

UNIVERSIDAD PRIVADA FRANZ TAMAYO
FACULTAD DE INGENIERÍA
CARRERA INGENIERIA DE SISTEMAS



**"DESARROLLO DE UNA
MULTIPLATAFORMA PARA EL
FORTALECIMIENTO DE LA PRESENCIA
DIGITAL Y EL SEGUIMIENTO DE
BENEFICIARIOS DE LA FUNDACIÓN
ONCOFELIZ"**

**PROYECTO DE GRADO PARA OPTAR
AL TÍTULO ACADÉMICO DE LICENCIATURA
EN INGENIERÍA DE SISTEMAS**

POSTULANTE: Jhoselin Diana Cespedes Brañez

TUTOR: Ing. Fernando Hinojosa Sánchez

Cochabamba – Bolivia

2025

ÍNDICE DE CONTENIDO

Resumen de trabajo.....	1
Generalidades	2
Antecedentes	2
Descripción del objeto de estudio	3
Diseño técnico de la investigación.....	3
Planteamiento del problema.....	3
Formulación del problema.....	5
Objetivo general.....	5
Objetivos específicos.....	5
Definición de variables	5
Justificación	6
Justificación social	6
Justificación técnica	6
Justificación económica	6
Metodología y técnicas de la investigación	7
Tipo de investigación	7
Alcance de la investigación	8
Diseño de la investigación	8
Técnicas e instrumentación de recolección de datos	8
Fuentes de información	9
Alcances y aportes.....	9
Alcance espacial	9
Alcance temporal.....	9
Alcance temático	10
Límites	10
CAPÍTULO I.....	11
MARCO TEÓRICO.....	11
1.1. Transformación digital en organizaciones sin fines de lucro	11
1.1.1 Impacto de la digitalización en organizaciones no lucrativas	11
1.1.2 Plataformas digitales y su rol en la captación de donaciones	12
1.1.3 Beneficios de la digitalización en la gestión de recursos y comunicación	13
1.1.4. Altos costos, barreras de acceso y rol de las ONG en el cáncer infantil en Bolivia.....	13
1.2. Sistemas de notificaciones automáticas	14
1.2.1. Beneficios de los sistemas de notificación	15
1.2.2. Impacto de la adherencia al tratamiento	15

1.2.3. Automatización de procesos en organizaciones benéficas	15
1.2.4. Tecnologías para la implementación de notificaciones push y correo electrónico.....	16
1.3. Usabilidad y accesibilidad en plataformas digitales	16
1.3.1. Diseño centrado en el usuario (DCU)	17
1.3.2. Accesibilidad web y directrices WCAG	18
1.3.3. Diseño Inclusivo en plataformas digitales	19
1.3.4. Principios de diseño de interfaces intuitivas y amigables	19
1.4. Seguridad y gestión de datos	20
1.4.1. Protección de datos personales y privacidad	20
1.4.2. Normativas legales sobre la protección de datos (GDPR, Ley de Protección de Datos Personales)	20
1.4.3. Gestión segura de bases de datos	21
1.4.4. Estrategias para el cifrado y protección de la información	22
1.5. Evaluación del impacto	23
1.5.1. Método de evaluación de impacto en plataformas digitales	23
1.5.2. Pruebas de usuario y retroalimentación continua para mejora continua	23
1.5.3. Análisis de satisfacción del usuario y pruebas de usabilidad	23
1.6. Integración de herramientas digitales y asistentes virtuales de IA	24
1.6.1. Asistentes virtuales basados en inteligencia artificial.....	24
1.7. Ingeniería de software aplicada al proyecto	25
1.7.1. Fundamentos de la ingeniería de software	25
1.7.2. Metodologías de desarrollo de software	25
1.7.3. Análisis de requerimientos del sistema	26
1.7.4. Diagramas UML.....	26
1.8. Arquitectura y diseño del sistema.....	28
1.8.1. Arquitectura de software	28
1.8.1.1. Arquitectura cliente – servidor	28
1.8.2. Diseño de la base de datos.....	29
1.8.3. Herramientas y tecnologías utilizadas	30
1.8.4. Comunicación con APIs	30
1.9. Inteligencia artificial y aprendizaje automatizado	30
1.9.1. Modelado de lenguaje natural	31
1.9.2. OpenAI y ChatGPT	31
1.9.3. Asistentes virtuales	31
1.9.4. Bots conversacionales personalizados	31
1.10. Tecnologías de desarrollo para plataformas web y móviles	32
1.10.1. Desarrollo web	33

