizar tareas, asignar responsabilidades y seguir el progreso de los sprints en SCRUM, mejorando la eficiencia y transparencia del equipo [40].

Pruebas Automatizadas

- Jest: Framework de pruebas unitarias que facilita la creación y ejecución de pruebas automatizadas en JavaScript y TypeScript, asegurando la calidad del software [29].

- Artillery: Herramienta de pruebas de carga y rendimiento que permite simular tráfico concurrente y medir métricas como latencia, throughput y tiempos de respuesta en aplicaciones web y APIs. Artillery es especialmente útil para evaluar el comportamiento del sistema bajo condiciones controladas de estrés, ayudando a identificar cuellos de botella y garantizar que los umbrales de rendimiento definidos sean cumplidos [50].
- Playwright: Framework de automatización para pruebas end-to-end (E2E) que permite simular interacciones de usuario en navegadores web de manera confiable y multiplataforma. Playwright facilita la creación de pruebas que validan flujos críticos de la aplicación, desde la interfaz hasta la lógica de negocio, asegurando que las funcionalidades principales operen correctamente en diferentes entornos y dispositivos [51].

■ **Análisis de Alternativas y justificación**

Para abordar la problemática de la evaluación de los formatos de consentimiento informado en los proyectos de investigación presentados al Comité de Ética del Hospital Universitario San Ignacio, se han considerado varias alternativas de solución además de la propuesta principal basada en la adaptación de un modelo de lenguaje grande (LLM) mediante ingeniería de prompts. A continuación, se describen detalladamente cada una de las alternativas consideradas, y sus impactos en los stakeholders.

● *Alternativa 1: Vectorización de los documentos de consentimiento informado*

Una primera alternativa considerada es la vectorización de la información contenida en los documentos de consentimiento informado. Este enfoque implica convertir los textos en representaciones numéricas en un espacio multidimensional mediante técnicas de procesamiento de lenguaje natural (NLP). Al transformar los documentos en vectores, se facilita el análisis semántico y la identificación de patrones dentro de los textos, lo que permite determinar con mayor precisión si cumplen con las normativas éticas establecidas. [15][41][42]

En esta solución, los documentos serían preprocesados para normalizar el texto y eliminar ruido. Posteriormente, se aplicarían modelos de embeddings como Word2Vec, GloVe o Doc2Vec para obtener representaciones vectoriales que capturan relaciones semánticas y contextuales entre palabras y frases. Lo que permitiría comparar los documentos con vectores de referencia que encapsulan los criterios éticos, evaluando el grado de similitud y detectando áreas de incumplimiento o falta de información.

El impacto técnico de esta alternativa en los stakeholders es significativo. Ya que se le proporcionaría una herramienta avanzada que mejora la precisión en la detección de inconsistencias e irregularidades a los evaluadores, reduciendo la necesidad de revisiones exhaustivas manuales. Además, al facilitar el análisis semántico, se podrían identificar matices en el lenguaje que podrían pasar desapercibidos en una revisión tradicional. Cabe destacar que este enfoque requiere conocimientos especializados en procesamiento de lenguaje natural para su implementación, y por esta misma razón, no es la propuesta principal. [15]

- *Alternativa 2: Desarrollo de un sistema basado en reglas y expresiones regulares*

Otra opción es desarrollar un sistema de evaluación automatizada fundamentado en reglas predefinidas y expresiones regulares. Este sistema se enfocaría en identificar patrones específicos y palabras clave en los documentos que indican el cumplimiento o incumplimiento de los criterios éticos [43][44]. Al establecer un conjunto de reglas lógicas y patrones textuales, el sistema podría escanear los consentimientos informados para detectar la presencia o ausencia de secciones y términos obligatorios.

Este enfoque tendría un impacto técnico reducido en los stakeholders. Esto se debe a que se ofrecería una herramienta sencilla que agiliza la revisión inicial al resaltar elementos clave, facilitando la identificación rápida de posibles omisiones o inconsistencias. Sin embargo, la rigidez del sistema podría reducir su desempeño ante variaciones en el lenguaje o estructura de los documentos, limitando su eficacia. Lo que resulta en una mejora poco significativa en la velocidad de evaluación ética, con un efecto menor en la aceleración de proyectos de investigación.

- *Alternativa 3: Utilización de plataformas comerciales de análisis de documentos*

Otra alternativa es emplear plataformas comerciales existentes que ofrecen servicios de análisis y extracción de información de documentos, como Google Cloud Document AI, Amazon Textract o IBM Watson Discovery [17][45][46][47][14]. Estas herramientas utilizan inteligencia artificial para procesar documentos no estructurados y pueden configurarse para identificar elementos clave en los consentimientos informados.

Implementar esta solución implicaría cargar los documentos en la plataforma seleccionada, o integrar una plataforma propia mediante una API, para que los documentos sean procesados y se extraiga la información relevante según los criterios éticos establecidos, lo que reduciría significativamente el tiempo dedicado al análisis manual. Sin embargo, manejar información sensible a través de plataformas de terceros plantea desafíos en términos de privacidad y seguridad de los datos, lo que podría complicar su uso debido a las normativas de protección de datos y la necesidad de garantizar la confidencialidad de la información.

No obstante, el impacto técnico en los stakeholders sería considerable. Esto debido a que los evaluadores contarían con una herramienta poderosa que mejoraría la eficiencia y precisión en la revisión de los documentos. Por otro lado, esta alternativa implica costos asociados al uso de servicios en la nube, los cuales pueden ser significativos dependiendo del volumen de documentos procesados, lo que haría menos interesante esta propuesta.

- *Alternativa 4: Entrenamiento de un modelo de inteligencia artificial específico con datos locales*

Finalmente, se ha considerado la posibilidad de entrenar un modelo de inteligencia artificial utilizando datos locales de consentimiento informado previamente evaluados. Este modelo, basado en técnicas de aprendizaje profundo, aprendería a identificar el cumplimiento de los criterios éticos a partir

de ejemplos etiquetados, permitiendo así una evaluación asistida adaptada a las necesidades del comité. [16][48][49]

Para implementar esta solución, sería necesario recopilar un conjunto amplio de documentos evaluados, pre-procesarlos y etiquetarlos según los criterios establecidos. Posteriormente, se entrenaría un modelo utilizando algoritmos de redes neuronales y ajustándolo para maximizar su precisión. Sin embargo, durante este proceso, el manejo de datos sensibles plantea consideraciones éticas y legales que deben ser cuidadosamente gestionadas para garantizar la privacidad y seguridad de la información.

Además, este enfoque representa un desafío significativo. Esto debido a que el equipo actualmente carece del conocimiento especializado necesario para desarrollar una solución de este tipo, por lo que se requeriría un aprendizaje considerable para adquirir habilidades en el entrenamiento y ajuste de modelos de aprendizaje profundo. Asimismo, sería necesaria una inversión en recursos computacionales, como GPUs, para el entrenamiento del modelo, lo cual podría representar un costo adicional.

El impacto técnico en los stakeholders sería notable. Para los evaluadores, este sistema ofrecería una herramienta altamente precisa que podría reducir drásticamente el tiempo de revisión y mejorar la consistencia en las evaluaciones. No obstante, debido a los desafíos mencionados, la implementación de esta alternativa podría ser compleja y demandante en términos de tiempo y recursos, lo que la convierte en una opción menos viable en comparación con la solución propuesta.

Justificación de la elección de la solución propuesta

La solución propuesta, basada en el uso de modelos de lenguaje grandes (LLMs) mediante ingeniería de prompts, se ha seleccionado por ofrecer un equilibrio óptimo entre eficiencia, precisión y viabilidad técnica. A diferencia de las otras alternativas, esta opción permite aprovechar modelos existentes sin incurrir en costos de desarrollo o en la necesidad de manejar grandes cantidades de datos sensibles para entrenamiento. Además, al ajustar los LLMs mediante prompts, se obtiene una herramienta flexible y adaptable que puede comprender variaciones en el lenguaje y estructura de los documentos, superando las limitaciones de los sistemas basados en reglas o expresiones regulares. Esto facilita su implementación, convirtiéndola en una solución mucho más efectiva para asistir a los evaluadores sin reemplazarlos, al mismo tiempo que garantiza un alto nivel de cumplimiento ético en los proyectos de investigación.[11][17]

2.3. Plan de aceptación de producto

Este plan tiene como propósito definir, de manera clara y medible, los criterios de aceptación para cada uno de los entregables desarrollados durante el proyecto. Estos criterios fueron acordados entre el equipo de desarrollo y el Director de Tesis. El objetivo es garantizar que cada entrega cumpla con los estándares de calidad, funcionalidad y valor definidos, y que sea aceptada de manera satisfactoria al finalizar cada sprint.

Dado que el proyecto se gestiona bajo la metodología Scrum, cada entregable se valida iterativamente en las reuniones de revisión de sprint, donde el Director verifica que se cumplan los criterios establecidos antes de considerar el incremento como completado. Los criterios presentados a continuación son objetivos, verificables y están alineados

con el propósito de asegurar que el producto entregue valor real al proceso de evaluación ética del comité.

1. Levantamiento y Validación de Requerimientos (Sprint 1 y 2)

■ *Criterios de Aceptación:*

- Entregar una lista priorizada de historias de usuario que reflejen las necesidades y expectativas del comité de ética.
- Presentar mockups interactivos que representen las funcionalidades y flujos de la plataforma.
- Validar y aprobar estos entregables en una sesión de revisión con el Director y actores clave del comité.

■ *Técnicas y Herramientas de Medición:*

- Revisión de historias de usuario en la herramienta de gestión (ClickUp).
- Validación de mockups mediante herramientas como Figma.
- Retroalimentación registrada en actas o comentarios durante las sesiones de revisión.