1.10.2. Desarrollo móvil	34
1.10.3. Backend y bases de datos	34
1.11. Optimización para la experiencia del usuario.....	37
1.11.1. Principios para la experiencia del usuario en plataformas digitales	37
1.11.3. Técnicas de diseño responsive y adaptabilidad a diferentes dispositivos	37
CAPÍTULO II	38
INGENIERÍA DEL PROYECTO	38
2.1. Metodología del desarrollo.....	38
2.2. Requerimientos.....	40
2.2.1. Requerimientos funcionales (RF)	40
2.2.2. Requerimientos no funcionales (NF)	54
2.3. Diseño del proyecto.....	57
2.3.1. Arquitectura del software (Cliente - Servidor)	57
2.3.2. Diagramas UML.....	58
2.4. Desarrollo	67
2.4.1. Tecnologías de desarrollo	67
2.4.2. Implementación del backend	68
2.4.3. Implementación del frontend	70
2.4.4. Integración de herramientas con inteligencia artificial (IA)	72
2.4.5. Interfaces	75
2.5. Pruebas	81
2.5.1. Pruebas de interacción con la base de datos	81
2.5.2. Pruebas de integración con APIs externas	82
2.5.3. Pruebas de caja negra	84
2.6.4. Pruebas de usabilidad con usuarios	91
CAPÍTULO III	93
COSTOS	93
3.1. Recolección de datos	93
3.1.1. Tamaño del proyecto (KLOC)	93
3.1.2. Modo de desarrollo del proyecto	94
3.1.3. Factores de escalabilidad.....	94
3.1.4. Factores de costos.....	95
3.2. Cálculo de escalabilidad ajustada	96
3.3. Cálculo del esfuerzo	96
3.4. Cálculo del esfuerzo y tiempo	97
3.5. Presupuesto estimado del proyecto	97
CAPÍTULO IV	98

CONCLUSIONES Y RECOMENDACIONES	98
4.1. Conclusiones	98
4.2. Recomendaciones	99
ABREVIATURAS	100
GLOSARIO	101
API REST	101
Arquitectura Cliente-Servidor	101
Backend	101
COCOMO II.....	101
Frontend	101
Inteligencia artificial.....	101
NFC	101
Node.js	102
ORM	102
PostgreSQL.....	102
SCRUM.....	102
SUS	102
Token JWT.....	102
UI.....	102
APÉNDICES.....	103
Apéndice A: Cuestionario y resultados cuantitativos	103
A.1. Preguntas de la encuesta dirigida al público en general.....	103
A.2. Resultados gráficos de la encuesta	106
Sección 1: Perfil del encuestado	106
Sección 2: Captación de fondos y donaciones.....	107
Sección 3: Seguimiento y apoyo a las familias	109
BIBLIOGRAFÍA	112

ÍNDICE DE FIGURAS

Figura 1. Árbol de problemas	4
Figura 2. Plataformas digitales y su rol en la captación de donaciones	12
Figura 3. Ilustración de notificación push en dispositivo móvil	14
Figura 4. Etapas del diseño DCU	17
Figura 5. Principios de accesibilidad WCAG.....	19
Figura 6. Tipos de diagramas UML	27
Figura 7. Esquema arquitectura cliente - servidor	28
Figura 8. Esquema arquitectura microservicios	29
Figura 9. Ilustración prisma como intermediario en el desarrollo de aplicaciones con bases de datos modernas	35
Figura 10. API web service.....	36
Figura 11. Modelo arquitectura cliente servidor	58
Figura 12. Diagrama de casos de uso usuario anónimo.....	59
Figura 13. Diagrama de casos de uso administrador.....	59
Figura 14. Diagrama de casos de uso psicólogo	60
Figura 15. Diagrama de casos de uso trabajador social.....	60
Figura 16. Diagrama de casos de uso beneficiario	61
Figura 17. Diagrama de despliegue	62
Figura 18. Diagrama de paquetes backend	63
Figura 19. Diagrama de paquetes frontend	63
Figura 20. Diagrama de componentes	64
Figura 21. Diagrama de clases	66
Figura 22. Diseño Entidad Relación	67
Figura 23. Estructura backend.....	69
Figura 24. Estructura backend.....	70
Figura 25. Estructura frontend	71
Figura 26. Estructura frontend	72
Figura 27. Petición al modelo (prompt y parámetros)	73

Figura 28. Procesamiento de la respuesta en el frontend	73
Figura 29. Acción personalizada para interactuar con GPT	73
Figura 30. Organización de archivos principales del modelo de IA Rasa	74
Figura 31. Configuración de respuestas automáticas en el archivo domain.yml	75
Figura 32. Entrenamiento de intenciones de saludo en el archivo nlu.yml	75
Figura 33. Inicio de sesión web administrativa	76
Figura 34. Panel administrativo de la plataforma web con opciones de gestión de usuarios, eventos y contenidos.....	76
Figura 35. Interfaz del asistente virtual para apoyar en la creación y personalización de contenidos digitales	77
Figura 36. Vista informativa de la plataforma web con acceso a secciones de noticias, actividades y servicios disponibles	78
Figura 37. Acceso público a campañas activas y visualización del perfil informativo de cada actividad.....	78
Figura 38. Canales de contacto y campañas, consultas y soporte en línea disponibles en la vista informativa de la plataforma	79
Figura 39. Formulario de donación	79
Figura 40. Chatbot web informativa	80
Figura 41. Vista general de la aplicación móvil con funciones orientadas a usuarios, incluyendo notificaciones y seguimiento	81
Figura 42. Obtener eventos registrados	82
Figura 43. Registrar nuevo usuario	82
Figura 44. Eliminar usuario por ID	82
Figura 45. Prueba en postman publicación en Facebook.....	83
Figura 46. Prueba Postman Api de OpenRouter	84
Figura 47. Pregunta 1: Indique su edad.....	106
Figura 48. Pregunta 2: ¿Con que frecuencia interactúa con la fundación OncoFeliz?	106
Figura 49. Pregunta 3: ¿Como conoció la Fundación OncoFeliz?	107
Figura 50. Pregunta 4: Si ha realizado donaciones, ¿Que método ha utilizado? ...	107

Figura 51. Pregunta 5: ¿Como calificaría la facilidad del proceso actual para donar a la Fundación?	108
Figura 52. Pregunta 6: ¿Estaría dispuesto a través de una plataforma digital con opciones de pago seguras?	108
Figura 53. Pregunta 7: ¿Como recibe información sobre la disponibilidad de apoyo y medicamentos?	109
Figura 54. Pregunta 8: ¿Ha tenido dificultades para acceder a medicamentos o apoyo de la fundación?	109
Figura 55. Pregunta 9: ¿Le gustaría recibir notificaciones y recordatorios sobre el tratamiento de su hijo /a mediante una aplicación móvil?	110
Figura 56. Pregunta 10: ¿Cree que una plataforma digital ayudaría a mejorar la gestión y transparencia de la fundación?	110
Figura 57. Pregunta 11: ¿Que funcionalidades considera más importantes en una plataforma digital?.....	111