2. Desarrollo Iterativo de Funcionalidades Iniciales (Sprints 3 - 6)

■ *Criterios de Aceptación:*

- Entregar funcionalidades incrementales que permitan la navegación, carga de documentos y visualización de resultados.
- Garantizar que las funcionalidades cumplan con las historias de usuario y los criterios definidos en el backlog.
- Validar el funcionamiento en la revisión de sprint con el Director.

■ *Técnicas y Herramientas de Medición:*

- Pruebas de desarrollo ejecutadas por el equipo y validadas en la revisión del sprint.
- Demostración funcional en ambiente controlado ante el Director.

3. Integración del Modelo de IA y Ajustes (Sprint 7)

■ *Criterios de Aceptación:*

- Integrar el modelo de IA en la plataforma, permitiendo que procese documentos reales.
- Asegurar que el modelo entregue resultados alineados con los criterios normativos definidos.

- Permitir que el Director valide el funcionamiento del modelo a través de una sesión en la que se le presente su desempeño y características.

■ *Técnicas y Herramientas de Medición:*

- Se ejecutan pruebas A/B para comparar distintas versiones del modelo, así como diferentes modelos, como por ejemplo DeepSeek y Gemini.
- Evaluación de precisión y relevancia mediante validaciones con casos reales proporcionados por el comité.
- Sesión con el Comité de Ética para validar cuál modelo entrega mejores respuestas y se ajusta más a sus preferencias.

4. Pruebas Funcionales, Validación con Usuarios y Preparación de Entrega (Sprint 8)

■ *Criterios de Aceptación:*

- Ejecutar pruebas funcionales, de integración, carga y end-to-end (E2E) completas para validar el comportamiento integral del sistema.
- Implementar ajustes en las funcionalidades según los hallazgos detectados durante las pruebas.
- Documentar todos los resultados y correcciones, y elaborar el Manual de Usuario para facilitar la comprensión y uso de la plataforma por parte de los usuarios finales.
- Completar la documentación técnica final, incluyendo la Guía de Despliegue, la Memoria Técnica, el Informe de Pruebas y el Informe de Seguridad, incorporando hallazgos y evidencias relevantes.

■ *Técnicas y Herramientas de Medición:*

- Registro detallado de las pruebas funcionales, así como su documentación, en el sistema de gestión de archivos utilizado (Google Drive).
- Grabaciones de las sesiones de validación con usuarios.
- Validación y aprobación de los ajustes por parte del Director en la revisión del sprint.

5. Retrospectiva y Cierre del Proyecto (Sprint 9)

■ *Criterios de Aceptación:*

- Consolidar y entregar la documentación final del proyecto, que incluye la Memoria de Trabajo de Grado, el Plan de Proyecto, la Especificación de Requisitos (SRS), la Especificación del

Diseño (SDD), la Propuesta de Trabajo de Grado, y demás entregables aprobados.

- Realizar la presentación formal de la plataforma completa y validada ante el Director de Tesis y los jurados evaluadores, garantizando que esté lista para su evaluación académica.
- Obtener la aprobación formal del cierre del proyecto por parte del Director de Tesis y los jurados evaluadores.

■ *Técnicas y Herramientas de Medición:*

- Validación del cierre durante presentación final del proyecto, donde se revisan los entregables y resultados alcanzados.
- Registro del acta de cierre firmada por el Director de Tesis.
- Documentación de la entrega formal del proyecto en los sistemas de control y gestión de proyectos de grado de la Pontificia Universidad Javeriana (Trazadocs), garantizando la trazabilidad y disponibilidad futura.

2.4. Organización del proyecto y comunicación

■ **Interfaces Externas**

Entidad	Descripción	Contacto
Hospital Universitario San Ignacio	El Comité de Ética es el principal stakeholder del proyecto, encargado de evaluar la conformidad ética de los proyectos de investigación presentados. Este comité será el usuario final de la plataforma desarrollada.	Evaluadora: Angie Kares Cardona Camacho akcardona@husi.org.c Directora Oficina de Investigación: María Margarita Manrique Andrade mmmanrique@husi.org.co

Tabla 4. Interfaces Externas

■ **Organigrama y descripción de roles**

El equipo de desarrollo está conformado por cuatro integrantes, quienes asumirán roles principales basados en sus especialidades, pero colaborarán de manera conjunta en todas las áreas del proyecto para asegurar una ejecución fluida y sinérgica. A continuación, se describen los roles y responsabilidades clave, y cómo cada miembro liderará un aspecto del desarrollo de la plataforma.

1. Líder de Desarrollo de Software – Laura Valentina Ovalle

- *Responsabilidades:*
 - Liderar la arquitectura y el desarrollo del backend de la plataforma.

- Asegurar la escalabilidad y eficiencia del sistema.
 - Supervisar la integración del modelo de inteligencia artificial en la plataforma.
 - *Colaboración:* Aunque será el principal encargado del backend, Laura colaborará estrechamente con los demás en la implementación de la interfaz y en la resolución de problemas técnicos que surjan en otras áreas.
2. Líder de Frontend y Experiencia de Usuario – Jose Manuel Rodríguez
- *Responsabilidades:*
 - Encargado del diseño e implementación de la interfaz de usuario (UI).
 - Asegurar que la plataforma sea accesible y fácil de usar para los miembros del Comité de Ética.
 - Liderar las pruebas de usabilidad y la implementación de mejoras visuales y funcionales.
 - *Colaboración:* Jose Manuel Rodríguez liderará las decisiones sobre diseño y experiencia de usuario, pero colaborará en la integración con el backend y la optimización del sistema junto con el equipo.
3. Líder de Modelos de Inteligencia Artificial y Prompts – Michael Joel González
- *Responsabilidades:*
 - Liderar la configuración y ajuste del modelo de IA utilizando técnicas de *prompt engineering*.
 - Supervisar el funcionamiento del modelo en la plataforma, asegurando que cumpla con los criterios éticos establecidos.
 - Realizar pruebas de precisión y mejorar el rendimiento del modelo.
 - *Colaboración:* Michael liderará la implementación del modelo de IA, pero colaborará con el equipo en la integración del modelo en el sistema y en el ajuste de las funcionalidades que dependan de su desempeño.
4. Líder de Gestión de Proyecto y Calidad – María Paula Cardona
- *Responsabilidades:*
 - Encargada de la planificación y gestión del proyecto, asegurando que se cumplan los plazos y los objetivos de cada sprint.
 - Supervisar las métricas de calidad, control de versiones y documentación del proyecto.
 - Asegurar que se sigan los procesos de desarrollo ágil (Scrum) y que se mantenga la comunicación fluida con los stakeholders.
 - *Colaboración:* Aunque su foco estará en la gestión, María Paula participará en el desarrollo y pruebas de la plataforma, además de coordinar las interacciones con el Comité de Ética.

Trabajo Colaborativo

Si bien cada miembro del equipo lidera un área específica, todos los integrantes trabajarán en conjunto en todos los aspectos del proyecto, desde la implementación de

funcionalidades hasta las pruebas y la optimización del sistema. Esta estructura asegura flexibilidad y soporte mutuo en cada fase del desarrollo.

3. Administración del proyecto

3.1. Métodos y herramientas de estimación

Dado que el proyecto se desarrollará bajo el marco metodológico SCRUM, los métodos de estimación se basarán en enfoques de desarrollo ágil. Esto permitirá realizar estimaciones basadas en la planificación iterativa y flexible, acorde a la naturaleza del proyecto. ágil. Esto permitirá realizar estimaciones basadas tanto en la planificación iterativa y flexible, acorde a la naturaleza del proyecto.

Estimación del Software

Para la estimación del tamaño y complejidad del software, se utilizarán las siguientes técnicas:

1. SCRUM: Estimación Ágil para el Desarrollo de la Plataforma
 - Para las fases de desarrollo del software y la implementación de la plataforma, se aplicarán técnicas ágiles de estimación basadas en Scrum. Las tareas se desglosarán en historias de usuario y se priorizarán en función de su valor para el cliente.
 - Se utilizará la técnica de Planning Poker como método de estimación, donde los desarrolladores asignarán puntos de historia (story points) para estimar el esfuerzo relativo de cada tarea. Las tareas más complejas, como la integración del modelo de IA o la creación de la interfaz de usuario, recibirán más puntos, mientras que tareas más simples, como actualización del Backlog, recibirán menos puntos.
 - Se usará Click-Up como herramienta de gestión ágil para organizar los sprints, asignar tareas y registrar las estimaciones de puntos de historia.
2. Modelado y Evaluación del Modelo de IA
 - La estimación de la fase de modelado y evaluación se realizará considerando la complejidad del modelo de inteligencia artificial que se utilizará. Se tendrá en cuenta tanto el tiempo necesario para integrar el modelo como el esfuerzo requerido para evaluarlo.
 - Se empleará el método de métricas de desempeño, donde se estimarán los tiempos necesarios para obtener respuestas con la precisión deseada. Se considerarán pruebas de precisión y tiempos de respuesta, utilizando datos de prueba proporcionados por el comité.

Estimación del Proyecto

Dado que el tiempo total disponible para el desarrollo del proyecto es de aproximadamente 18 semanas (lo que dura el semestre académico en la Pontificia Universidad Javeriana), se utilizará el marco de trabajo Scrum, dividiendo el proyecto en sprints iterativos de dos semanas cada uno. Esto permitirá un progreso constante, entregando funcionalidades parciales revisadas continuamente por el Comité de Ética y el Director de Tesis, asegurando calidad y alineación con los requerimientos definidos.

1. Levantamiento y validación de requerimientos

Esta etapa inicial se enfocará en la identificación detallada de requisitos funcionales y no funcionales junto con el Comité de Ética. Se entregarán mockups interactivos para validar los principales flujos de uso de la plataforma.

- *Tiempo estimado:* 4 semanas (2 sprints).