ÍNDICE DE TABLAS

Tabla 1. Definición de variables	5
Tabla 2. Conceptos clave de los diagramas entidad-relación.....	30
Tabla 3. Aspectos clave de RASA en el desarrollo de chatbots	32
Tabla 4. Roles del equipo SCRUM.....	38
Tabla 5. Artefactos del proceso SCRUM	39
Tabla 6. Cronograma de sprints.....	39
Tabla 7. Trazabilidad entre objetivos, requerimientos y módulos	40
Tabla 8. RF1 Mostrar barra de navegación web informativa.....	41
Tabla 9. RF2 publicar artículos del blog	41
Tabla 10. RF3 visualizar eventos.....	42
Tabla 11. RF4 exponer casos recuperados	42
Tabla 12. RF5 informar sobre campañas	42
Tabla 13. RF6 ofrecer formulario de contacto	43
Tabla 14. RF7 incluir pie de página	43
Tabla 15. RF1 gestión de eventos	44
Tabla 16. RF2 exportar eventos.....	45
Tabla 17. RF3 gestión de campañas	45
Tabla 18. RF4 exportar campañas.....	46
Tabla 19. RF5 publicar noticia	46
Tabla 20. RF6 gestión de noticias	46
Tabla 21. RF7 registrar usuario	47
Tabla 22. RF8 registrar usuario por roles	47
Tabla 23. RF9 gestión de usuarios	47
Tabla 24. RF10 cambiar contraseña	48
Tabla 25. RF11 habilitar donación con QR	48
Tabla 26. RF12 ver historial de donaciones	48
Tabla 27. RF13 exportar donaciones a Excel/PDF.....	49
Tabla 28. RF14 registrar paciente	49

Tabla 29. RF15 gestión de pacientes	50
Tabla 30. RF16 registrar cita de psicólogo.....	50
Tabla 31. RF17 registrar historial de tratamiento	50
Tabla 32. RF18 registrar expediente social	51
Tabla 33. RF19 registrar evaluación de vulnerabilidad	51
Tabla 34. RF20 registrar derivación.....	51
Tabla 35. RF21 registrar donación a paciente	52
Tabla 36. RF22 visualización del historial clínico	52
Tabla 37. RF23 calendario de citas.....	53
Tabla 38. RF24 notificaciones de citas	53
Tabla 39. RF25 chat interactivo	53
Tabla 40. RNF - 01	54
Tabla 41. RNF - 02	54
Tabla 42. RNF - 03	54
Tabla 43. RNF - 04	55
Tabla 44. RNF - 05	55
Tabla 45. RNF - 05	55
Tabla 46. RNF - 07	56
Tabla 47. RNF - 08.....	56
Tabla 48. RNF - 09	56
Tabla 49. Stack tecnológico del sistema.....	68
Tabla 50. Pruebas de caja negra.....	85
Tabla 51. Encuestas de usabilidad	91
Tabla 52. Indicadores de usabilidad	92
Tabla 53. KLOCK backend.....	93
Tabla 54. KLOCK frontend	94
Tabla 55. Factores de escalabilidad	95
Tabla 56. Factores de costos	95
Tabla 57. Total, presupuesto.....	97

Tabla 58. Abreviaturas y siglas..... 100

Resumen de trabajo

El proyecto tiene como objetivo el desarrollo de una plataforma web y móvil para fortalecer la presencia digital de la fundación OncoFeliz en el departamento de Cochabamba, Bolivia, con esta solución se busca optimizar las estrategias de captación de recursos al facilitar las donaciones en línea y sistematizar el seguimiento de los beneficiarios, reemplazando los procesos manuales.

Para la implementación del sistema se realizó un análisis de las necesidades mediante encuestas y entrevistas, también se realizó un análisis detallado de los requerimientos considerando las necesidades de OncoFeliz. Se diseñó una plataforma web con acceso por roles que permite gestionar información de beneficiarios, campañas y donaciones de forma ordenada, también se integró funciones con IA como chatbots para responder preguntas frecuentes, y un asistente para la generación de contenido.

Se incorporó una aplicación móvil complementaria que permite a padres y tutores de los beneficiarios el registro y seguimiento, al incorporar notificaciones automáticas y recordatorios de citas, un historial digital, también se incorporaron funciones con IA como asistente virtual, estas funciones optimizan el seguimiento y la comunicación.

Generalidades

Antecedentes

El cáncer infantil se convirtió una problemática seria en Bolivia, no solo por la cantidad de casos reportados también por el impacto emocional y económico que se genera en las familias afectadas. Según la OMS entre 2016 y 2023 se registraron 3.347 casos en Bolivia, lo que equivale a un promedio de 418 diagnósticos anuales. Mas allá de las cifras, el acceso a tratamientos oncológicos pediátricos sigue limitado debido a esto muchas familias y muchas familias deben enfrentar grandes sacrificios para poder acceder un tratamiento oncológico que en muchos casos es inaccesible (Ministerio de Salud y Deportes, 2023).

La Fundación OncoFeliz es una organización sin fines de lucro que fue fundada en el año 2009 que logro obtener su personería jurídica desde el 2012, decidió apoyar a niños con cáncer que son diagnosticados en el Hospital Manuel Ascencio Villarreal, en el departamento de Cochabamba Bolivia, este al ser un hospital de tercer nivel el 100% de las familias que acuden son de escasos recursos y no cuentan con los medios para poder cubrir los altos costos de un tratamiento o exámenes médicos, por esta razón se decidió apoyar a este grupo en específico para que gracias a su ayuda puedan acceder a los tratamientos. La fundación apoya a niños con cáncer desde recién nacidos hasta los 15 años y se encarga de proveer los recursos para cubrir los estudios, tratamientos y asistencia psicológica. Es importante destacar que es una organización externa al hospital, ya que al ser esta una institución pública, no puede recibir donaciones en forma directa, debido a esto, la Fundación canaliza las donaciones para ayudar a los niños diagnosticados con cáncer, cuyos medicamentos y estudios no pueden ser cubiertos completamente por el Servicio Universal de Salud (SUS).

En los últimos años OncoFeliz logró cubrir estudios médicos, medicación y otros insumos para numerosos pacientes gracias al esfuerzo de donantes y voluntarios, según un reportaje en el año 2024 la Fundación OncoFeliz pudo apoyar a 75 niños que lograron pasar a una etapa de remisión, y se dio seguimiento a 45 pacientes que ya se encontraban en remisión, en el mismo año se logró implementar un laboratorio histopatológico en el hospital lo que permite realizar más de 100 estudios mensuales de biopsia y citología, reduciendo tiempos de diagnóstico (Opini3n, 2024).

Conseguir nuevos donantes es fundamental para cualquier organizaci3n sin fines de lucro. Incluso las misiones m3s impactantes pueden tener dificultades para mantener su impulso sin un flujo constante de apoyo. Sin embargo, atraer donantes no se trata

solo de abarcar una amplia red. Se trata de conectar con las personas adecuadas, cultivar relaciones y transformar el interés en apoyo sostenido.

Este proyecto tiene como objetivo ampliar la presencia digital en la fundación, fortalecer las estrategias de captación de recursos al facilitar las donaciones en línea y sistematizar el seguimiento de beneficiarios. La plataforma no solo permitirá las donaciones en línea, mostrar el impacto del trabajo de la fundación, sino también establecer un sistema de monitoreo proporcionando una atención más organizada y sistemática a sus beneficiarios.

Descripción del objeto de estudio

El objeto de estudio del presente proyecto es el trabajo que realiza la Fundación OncoFeliz en cuanto a la comunicación de sus actividades, la captación de donaciones y el seguimiento de los niños beneficiarios en la ciudad de Cochabamba, Bolivia. Estas tareas incluyen dar a conocer el impacto del apoyo de la fundación, organizar las donaciones que recibe y un seguimiento adecuado de los beneficiarios.

El estudio se enfoca en reconocer las dificultades que enfrentan y plantear una solución digital que se divide en dos partes, una página web que muestre información y permita organizar las donaciones, y una aplicación móvil enfocada para que los beneficiarios tengan un seguimiento organizado.

Diseño técnico de la investigación

Planteamiento del problema

La Fundación OncoFeliz es una organización sin fines de lucro que ayuda a niños con cáncer que no tienen los recursos para pagar su tratamiento. Les da apoyo psicológico, social y económico, además de acompañar a las familias durante todo el proceso. Sin embargo, la fundación tiene varios problemas, sobre todo porque no cuenta con una plataforma digital que le permita mostrar su trabajo ni recibir donaciones en línea.

Por ahora, todo se hace de forma manual. Las donaciones se realizan por depósito o de manera personal, lo que hace que el proceso sea más lento y complicado. Además, las campañas y actividades se publican solo en redes sociales, lo que no ayuda mucho a llegar a más personas ni a mostrar todo el esfuerzo que realiza la fundación para apoyar a los niños y sus familias, esto hace que no tenga una buena presencia digital ni mucha visibilidad ante nuevos donantes.