2. Desarrollo iterativo de la plataforma

Durante esta fase, se desarrollará progresivamente la interfaz web y las funcionalidades básicas como la carga de documentos, visualización de resultados preliminares y gestión de usuarios. El desarrollo se realizará mediante iteraciones frecuentes con revisiones continuas del Comité.

- *Tiempo estimado:* 8 semanas (4 sprints).

3. Integración y ajuste del modelo de inteligencia artificial

En esta etapa se integrará un modelo de IA previamente evaluado (por ejemplo, DeepSeek o Gemini), ajustándolo mediante técnicas avanzadas de prompt engineering para que cumpla con los criterios éticos definidos por el Comité. Paralelamente, se realizarán pruebas de precisión y relevancia, permitiendo ajustes necesarios según la retroalimentación obtenida.

- *Tiempo estimado:* 2 semanas (1 sprint).

4. Pruebas funcionales, validación con usuarios y preparación para la entrega

En esta etapa se ejecutarán pruebas completas de integración, end-to-end (E2E) y carga utilizando Jest, Playwright y Artillery respectivamente, asegurando que la plataforma cumple con los criterios definidos de rendimiento y calidad. Los hallazgos serán documentados detalladamente y corregidos según prioridad. Finalmente, se elaborará la documentación técnica, incluyendo Manual de Usuario, Guía de Despliegue, Informe de Pruebas y el Informe de Seguridad.

- *Tiempo estimado:* 2 semanas (1 sprint).

5. Cierre del proyecto

La fase final consiste en consolidar todos los entregables del proyecto: Memoria de Trabajo de Grado, Plan de Proyecto, Especificación de Requisitos (SRS), Especificación de Diseño (SDD), Propuesta de Trabajo de Grado, Licencia de Uso, Carta de Autorización y Descripción de la Tesis. Se realizará una presentación formal ante el Director de Tesis y los jurados evaluadores para obtener su aprobación final. Finalmente, todos los entregables se archivarán en el sistema institucional de gestión documental (Trazadocs), garantizando la transferencia adecuada de conocimiento y facilitando su disponibilidad futura.

- *Tiempo estimado:* 2 semanas (1 sprint).

Estimación de presupuesto

No se ha asignado un presupuesto económico específico para este proyecto. Por tanto, las estimaciones se enfocan exclusivamente en la gestión de tiempos y recursos humanos y técnicos necesarios para la ejecución eficiente de las tareas. En lugar de calcular costos monetarios, se prioriza la optimización de la planificación temporal y la utilización efectiva de los recursos disponibles, tales como las horas de trabajo del equipo, la infraestructura tecnológica existente y el acceso a herramientas de desarrollo y prueba. Este enfoque permite maximizar la productividad y garantizar el cumplimiento de los objetivos dentro de las restricciones de tiempo establecidas.

3.2. Inicio del proyecto

■ Entrenamiento del personal

Debido a que el desarrollo de este proyecto requiere conocimientos técnicos en varias áreas pertenecientes a la disciplina de la Ingeniería de sistemas, a continuación se define el plan de entrenamiento para cada una de ellas por separado:

1. Desarrollo de Software

El equipo de desarrollo tiene pleno dominio en todos los aspectos relacionados con desarrollo de software y gestión de proyectos ágiles, razón por la cual no será necesario entrenamiento adicional en esta área, ya que los integrantes aplicarán sus conocimientos en la implementación y desarrollo de la plataforma de evaluación ética.

2. Prompt Engineering

El equipo necesita adquirir habilidades en prompt engineering para diseñar las instrucciones que el modelo de IA utilizará para identificar el cumplimiento de normativas éticas en los proyectos evaluados. De esta manera, el proyecto de enseñanza se realizará por medio de las siguientes estrategias:

- Se realizarán sesiones de capacitación internas en las cuales el equipo investigará y aprenderá técnicas de creación y refinamiento de prompts.
- Se fomentará la investigación continua y la revisión de recursos especializados en prompt engineering, de manera que el equipo pueda aplicar las mejores prácticas en esta área.

3. Integración de LLM

La integración de modelos de lenguaje de gran tamaño (LLM) es una tarea que requiere un conocimiento especializado en lenguajes y técnicas de integración de LLM [22][17]. Para esto, el equipo de desarrollo contará con la orientación de un experto en IA, quien guiará al equipo en los aspectos más técnicos de esta parte del proyecto.

El experto en IA proporcionará la capacitación necesaria sobre los lenguajes y técnicas adecuados para la integración del LLM, asegurando que el modelo funcione correctamente en la plataforma desarrollada.

Para lograr esto, se utilizarán las siguientes estrategias:

- El experto en IA impartirá sesiones prácticas sobre las mejores prácticas para integrar y optimizar LLM dentro del sistema.
- El equipo participará en revisiones regulares con el experto para asegurar que el modelo está siendo desplegado de manera eficiente y que cualquier desafío técnico sea resuelto oportunamente.
- A lo largo del proyecto, el experto en IA también ofrecerá asesoría continua para guiar al equipo en la integración del LLM dentro del flujo de trabajo de la plataforma.

■ Infraestructura

A continuación, se detallan las tareas necesarias para obtener, poner en marcha y mantener todas las herramientas de software e infraestructura de hardware necesarias para el desarrollo del proyecto. Se asignan roles específicos y se establecen cronogramas claros para cada actividad.

Para la obtención y puesta en marcha de herramientas de software, el equipo utilizará diversas plataformas seleccionadas previamente en la sección de Lenguajes y Herramientas. Las principales herramientas serán GitHub para el control de versiones, ClickUp para la gestión del proyecto, Google Docs para documentación colaborativa, y Figma para el diseño de interfaces y experiencia de usuario.

El Líder de Gestión de Proyecto y Calidad será responsable de la configuración inicial de estas herramientas durante la primera semana del proyecto. Sus actividades incluirán la creación y configuración de repositorios en GitHub, la estructuración inicial de tableros y tareas en ClickUp, y la preparación de espacios compartidos en Google Docs y Figma. Esta configuración inicial asegurará que todos los miembros del equipo tengan acceso adecuado y puedan familiarizarse rápidamente con las herramientas seleccionadas.

En cuanto a la infraestructura de hardware, el desarrollo se llevará a cabo utilizando los equipos personales de los integrantes del equipo. El Líder de Desarrollo de Software realizará recomendaciones técnicas oportunas sobre posibles actualizaciones o mejoras de hardware si estas resultan necesarias para garantizar el rendimiento adecuado del entorno de desarrollo local.

Se gestionarán recursos computacionales adicionales para el procesamiento de los modelos de lenguaje (LLMs), utilizando servicios en la nube, como Groq u otras plataformas adecuadas. El Líder de Modelos de Inteligencia Artificial y Prompts será responsable de configurar y administrar el entorno seleccionado, garantizando la continuidad operativa del desarrollo y permitiendo ajustes eficientes en los modelos.

Respecto al desarrollo del backend, aunque inicialmente se seleccionaron TypeScript y Express.ts, el equipo contempla la posibilidad de implementar otras tecnologías, como Java con Spring Boot, según las necesidades específicas del Hospital Universitario San

Ignacio. El Líder de Desarrollo de Software evaluará estas alternativas en las fases iniciales del proyecto, considerando rendimiento, escalabilidad y facilidad de mantenimiento.

La integración y ajuste del modelo de IA mediante técnicas de prompt engineering será responsabilidad del Líder de Modelos de Inteligencia Artificial y Prompts. Esta persona coordinará la implementación y ajustes continuos del modelo, tanto en entornos locales como en los servicios en la nube seleccionados, asegurando resultados alineados con los requerimientos del Comité de Ética.

El mantenimiento continuo de la infraestructura de software será responsabilidad de todo el equipo, con tareas programadas mensualmente y revisiones adicionales según necesidades puntuales. Estas actividades incluirán actualizaciones periódicas de todas las herramientas, asegurando versiones actualizadas, seguras y compatibles.

La creación de mockups interactivos y el diseño de la experiencia de usuario serán tareas a cargo del Líder de Frontend y Experiencia de Usuario. Utilizando Figma, este líder garantizará que los diseños sean intuitivos, accesibles y alineados con los objetivos definidos en el proyecto.

Finalmente, el control y gestión del código fuente se realizará mediante GitHub. Todos los desarrolladores serán responsables de mantener actualizados los repositorios, realizando integraciones y revisiones de código al finalizar cada sprint, asegurando así la calidad y coherencia del producto entregado.