El seguimiento de los beneficiarios también se realiza de forma manual, a través de registros dispersos en documentos y hojas de cálculo, sin un historial centralizado de la ayuda brindada ni recordatorios automáticos de citas con el psicólogo o el trabajador social. Esta falta de sistematización genera dificultades para llevar un control ordenado de las asistencias, actividades y apoyos otorgados, afectando la continuidad de la atención y la coordinación interna del equipo de la Fundación.

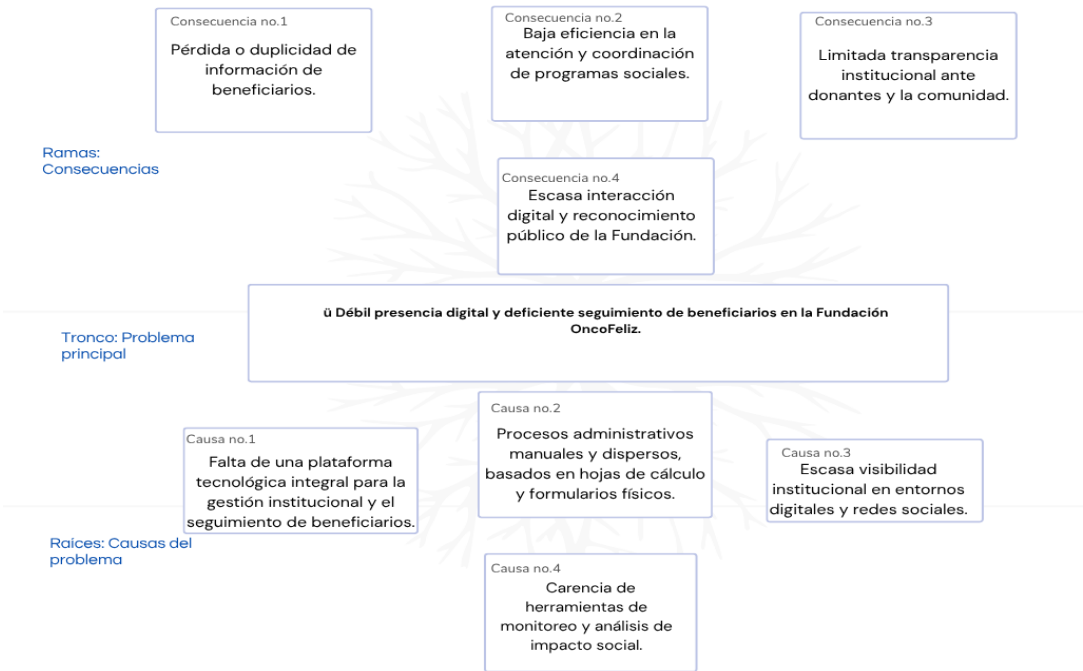
Ante esta situación se propone desarrollar una plataforma digital que este compuesta por una página web y una aplicación móvil.

La página web servirá para mostrar las campañas, los casos de recuperación, el impacto del trabajo de la fundación, además facilitará que las personas puedan hacer donaciones en línea de manera rápida y segura.

La aplicación móvil estará orientada a los beneficiarios y sus familias, brindándoles acceso a recordatorios de citas, historial de apoyos recibidos y herramientas para mantener una comunicación constante con la Fundación, mejorando la organización y continuidad del acompañamiento social y psicológico.

Árbol de problemas

Figura 1. Árbol de problemas



Fuente. Elaboración propia (2025).

Formulación del problema

¿Cómo puede una plataforma digital con página web y aplicación móvil ayudar a mejorar la presencia digital de la Fundación OncoFeliz y facilitar el seguimiento de sus beneficiarios?

Objetivo general

Desarrollar una plataforma digital para la Fundación OncoFeliz, que integre una página web y una aplicación móvil, con el fin de fortalecer su presencia en línea, facilitar la captación de donaciones y optimizar el seguimiento de beneficiarios.

Objetivos específicos

- Identificar las necesidades de la Fundación OncoFeliz en relación al seguimiento de beneficiarios, captación de donaciones y la visibilidad digital.
- Desarrollar la plataforma web aplicando tecnologías actuales de programación y metodologías ágiles (SCRUM), garantizando la escalabilidad, accesibilidad y seguridad del sistema.
- Desarrollar una aplicación móvil que permita el seguimiento de beneficiarios, que faciliten el registro, control y actualización de datos sobre los programas de apoyo, asistencias y donaciones.
- Diseñar la arquitectura y base de datos del sistema, estructurando los módulos que permitan la gestión eficiente de usuarios, beneficiarios, programas y actividades institucionales.
- Integrar herramientas digitales, asistente virtual basada en IA, inteligencia artificial para mejorar la personalización de contenidos.

Definición de variables

Tabla 1. Definición de variables

Variable	Tipo	Definición conceptual	Definición operativa
Implementación de la plataforma digital (web + app móvil)	Independiente	Conjunto de herramientas digitales orientadas a sistematizar la gestión institucional, centralizando procesos de comunicación, donaciones y seguimiento de beneficiarios.	Desarrollo e implementación de una página web funcional y una aplicación móvil que integren módulos de gestión de donaciones, comunicación y monitoreo de beneficiarios.
Presencia digital de la Fundación OncoFeliz	Dependiente	Nivel de visibilidad y posicionamiento institucional en entornos digitales, vinculado al uso de la plataforma y la interacción en medios virtuales.	Se entiende como la disponibilidad de la plataforma web y la aplicación móvil como medios oficiales de información, comunicación y proyección institucional en el entorno digital.

Gestión de donaciones	Dependiente	Mecanismos digitales habilitados para organizar, registrar y facilitar los aportes de los donantes de manera confiable y accesible.	Número de donaciones registradas en la plataforma, tiempos de procesamiento y reportes de transparencia generados automáticamente.
Seguimiento de beneficiarios	Dependiente	Administración de la información de familias beneficiarias y control de procesos que permitan monitorear su historial, actividades y apoyo recibido.	Se evaluará mediante la cantidad de beneficiarios con historial actualizado en la plataforma y el acceso oportuno a información de seguimiento y citas.

Fuente. Elaboración propia (2025).

Justificación

En el siguiente apartado se describen las justificaciones del proyecto como tecnología, social y económica.

Justificación social

El proyecto tiene como objetivo robustecer la labor de la Fundación OncoFeliz al ofrecer una herramienta que mejore la conexión entre la organización y la comunidad. Una mejor divulgación de las actividades para sensibilizar a la sociedad, mientras que disponer de una plataforma digital permitirá que el proceso de donaciones sea fácil y directo, lo cual incrementará la confianza en los donantes y los posibles donantes.

Justificación técnica

El presente proyecto propone aplicar una solución digital moderna y escalable que atienda las demandas actuales de la Fundación OncoFeliz, donde muchos de los procesos todavía se llevan a cabo de manera dispersa y manual. La administración de la información de los beneficiarios se optimiza y el contacto con donantes al mostrar el impacto del trabajo realizado se fortalece al implementar una plataforma web que brinde acceso diferenciado según roles, así como una aplicación móvil con funciones para notificar y hacer seguimiento.

Justificación económica

De acuerdo con investigaciones de la Asociación de Fundraising Digital, las plataformas digitales han demostrado ser instrumentos útiles para incrementar la captación de fondos en entidades sin fines de lucro. Las organizaciones que utilizan plataformas para donaciones virtuales tienen el potencial de elevar sus ingresos hasta un 30%.

Metodología y técnicas de la investigación

Balestrini (2000) señala que el marco metodológico “es el conjunto de procesamiento a seguir con la finalidad de lograr los objetivos de la información de forma válida y con una lista alta de precisión” (p.44). En otras palabras, es la recolección y el análisis de la información, que permite la interpretación de los resultados en función del problema que se investiga.

Tipo de investigación

Los métodos mixtos de investigación son enfoques que combinan tanto elementos cualitativos como cuantitativos en un estudio o investigación, estos métodos buscan aprovechar las fortalezas de ambos enfoques para obtener una comprensión más completa y profunda de un fenómeno o problema de investigación.

En esta investigación se aplicará un enfoque mixto, el cual combina estrategias cualitativas y cuantitativas con el fin de proporcionar una comprensión más completa del fenómeno estudiado. Como lo señalan Hernández Sampieri, Fernández Collado y Baptista Lucio (2014), “el enfoque mixto no sólo busca corroborar hallazgos, sino también ampliar, enriquecer y profundizar en la información obtenida” (p. 567).