3.3. Planes de trabajo del proyecto

■ Descomposición de actividades

A continuación se presenta el diagrama WBS (Work Breakdown Structure) que presenta la estructura jerárquica y la descomposición detallada de las actividades y entregables del proyecto, facilitando la planificación, gestión y seguimiento efectivo de cada fase (Anexo 13):

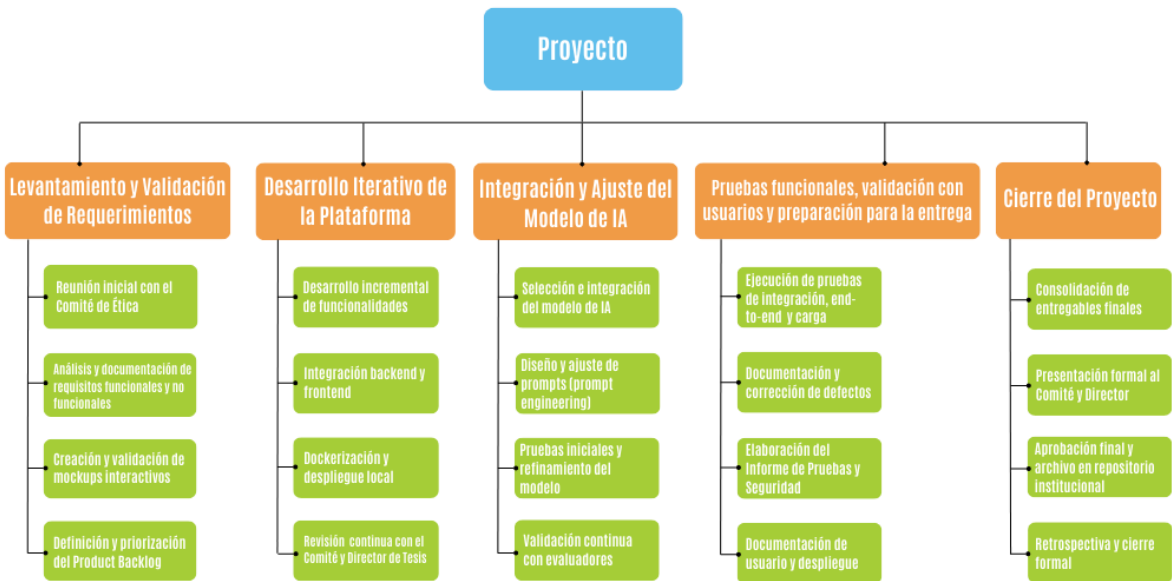


Figura 7 - WBS Descomposición de actividades

- Levantamiento y Validación de Requerimientos
 - *Reunión inicial con el Comité de Ética*: Se realiza un encuentro para identificar las normativas aplicables y definir los requerimientos funcionales y no funcionales del sistema, estableciendo la base para las actividades posteriores.
 - *Análisis y documentación de requisitos*: Se evalúan los procesos actuales y se documentan los requerimientos que guiarán el diseño y desarrollo de la plataforma.
 - *Creación y validación de mockups interactivos*: Se diseñan prototipos preliminares de la interfaz que serán revisados y validados por el Comité para asegurar la alineación con sus expectativas.
 - *Definición y priorización del Product Backlog*: Se organiza y prioriza el conjunto de historias de usuario y tareas que conformarán el backlog del proyecto.

- Desarrollo Iterativo de la Plataforma
 - *Desarrollo incremental de funcionalidades*: Se implementan progresivamente las funcionalidades definidas, combinando frontend y backend.
 - *Integración backend y frontend*: Se asegura la correcta integración entre ambos componentes para garantizar una experiencia de usuario fluida.
 - *Dockerización y despliegue local*: Se prepara el entorno de ejecución utilizando contenedores para facilitar el despliegue y pruebas.
 - *Revisión continua con el Comité y Director de Tesis*: Se presentan avances periódicos para obtener retroalimentación y realizar ajustes oportunos.

- Integración y Ajuste del Modelo de IA
 - *Selección e integración del modelo de IA*: Se elige el modelo que mejor se ajusta a los requerimientos éticos y se integra a la plataforma.
 - *Diseño y ajuste de prompts (prompt engineering)*: Se refinan las instrucciones que recibe el modelo para mejorar su precisión.
 - *Pruebas iniciales y refinamiento*: Se realizan pruebas para evaluar y optimizar el desempeño del modelo.
 - *Validación continua con evaluadores*: Se obtiene retroalimentación constante del Comité de Ética para garantizar el cumplimiento de criterios.

- Pruebas Funcionales, Validación con Usuarios y Preparación para la Entrega
 - *Ejecución de pruebas de integración, end-to-end y carga*: Se realizan pruebas completas para validar el comportamiento y rendimiento del sistema.
 - *Documentación y corrección de defectos*: Se registran los hallazgos y se aplican las correcciones necesarias.
 - *Elaboración del Informe de Pruebas y Seguridad*: Se documentan los resultados y evidencias de las pruebas realizadas.
 - *Documentación de usuario y despliegue*: Se preparan materiales de apoyo para el uso y la implementación del sistema.

- Cierre del Proyecto
 - *Consolidación de entregables finales*: Se reúnen todos los documentos y productos desarrollados durante el proyecto.
 - *Presentación formal al Comité y Director*: Se realiza la presentación final del proyecto para su evaluación.
 - *Aprobación final y archivo en repositorio institucional*: Se obtiene la aceptación formal y se archiva toda la documentación.
 - *Retrospectiva y cierre formal*: Se realiza una evaluación final del proceso para identificar aprendizajes y concluir el proyecto oficialmente.

■ Calendarización

A continuación, se presentará el BPMN que se realizó (ver Anexo 12), para modelar cómo sería el proceso de Scrum para tareas iterativas:

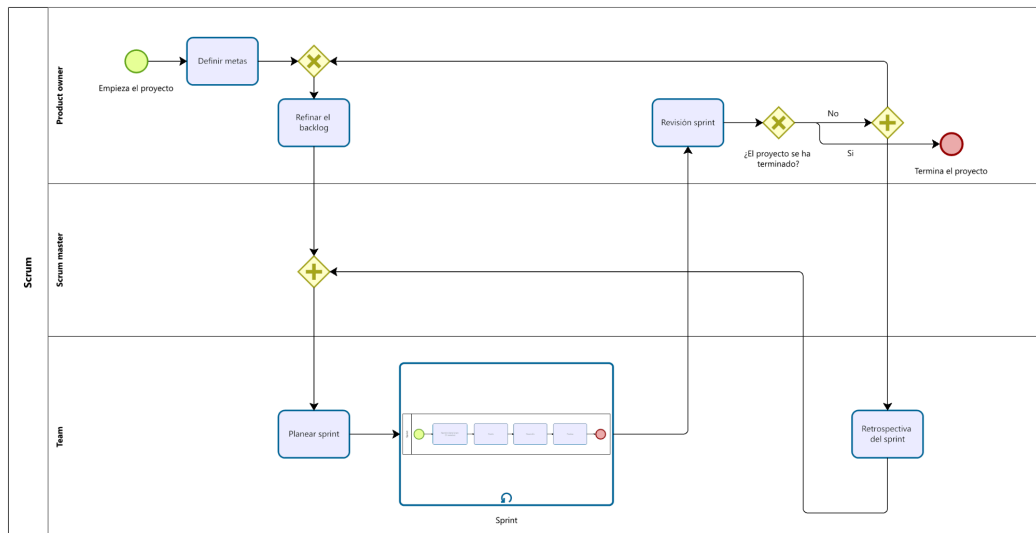


Figura 8 - BPMN Scrum de tareas iterativas

Este diagrama muestra el ciclo Scrum desde una perspectiva de roles (Product Owner, Scrum Master, y el equipo). El proceso comienza con la definición de metas y el refinamiento del backlog, seguido por la planificación del sprint y el ciclo del sprint en sí (representado en el segundo diagrama). Al finalizar el sprint, se realiza una revisión para verificar el progreso, y surge una decisión clave: ¿el proyecto ha llegado a su fin o necesita más iteraciones? Si la respuesta es no, el equipo regresa al backlog para planear el siguiente sprint. Si la respuesta es sí, el proyecto finaliza.

Adicional a eso, antes de comenzar un nuevo sprint, se lleva a cabo una retrospectiva, en la que el equipo reflexiona sobre el proceso y busca mejoras, promoviendo así la mejora continua en cada iteración.

El siguiente BPMN (ver Anexo 11) explica el subproceso de sprint:

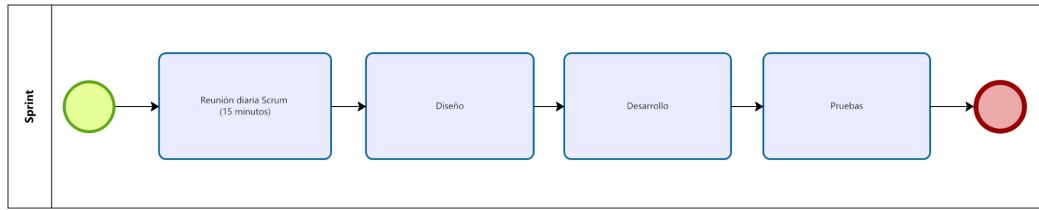


Figura 10 - BPMN Sprints

Este diagrama detalla el flujo de trabajo típico dentro de cada sprint. El proceso inicia con una reunión diaria de Scrum, cuya duración es de 15 minutos, en la que el equipo identifica posibles bloqueos y coordina las tareas del día. A continuación, se llevan a cabo las fases principales del sprint: diseño, desarrollo y pruebas, que constituyen las actividades clave para la entrega incremental de funcionalidades. Este ciclo asegura un avance constante y la rápida detección de problemas, promoviendo la colaboración y la calidad en el producto final.

4. Monitoreo y control del proyecto

4.1. Administración de requerimientos

Durante el proceso de administración de requerimientos, se generarán los siguientes artefactos clave:

1. Historias de Usuario

- Las historias de usuario documentarán de manera simple y clara las funcionalidades que los usuarios finales (los miembros del comité de ética) necesitan. Estas historias facilitarán la comprensión y priorización de las tareas en función de su valor y urgencia, permitiendo adaptar el desarrollo a los cambios y nuevos requerimientos.

2. Épicas

- Las historias de usuario pertenecen a diferentes épicas, de esta forma se tendrá una mejor organización respecto a las etapas de desarrollo, buscando en cada sprint completar las funcionalidades de una épica.

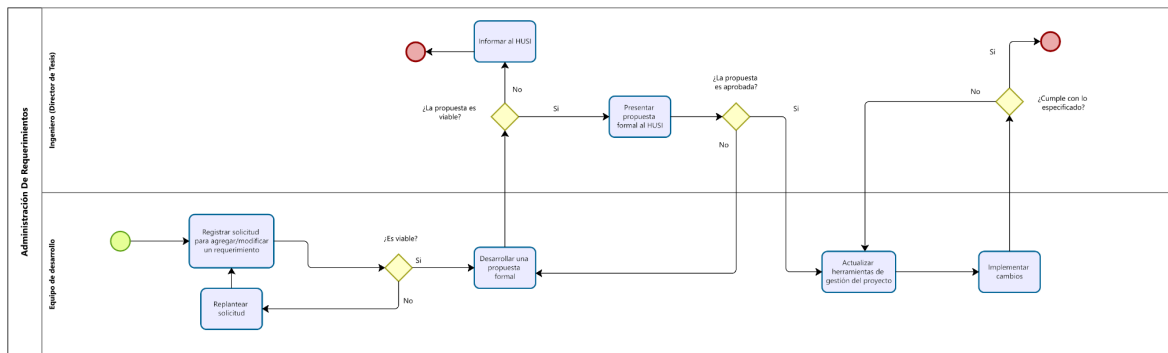


Figura 11 - BPMN Administración de Requerimientos

En cuanto al proceso de administración de requerimientos (ver anexo 2), se comenzará con la detección de nuevas necesidades o cambios por parte de los clientes o los desarrolladores. Los clientes podrán solicitar modificaciones o nuevas funcionalidades durante el desarrollo, mientras que los desarrolladores identificarán mejoras o ajustes necesarios en función del progreso del proyecto.

En general, el proceso de levantamiento de requerimientos será iterativo y su flujo de actividades estará definido de la siguiente manera:

1. Solicitud de Cambio o Nuevo Requerimiento:
 - Los clientes o desarrolladores pueden proponer nuevas historias de usuario o solicitar cambios a las historias existentes.
2. Análisis y Evaluación del Requerimiento:
 - El equipo de desarrollo analizará la viabilidad del nuevo requerimiento o cambio propuesto, evaluando su impacto en el sistema y los recursos necesarios para implementarlo. En esta fase, se actualizarán las matrices de trazabilidad para reflejar cualquier modificación.
3. Aprobación del Cambio:
 - Si el cambio es viable, se presentará una propuesta formal al director de tesis para su aprobación o rechazo, y en caso de que sea aprobada se presentará la propuesta formal al HUSI para verificar también su aprobación. Una vez aprobado, el nuevo requerimiento se incluirá en el backlog y se priorizará para su desarrollo en un sprint posterior.
4. Actualización de Artefactos:
 - Tras la aprobación del cambio, se actualizarán las historias de usuario y las matrices de trazabilidad, asegurando que todos los elementos afectados estén correctamente documentados y alineados con los objetivos del sistema.
5. Desarrollo e Implementación:
 - Los desarrolladores implementarán los cambios aprobados, siguiendo las historias de usuario actualizadas y asegurando que la nueva funcionalidad cumpla con los requerimientos definidos.

6. Verificación y Validación:

- Una vez completada la implementación, se realizará una verificación para asegurar que los cambios cumplen con las especificaciones solicitadas. El cliente revisará y validará el producto para confirmar que los nuevos requerimientos se han implementado correctamente.

Este proceso permitirá que los cambios se controlen y gestionen de manera efectiva a lo largo del desarrollo, manteniendo la calidad del sistema.

4.2. Monitoreo y control del progreso

El monitoreo del progreso del proyecto se realizará utilizando una combinación de métricas que permitan evaluar el estado del desarrollo en todo momento. Las métricas seleccionadas son las siguientes:

- *Número de requerimientos implementados*

Esta métrica hace referencia al número de historias de usuario completadas. Se considera que una historia de usuario está completa cuando ha pasado las pruebas de aceptación y ha sido validada por el cliente. La unidad de medida utilizada será la cantidad de historias de usuario completadas versus el total planificado.

- *Horas faltantes para completar una tarea*

Cada tarea estimada en un sprint tendrá un número de horas asignadas, y el equipo reportará cuántas horas faltan para su finalización. La unidad de medida utilizada serán las horas estimadas vs. horas trabajadas y restantes por tarea.

- *Velocidad del equipo*

- La velocidad se medirá en función de cuántos puntos de historia ha completado el equipo por sprint. Esta métrica ayuda a proyectar cuánto trabajo puede completarse en futuros sprints. La unidad de medida utilizada serán los puntos de historia completados por sprint.

- *Burn-down Chart*

- Se utilizará un burn-down chart para visualizar el trabajo restante en comparación con el tiempo disponible. Este gráfico mostrará el ritmo de finalización de tareas y si el equipo está avanzando según lo previsto. La unidad de medida utilizada serán las tareas completadas vs. tiempo restante en el sprint.

En cuanto al progreso, este se reportará en función de las métricas descritas y se organizará en actividades recurrentes:

1. Reuniones Semanales (Daily Scrum)

- Responsables: Todos los miembros del equipo.
- Frecuencia: 3 veces a la semana.
- Descripción: Durante estas reuniones de no más de 15 minutos, cada miembro del equipo reportará qué tareas ha completado, qué tareas está

trabajando y si existen bloqueos que impiden avanzar. Las historias de usuario completadas se marcarán como finalizadas, y las horas trabajadas y faltantes se reportarán.

- Progreso reportado: Número de historias de usuario completadas, horas trabajadas vs. horas restantes, bloqueos o retrasos identificados.

2. Revisión de Sprint

- Responsables: Todo el equipo y director.
- Frecuencia: Al final de cada sprint (cada 2 semanas).
- Descripción: En la revisión de sprint, el equipo presentará al director todas las historias de usuario completadas. El director validará si las historias han sido implementadas de manera satisfactoria.
- Progreso reportado: Validación del número de historias de usuario completadas en el sprint.

3. Revisión de Product Backlog y Ajustes de Prioridades

- Responsables: Equipo de desarrollo.
- Frecuencia: Después de cada sprint o cuando sea necesario.
- Descripción: El equipo actualizará el backlog del producto con base en las historias de usuario completadas, ajustando las prioridades según las necesidades del cliente y la velocidad del equipo. Si una historia de usuario no ha sido completada en el sprint actual, se reprogramará para un sprint futuro.
- Progreso reportado: Actualización del backlog con nuevas historias o cambios en la prioridad de las historias existentes.

En caso de que el progreso del proyecto no avance según lo esperado, se tomarán las siguientes acciones correctivas para corregir el curso y asegurar que se cumplan los plazos y objetivos:

1. Análisis de Causas de Retrasos

- Responsables: Equipo de desarrollo.
- Momento: Cuando el burn-down chart o la velocidad del equipo indiquen que no se están cumpliendo los objetivos del sprint.
- Descripción: Se realizará un análisis para identificar las causas de los retrasos. Esto puede incluir bloqueos en tareas específicas, falta de recursos o problemas con las dependencias de las tareas.
- Acción correctiva: Resolver los bloqueos prioritariamente, reasignar recursos o ajustar las dependencias entre tareas para asegurar que no se retrasen futuras historias de usuario.

2. Reasignación de Tareas

- Responsables: Equipo de desarrollo.
- Momento: Cuando se detecte que algunos miembros del equipo tienen tareas pendientes mientras otros están sobrecargados de trabajo.
- Descripción: Se revisará la asignación de tareas para asegurar que el equipo está trabajando de manera equilibrada. Las tareas se distribuirán nuevamente, si es necesario, para que todo el equipo pueda contribuir de manera eficiente al progreso del sprint.

- Acción correctiva: Distribuir nuevamente tareas entre los miembros del equipo para equilibrar la carga de trabajo y asegurar que las historias de usuario se completen según lo planificado.
3. Revisión de Estimaciones de Tiempo
- Responsables: Equipo de desarrollo.
 - Momento: Si el equipo constantemente no logra completar las historias de usuario planificadas dentro del sprint.
 - Descripción: Se analizarán las estimaciones de tiempo hechas al inicio del sprint para determinar si son realistas. Si las estimaciones resultan ser inadecuadas, se ajustarán las futuras estimaciones para reflejar con mayor precisión la capacidad del equipo.
 - Acción correctiva: Ajustar las estimaciones de tiempo para tareas futuras y planificar los sprints con más realismo.
4. Repriorización del Backlog
- Responsables: Equipo de desarrollo y director.
 - Momento: Cuando se detecte que algunos requerimientos son más urgentes que otros o si el progreso del proyecto no está alineado con las necesidades del cliente.
 - Descripción: Si se detecta que algunos requerimientos han cambiado de prioridad, se priorizará nuevamente el backlog para asegurar que las historias de usuario de mayor valor se aborden primero.
 - Acción correctiva: Priorizar nuevamente las historias de usuario en el backlog para que el equipo trabaje en las tareas más críticas primero.
5. Ajuste del Alcance
- Responsables: Equipo de desarrollo y director.
 - Momento: Si el progreso general del proyecto muestra que no se podrán cumplir todos los requerimientos dentro del plazo de los 4 meses.
 - Descripción: En caso de que se detecte que el alcance del proyecto es mayor de lo previsto, el equipo de desarrollo se reunirá con el director para discutir posibles ajustes al alcance, eliminando o posponiendo ciertas funcionalidades para asegurarse de que las más críticas se completen a tiempo.
 - Acción correctiva: Ajustar el alcance del proyecto, eliminando o posponiendo requerimientos menos críticos para cumplir con los plazos establecidos.

4.3. Cierre del proyecto

Para concluir el proyecto de manera efectiva, se realizarán las siguientes acciones:

Primero, se llevará a cabo una retrospectiva detallada de todas las fases del proyecto, en la que el equipo analizará los logros alcanzados, los obstáculos enfrentados y las lecciones aprendidas. Esta revisión permitirá generar una retroalimentación constructiva que contribuya a mejorar futuros proyectos y evaluar el desempeño global durante el desarrollo y la ejecución.

Además, se completará y entregará toda la documentación técnica y funcional requerida, que incluye la Memoria de Trabajo de Grado, el Plan de Proyecto, la Especificación de Requisitos (SRS), la Especificación de Diseño (SDD), la Propuesta de Trabajo de Grado, la Licencia de Uso, la Carta de Autorización y la Descripción de la Tesis. Todos estos documentos serán archivados en el sistema institucional de gestión documental Trazadocs, asegurando su trazabilidad y disponibilidad futura, y se le presentarán al Director, quien deberá dar el visto bueno, para poder continuar con la presentación final y así poder culminar el proyecto.

Finalmente, se realizará una presentación formal de la plataforma y los resultados del proyecto ante los jurados evaluadores, el Director de Tesis y el Comité de Ética del Hospital Universitario San Ignacio, donde se demostrará el funcionamiento completo de la herramienta y se responderán las inquietudes. Esta actividad marcará la finalización oficial y el cierre formal del proyecto.

5. Entrega del Producto

La entrega del producto final consistirá en una plataforma tecnológica avanzada diseñada para asistir al Comité de Ética del Hospital Universitario San Ignacio en la revisión y análisis de documentos de consentimiento informado. Para facilitar la implementación, el proceso de transición incluirá capacitaciones a los usuarios finales, entre ellos los evaluadores, investigadores y demás miembros del Comité de Ética, para que comprendan las funcionalidades de la plataforma y puedan operarla de manera eficiente. Estas capacitaciones se realizarán antes de la entrega final, garantizando que el personal esté preparado para utilizar el sistema sin inconvenientes.

Además, el proceso de entrega incluirá la transición de los procesos manuales y herramientas existentes hacia la nueva plataforma digital, centralizando la gestión de consentimientos informados en el sistema y eliminando las redundancias de procedimientos anteriores. Se procederá también con la inducción de la plataforma en el entorno hospitalario, siguiendo los estándares de seguridad y normativas nacionales e internacionales.

Los artefactos a entregar incluirán manuales técnicos y de usuario que detallan las funcionalidades y el mantenimiento de la plataforma, un conjunto de mockups que representan los flujos y la interfaz de usuario aprobada por el Comité, además de un paquete de contenedores que permitirá desplegar la plataforma en el momento que se requiera. Estos entregables se proporcionarán según los plazos establecidos en el cronograma, asegurando su alineación con los criterios de aceptación definidos en el Plan de Aceptación.

El equipo de desarrollo será el principal responsable de coordinar las entregas, asegurándose de que cada artefacto cumple con los estándares de calidad requeridos, bajo la supervisión del director de tesis. Para garantizar la operatividad del sistema, se llevarán a cabo pruebas de aceptación funcional y de seguridad en el entorno hospitalario, asegurando que el producto final cumpla con los objetivos de eficiencia y precisión esperados por el Comité de Ética.

6. Procesos de soporte

6.1. Ambiente de trabajo

Para asegurar un ambiente de trabajo óptimo en el desarrollo de la plataforma, el equipo ha establecido reglas claras y mecanismos de supervisión que promueven la colaboración, el respeto mutuo y el cumplimiento de los objetivos.

Microsoft Teams será la plataforma principal para la comunicación y colaboración del equipo, donde se documentará el progreso, se asignarán tareas y se supervisará el cumplimiento de las responsabilidades individuales. Además, se realizarán tres reuniones semanales para evaluar constantemente el avance y actuar oportunamente en caso de que algún integrante del equipo requiera apoyo en alguna tarea.

Las reglas de trabajo en equipo que regirán el proyecto incluyen una comunicación asertiva y constante entre todos los miembros, promoviendo un flujo de información abierto que permita detectar y resolver problemas en tiempo real. El respeto entre los integrantes es fundamental, cualquier conducta irrespetuosa que ocurra más de tres veces será discutida al interior del equipo para tomar acción. La puntualidad es igualmente importante, se espera que todos los miembros asistan a las reuniones de manera oportuna, y aquellos que acumulen cinco retrasos de más de 10 minutos deberán dialogar su situación con el equipo y el director de tesis para garantizar el compromiso de todos con el proyecto.

Cada miembro del equipo tendrá responsabilidades asignadas desde el inicio del proyecto, lo cual asegurará un reparto equitativo de las tareas. La revisión y validación del cumplimiento de estas tareas se realizará periódicamente en reuniones de seguimiento, con un enfoque en el progreso y los retos que cada fase pueda presentar. Para registrar este avance y documentar cualquier modificación, el equipo usará herramientas de gestión de proyectos en Microsoft Teams. De esta forma, tanto el equipo como el director de tesis podrán supervisar el desarrollo del proyecto y asegurar que se mantiene en línea con los plazos y estándares de calidad establecidos.

6.2. Análisis y Administración de Riesgos

La administración de riesgos será esencial durante el desarrollo de la plataforma para asegurar que el equipo de desarrollo y los stakeholders estén preparados para identificar y mitigar amenazas potenciales. Desde el inicio del proyecto, se establecerá un plan de gestión de riesgos que permita anticiparse, priorizar y manejar problemas técnicos y operativos de forma efectiva. Este proceso de identificación se llevará a cabo mediante reuniones y sesiones de análisis de riesgos con la participación del equipo de desarrollo y del director de tesis, estableciendo un enfoque colaborativo para la evaluación de riesgos.

Los riesgos más críticos incluyen la integración de la plataforma, la protección de información confidencial, y la capacidad del sistema de inteligencia artificial para adaptarse a la estructura y variabilidad de los documentos de consentimiento informados. Para cada uno, se priorizarán acciones preventivas y de mitigación según su probabilidad e impacto, lo cual permitirá asignar los recursos de manera eficiente y evitar sobrecargas innecesarias. Las revisiones de riesgos estarán programadas al cierre de cada fase y previo a cada iteración importante, donde se actualizará una tabla de riesgos que indique probabilidad de ocurrencia e impacto, clasificados por prioridad, revisando las acciones y recursos asignados.

Entre las estrategias preventivas destaca el uso de control de versiones, empleando ramas separadas para el desarrollo de funcionalidades que puedan afectar la operación general del sistema. En cuanto a la seguridad de la información, se implementarán estándares de encriptación de contraseñas y protocolos diseñados para garantizar el cumplimiento de las regulaciones de protección de datos vigentes en Colombia.

A continuación, se presentará el BPMN que se realizó, para modelar como sería este proceso de mitigación de riesgos:

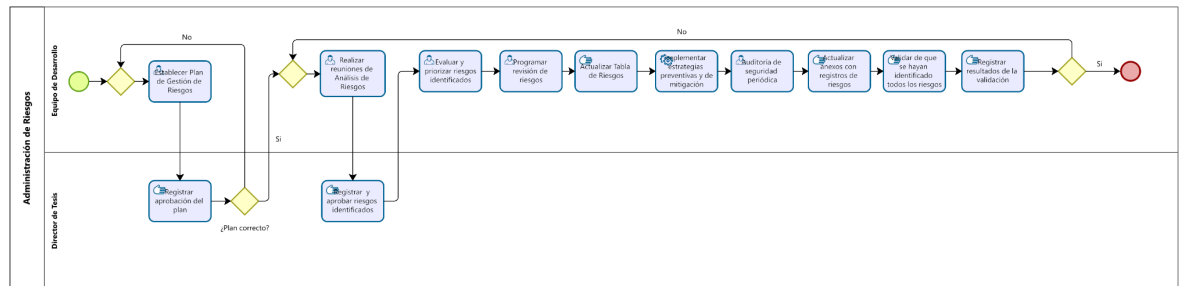


Figura 12 - BPMN Administración de Riesgos

Este diagrama BPMN representa la Administración de Riesgos en el desarrollo de la plataforma para el Comité de Ética del Hospital Universitario San Ignacio (ver anexo 8). En este proceso participan dos roles principales: el Equipo de Desarrollo y el Director de Tesis. El objetivo de esta secuencia es identificar, evaluar y mitigar los riesgos que podrían impactar la funcionalidad, seguridad y estabilidad de la plataforma.

El proceso inicia con la tarea Establecer Plan de Gestión de Riesgos, llevada a cabo por el Equipo de Desarrollo, donde se define una estrategia detallada para manejar riesgos técnicos y operativos. Esta tarea utiliza como artefactos un documento formal del Plan de Riesgos y se apoya en herramientas de gestión como Microsoft Excel o Google Sheets para registrar y hacer seguimiento a los riesgos.

Una vez finalizado, el Director de Tesis revisa el plan en la actividad Registrar aprobación del plan. Si el plan no cumple con los estándares necesarios, el flujo regresa a la primera tarea para realizar ajustes. En caso de aprobación, se procede a la tarea Realizar reuniones de análisis de riesgos, en la cual el equipo de desarrollo, en colaboración con el Director de Tesis, identifica los posibles riesgos. Esta tarea está soportada por herramientas de colaboración como Microsoft Teams para llevar a cabo las reuniones y discutir los riesgos identificados.

A continuación, en evaluar y priorizar riesgos identificados, los riesgos son clasificados por su probabilidad e impacto utilizando una matriz de riesgos como artefacto clave, para luego planificar su mitigación de acuerdo con su criticidad. Luego, en programar revisión de riesgos, se establecen puntos de control periódicos donde se revisará el estado de los riesgos, utilizando herramientas para programar dichas revisiones en el cronograma del proyecto. Posteriormente, en Actualizar Tabla de Riesgos, se registran todos los riesgos y sus detalles en la tabla de riesgos del proyecto, que actúa como artefacto central en este proceso.

Para mitigar los riesgos, se implementan en la actividad Implementar estrategias preventivas y de mitigación, que incluye pruebas en entornos simulados y aplicación de medidas de encriptación para la seguridad de los datos.

Finalmente, el proceso culmina en una decisión en la que se evalúa si todos los riesgos han sido gestionados adecuadamente. Si no es así, el flujo regresa a la auditoría de seguridad para implementar las medidas necesarias; en caso de que todos los riesgos estén bajo control, la última actividad, Registrar resultados de la validación, documenta el estado final de los riesgos, asegurando que la plataforma cumple con los requisitos de seguridad y estabilidad.

Los artefactos generados, como el Plan de Riesgos, la Tabla de Riesgos y los anexos de riesgos, se mantienen actualizados en cada etapa del proceso y proporcionan un marco sólido para la toma de decisiones y el control de riesgos a lo largo del desarrollo de la plataforma.

A continuación, se presentan la tabla de riesgos identificados en el proyecto y la tabla con los riesgos más críticos, que destacan las acciones de prevención y mitigación para cada uno.

Tabla con los riesgos que han sido identificados:

ID del Riesgo	Descripción del Riesgo	Probabilidad (1-5)	Impacto (1-5)	Prioridad (Probabilidad x Impacto)	Responsable
R1	Problemas de integración	4	5	20	Equipo de Desarrollo
R2	Falla de seguridad en protección de datos sensibles	5	5	25	Equipo de Desarrollo
R3	Incompatibilidad del modelo de IA con documentos específicos	3	4	12	Equipo de Desarrollo
R4	Retrasos en las entregas debido a cambios en los requisitos	3	5	15	Director
R5	Falta de habilidades técnicas para implementar el sistema	2	4	8	Director

Tabla 5. Riesgos identificados

Tabla con los riesgos más importantes:

ID del Riesgo	Acción de Prevención/Mitigación	Descripción de la Acción	Recursos Necesarios	Responsable	Fecha de Revisión
R1	Implementación de encriptación y auditorías de seguridad	Aplicar estándares de encriptación y realizar auditorías periódicas de seguridad	Herramientas de Seguridad	Equipo de desarrollo	Semestral
R2	Pruebas de integración de modelo	Crear entornos simulados para probar la compatibilidad y estabilidad de integración del modelo	Infraestructura de Pruebas	Equipo de Desarrollo	Mensual
R3	Reuniones de revisión y ajuste de alcance con stakeholders	Realizar reuniones frecuentes para revisar requisitos y ajustar el plan de desarrollo	Herramientas de Gestión	Director	Quincenal

Tabla 6. Riesgos más importantes

6.3. Administración de configuración y Documentación

La administración del código se gestionará mediante Git, utilizando GitHub como repositorio. La metodología que tendremos para desarrollar será mediante el uso de ramas específicas para cada funcionalidad, conocidas como feature branches (ramas de características).

Al finalizar cada sprint, estas ramas se fusionarán (merge) con la rama principal (main), garantizando que el código principal se mantenga estable y actualizado con las últimas funcionalidades aprobadas. Esta estrategia nos permitirá un desarrollo colaborativo y organizado, facilitando la resolución de conflictos y la integración continua de nuevas características sin comprometer la estabilidad del código existente.

Por otro lado, la documentación tanto de los requisitos funcionales y no funcionales del sistema, como de los aspectos técnicos y de diseño, se gestionará con herramientas como Google Docs, ClickUp y Figma. Esta documentación incluirá manuales de usuario, guías de mantenimiento, diagramas de arquitectura y mockups de la interfaz de usuario, y se actualizará de manera continua a lo largo del proyecto, reflejando los cambios implementados en la plataforma. Otra característica de estas herramientas es que nos permite colaboración en tiempo real, lo cual nos permitirá realizar ajustes conforme se identifiquen nuevos requisitos o modificaciones tras las revisiones con el Comité de Ética.

A continuación, se presentará el BPMN que se realizó, para modelar como sería el proceso de control de cambios:

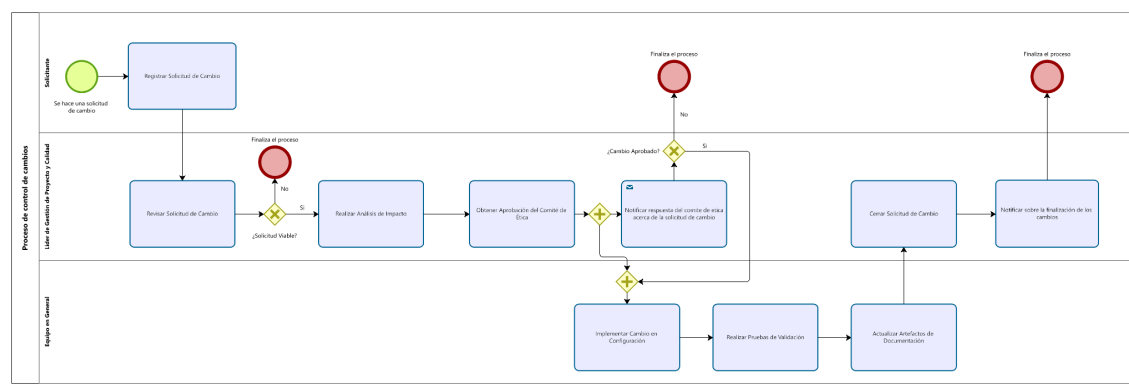


Figura 13 - BPMN Proceso de control de cambios

Este diagrama BPMN (ver Anexo 10) representa el Proceso de Control de Cambios en un sistema de gestión, específicamente diseñado para coordinar las solicitudes de cambio que surgen durante la ejecución de un proyecto. En este proceso participan tres roles principales: el Solicitante, el Líder de Gestión de Proceso y Calidad (Que en nuestro caso sería el Director), y el Equipo en General.

El proceso inicia cuando el Solicitante realiza una solicitud de cambio. Esta acción desencadena la primera tarea, Registrar Solicitud de Cambio, ejecutada por el Líder de Gestión de Proceso y Calidad. Una vez registrada la solicitud, se pasa a la actividad de Revisar Solicitud de Cambio para determinar su viabilidad.

En este punto, el flujo tiene dos opciones. Si la solicitud no es viable, el proceso finaliza. En caso de que sí lo sea, se procede a Realizar Análisis de Impacto. Este análisis permite evaluar los efectos potenciales del cambio en el proyecto. Posteriormente, se requiere Obtener Aprobación del Comité de Ética, asegurando que la solicitud cumple con los estándares éticos y de calidad establecidos.

Tras la revisión del Comité, surge una nueva decisión: ¿Cambio Aprobado?. Si el cambio no es aprobado, el proceso finaliza con la notificación al solicitante sobre la negativa. Si es aprobado, se envía una notificación al solicitante sobre la aprobación de la solicitud de cambio, permitiendo avanzar hacia la implementación.

El Equipo en General asume la tarea de Implementar Cambio en Configuración, donde se efectúan las modificaciones necesarias en el sistema. Una vez implementado el cambio, se realizan Pruebas de Validación para asegurar que el cambio funcione correctamente y no afecte otras áreas del sistema.

Posteriormente, se actualizan los artefactos de documentación para reflejar el cambio implementado y mantener la trazabilidad del proceso. Finalmente, el Líder de Gestión de Proceso y Calidad procede a Cerrar Solicitud de Cambio y Notificar sobre la Finalización de los Cambios, concluyendo así el proceso de control de cambios.

Tabla de artefactos

La siguiente tabla presenta una descripción detallada de los principales artefactos de documentación y código generados a lo largo del proyecto, así como el momento en el que son creados o refinados durante las diferentes fases y sprints. Esta organización permite visualizar claramente cómo se gestiona la producción y actualización de cada entregable, asegurando un desarrollo ágil, ordenado y alineado con los objetivos y necesidades del Comité de Ética y el Director de Tesis.

Artefacto	Descripción	Momento de Creación/Refinamiento
Código Fuente	Todo el código desarrollado para la plataforma web, incluyendo scripts y pruebas unitarias.	Creación en el inicio del proyecto y refinamiento en cada sprint.
Documentación de Requisitos	Detalles de los requisitos funcionales y no funcionales del sistema.	Creación durante la fase de Comprensión del Negocio y refinamiento continuo.
Documentación Técnica y Funcional	Manuales de usuario, guías de mantenimiento y diagramas de arquitectura.	Creación durante el desarrollo y actualización continua.
Mockups de la Interfaz de Usuario	Representaciones visuales de la interfaz de usuario para validar diseño y funcionalidad.	Creación inicial y refinamiento tras cada revisión con el Comité de Ética.
Requerimientos de Sprint	Lista de historias de usuario y tareas a completar durante cada sprint.	Definición durante la reunión de planeación del sprint.

Tabla 7. Artefactos

6.4. Métricas y proceso de Medición

El propósito de este apartado es definir las métricas clave que se utilizarán durante el proyecto para evaluar el progreso y la calidad del desarrollo. Las métricas permitirán al equipo tomar decisiones informadas y ajustar los planes de trabajo de acuerdo con el rendimiento y las necesidades detectadas.

Métricas Utilizadas

A continuación, se enumeran las métricas que serán empleadas a lo largo del proyecto, y las áreas en las que serán relevantes:

1. Velocidad del equipo (Ver sección 4.2 *"Monitoreo y Control del Progreso"*)
 - *Relacionada con:* Planificación de sprints y asignación de tareas.
 - *Descripción:* La velocidad del equipo se mide en función de los puntos de historia completados por sprint. Esta métrica ayuda a proyectar la cantidad de trabajo que se puede realizar en futuras iteraciones.
2. Cumplimiento de hitos y entregables (Ver sección 1.5 *"Resumen de Calendarización y Presupuesto"*)
 - *Relacionada con:* El seguimiento del cronograma del proyecto.
 - *Descripción:* Se mide el número de entregables completados según lo planificado. Esto permite evaluar el cumplimiento del cronograma y ajustar las prioridades de las tareas.
3. Defectos detectados en pruebas (Ver sección 4.1 *"Administración de Requerimientos"*)
 - *Relacionada con:* Plan de pruebas de calidad.

- *Descripción:* Métrica que recoge el número de errores o defectos encontrados en las pruebas de integración, carga, end-to-end y de usuario. Se busca reducir este número progresivamente conforme avanza el proyecto.
- 4. Satisfacción de los usuarios finales (Ver sección 2.4.1 "*Interfaces Externas*")
 - *Relacionada con:* Evaluación de la interfaz y funcionalidad de la plataforma.
 - *Descripción:* Se recolectarán datos sobre la satisfacción del Comité de Ética con el uso de la plataforma, evaluando aspectos como usabilidad, efectividad y precisión de los resultados generados por la IA.

Proceso de Recolección y Consolidación de Métricas

Para cada métrica, se describen los pasos para su recolección, consolidación y uso en el proyecto:

1. Velocidad del equipo
 - *Quién la recolecta:* El líder de Gestión del proyecto será responsable de recolectar esta métrica.
 - *Cuándo:* Al final de cada sprint durante la reunión de revisión de sprint.
 - *Herramientas:* ClickUp, donde se registran los puntos de historia completados.
 - *Procesamiento y consolidación:* El líder consolida la información generada por las herramientas de gestión, calculando la velocidad media del equipo para los siguientes sprints.
 - *Uso en el proyecto:* La velocidad del equipo se utiliza para planificar el trabajo futuro, ajustando el backlog y las prioridades de acuerdo con la capacidad de entrega.
2. Cumplimiento de hitos y entregables
 - *Quién la recolecta:* El líder de Gestión del proyecto será el encargado de hacer seguimiento a los entregables.
 - *Cuándo:* Al final de cada fase clave del proyecto, coincidiendo con las revisiones de los hitos del cronograma.
 - *Herramientas:* Calendario de proyecto en ClickUp o cualquier herramienta similar de gestión de proyectos.
 - *Procesamiento y consolidación:* Se realiza un seguimiento de los entregables planificados vs. completados, y se consolidan en un reporte mensual de progreso.
 - *Uso en el proyecto:* Esta métrica ayuda a mantener el cronograma bajo control y ajustar las asignaciones de tareas o recursos si se detectan retrasos.
3. Defectos detectados en pruebas
 - *Quién la recolecta:* Los miembros del equipo que realizan las pruebas, con supervisión del Líder de Desarrollo de Software.
 - *Cuándo:* Durante las pruebas de integración, carga, end-to-end y de usuario.
 - *Herramientas:* Se utilizarán Jest para pruebas de integración, Playwright para pruebas end-to-end (E2E) y Artillery para pruebas de carga,

- garantizando así la calidad, funcionalidad y rendimiento de la plataforma.
 - Procesamiento y consolidación: Los defectos detectados se categorizan y priorizan según su severidad. Se consolida un reporte por sprint con las métricas de calidad.
 - Uso en el proyecto: Esta métrica se utiliza para ajustar el enfoque de desarrollo, asegurando que los errores críticos se resuelvan antes de avanzar a la siguiente fase.
4. Satisfacción de los usuarios finales
- Quién la recolecta: Líder de Frontend y Experiencia de Usuario.
 - Cuándo: Al finalizar las pruebas de usuario (User Acceptance Testing - UAT) y después del despliegue de la plataforma.
 - Herramientas utilizadas: Reuniones presenciales o virtuales en las que se presenta la plataforma a los usuarios finales, incluyendo el Comité de Ética.
 - Procesamiento y consolidación: Los comentarios y sugerencias recogidos durante estas reuniones serán compilados en un informe de satisfacción que incluirá propuestas de mejora.
 - Uso en el proyecto: Esta métrica permite identificar oportunidades para optimizar la usabilidad y funcionalidad de la plataforma, enfocándose en ajustar las áreas críticas que impacten la experiencia del usuario final.

6.5. Control de calidad

El control de calidad asegura que cada entregable del proyecto cumpla con los requisitos de calidad previamente establecidos, alineándose con los objetivos del Comité de Ética y mejorando la confiabilidad del sistema final.

Actividades de Control de Calidad

1. Verificación y Validación:
Se realizarán revisiones continuas del código y de los artefactos desarrollados para asegurar que cumplen con los estándares de calidad y funcionalidad. Esto incluirá pruebas automatizadas de integración con Jest, pruebas end-to-end (E2E) utilizando Playwright y pruebas de carga con Artillery. Además, se llevarán a cabo pruebas de aceptación por parte de los usuarios finales (UAT) para validar el cumplimiento de los requisitos establecidos por el Comité de Ética.
2. Métricas de Calidad:
 - Defectos Detectados en Pruebas: Monitoreo de errores en cada sprint para ajustarlos antes de la siguiente fase.
 - Velocidad del Equipo: Evaluación de la capacidad de trabajo del equipo en cada iteración para planificar de manera adecuada los siguientes sprints.
 - Cumplimiento de Hitos: Seguimiento del progreso en entregables clave para asegurar que se cumplen los plazos y objetivos del cronograma:

- Revisión de cada entrega con el Comité de Ética, ajustando cualquier aspecto que no cumpla con los estándares o expectativas.
3. Documentación Continua:
- Todas las actividades y resultados se documentarán en herramientas de gestión como ClickUp, y se actualizarán los informes de progreso, lo que permite tomar decisiones de ajuste en función del rendimiento del equipo y del sistema.

7. Anexos

1. Diagramas BPMN

 BPMN

2. Diagrama BPMN: Administración de requerimientos

 AdiministracionRequerimientos.png

3. Diagrama BPMN: Ciclo de Vida

 CiclodeVida.png

4. Diagrama BPMN: Cierre de Proyecto

 CierreProyecto.png

5. Diagrama BPMN: Desarrollo iterativo

 DesarrolloIterativo.png

6. Diagrama BPMN: Levantamiento de Requerimiento

 LevantamientoRequerimiento.png

7. Diagrama BPMN: Pruebas Funcionales

 PruebasFuncionales.png

8. Diagrama BPMN: Mitigación

 Mitigacion.png


9. Diagrama BPMN: Integración Modelo de IA

 IntegracionIA.png

10. Diagrama BPMN: Proceso de Control de Cambios

 Proceso de control de cambios.png

11. Diagrama BPMN: Scrum

 Scrum.png

12. Diagrama BPMN: Sprint

 Scrum.png

13. Diagrama WBS:

 WBS.png

8. 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», en *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». Accedido: 10 de noviembre de 2024. [En línea]. Disponible en: <https://nodejs.org/en>
- [24] TypeScript. (s.f.). TypeScript: Typed JavaScript at Any Scale. Recuperado de <https://www.typescriptlang.org/>
- [25] Python Software Foundation. (s.f.). Python. Recuperado de <https://www.python.org/>
- [26] Angular. (s.f.). One framework. Mobile & desktop. Recuperado de <https://angular.io/>
- [27] React. (s.f.). A JavaScript library for building user interfaces. Recuperado de <https://reactjs.org/>
- [28] Express.js. (s.f.). Express - Node.js web application framework. Recuperado de <https://expressjs.com/>
- [29] Jest. (s.f.). Delightful JavaScript Testing. Recuperado de <https://jestjs.io/>
- [30] MongoDB. (s.f.). MongoDB: The application data platform. Recuperado de <https://www.mongodb.com/>
- [31] Git. (s.f.). Git. Recuperado de <https://git-scm.com/>
- [32] GitHub. (s.f.). GitHub: Where the world builds software. Recuperado de <https://github.com/>
- [33] Visual Studio Code. (s.f.). Code editing. Redefined. Recuperado de <https://code.visualstudio.com/>
- [34] Visual Paradigm. (s.f.). Visual Paradigm - UML, Agile, PMBOK, TOGAF, BPMN and More. Recuperado de <https://www.visual-paradigm.com/>
- [35] Microsoft Teams. (s.f.). Microsoft Teams. Recuperado de <https://www.microsoft.com/es-co/microsoft-teams/group-chat-software>
- [36] Microsoft Outlook. (s.f.). Outlook – free personal email and calendar from Microsoft. Recuperado de <https://outlook.live.com/>
- [37] Discord. (s.f.). Discord | Your Place to Talk and Hang Out. Recuperado de <https://discord.com/>
- [38] WhatsApp. (s.f.). WhatsApp. Recuperado de <https://www.whatsapp.com/>
- [39] Google Docs. (s.f.). Google Docs: Free Online Documents for Personal Use. Recuperado de <https://www.google.com/docs/about/>
- [40] ClickUp. (s.f.). One app to replace them all. Recuperado de <https://clickup.com/>
- [41] Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient Estimation of Word Representations in Vector Space. arXiv preprint arXiv:1301.3781.
- [42] Le, Q., & Mikolov, T. (2014). Distributed Representations of Sentences and Documents. In *Proceedings of the 31st International Conference on Machine Learning* (pp. 1188–1196).

- [43] Friedl, J. E. F. (2006). Mastering Regular Expressions (3rd ed.). O'Reilly Media.
- [44] Jurafsky, D., & Martin, J. H. (2009). Speech and Language Processing (2nd ed.). Prentice Hall.
- [45] Google Cloud. (s.f.). Document AI. Recuperado de <https://cloud.google.com/document-ai>
- [46] Amazon Web Services. (s.f.). Amazon Textract. Recuperado de <https://aws.amazon.com/textract/>
- [47] IBM. (s.f.). Watson Discovery. Recuperado de <https://www.ibm.com/cloud/watson-discovery>
- [48] Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep Learning. MIT Press.
- [49] Chollet, F. (2018). Deep Learning with Python. Manning Publications.
- [50] Artillery. (2023). Artillery: Modern load testing and functional testing. <https://artillery.io/docs/>
- [51] Microsoft. (2023). Playwright: Fast and reliable end-to-end testing for modern web apps. <https://playwright.dev/docs/intro>