Este enfoque permite aprovechar las fortalezas de ambos métodos para comprender las necesidades de la Fundación OncoFeliz, al recolectar datos que permitan sustentar el desarrollo de la plataforma digital enriqueciendo así el análisis y la interpretación de los resultados.

Desde el enfoque cualitativo se recogen percepciones y experiencias de los padres, voluntarios y personal de la fundación a través de entrevistas con el fin de conocer su opinión sobre la presencia en línea de la Fundación OncoFeliz y las dificultades en la comunicación institucional, así como aspectos vinculados a la continuidad de los tratamientos. Desde el enfoque cuantitativo, se utilizaron encuestas aplicadas a padres, donantes y voluntarios para identificar tendencias y patrones en sus respuestas respecto al uso de medios digitales, el interés por colaborar y las formas de interacción con la fundación.

Desde el enfoque cuantitativo, se utilizaron encuestas estructuradas dirigidas a padres, donantes, voluntarios y personal administrativo, con el fin de recopilar datos sobre la percepción de la presencia digital de la Fundación OncoFeliz, la facilidad para realizar donaciones en línea y el nivel de interacción con las plataformas digitales. También se recogieron datos relacionados con la continuidad en los tratamientos, el uso de recordatorios y la utilidad de los canales de comunicación ofrecidos por la fundación.

Alcance de la investigación

La presente investigación posee un alcance exploratorio-descriptivo ya que se orienta a indagar también caracterizar los desafíos que enfrenta la Fundación OncoFeliz en relación con su limitada presencia digital la captación de donaciones y seguimiento de beneficiarios. El enfoque exploratorio permite identificar aspectos poco estudiados o que no son comprendidos dentro de este contexto social y tecnológico específico. Este método se aplica principalmente en contextos donde hay escasa información, como en los procesos de administración digital y comunicación institucional en entidades no lucrativas como la Fundación OncoFeliz, o cuando no se cuenta con muchos datos o registros sistemáticos.

El doble enfoque exploratorio-descriptivo proporciona una base sólida para comprender la situación actual para proponer el desarrollo de una plataforma digital como alternativa viable para mejorar la comunicación institucional, facilitar la captación de fondos y garantizar un mejor seguimiento a los tratamientos de los niños beneficiarios.

Diseño de la investigación

El diseño de la presente investigación es no experimental y transversal, debido a que no se manipulan variables deliberadamente, se observa el fenómeno tal como ocurre en su contexto natural, este tipo de diseño los datos se recolectan en un solo momento con el propósito de analizar cómo se presentan las variables en una situación específica. Este diseño se eligió debido a que el objetivo principal del estudio es diagnosticar una problemática institucional sin intervenir directamente en sus procesos internos. La intención no es modificar comportamientos ni introducir cambios durante la fase de recolección de datos solo es comprender la situación actual para plantear propuestas tecnológicas que respondan a las necesidades identificadas, Un enfoque transversal permite captar información en un momento determinado lo que resulta útil para realizar un análisis contextual preciso y actualizado.

Técnicas e instrumentación de recolección de datos

Entrevistas y encuestas con diferentes usuarios (donantes, voluntarios, beneficiarios y posibles donantes) y el análisis documental de los procesos actuales.

- Entrevistas con padres y miembros del personal de la fundación para conocer en profundidad sus experiencias y necesidades.
- Encuestas estructuradas aplicadas a donantes actuales, padres de familia y posibles donantes, con el objetivo de recolectar información cuantitativa y

cualitativa sobre sus hábitos digitales, nivel de confianza, inclinación a donar y opiniones sobre la comunicación institucional. La encuesta cuenta de 13 preguntas en total, entre las cuales 12 fueron de selección múltiple y 1 fue abierta para permitir la expresión libre de sugerencias o comentarios

- Revisión documental de publicaciones institucionales del funcionamiento actual en redes sociales.

Fuentes de información

Las fuentes de información incluyen:

- Padres de familia beneficiarios de la Fundación OncoFeliz.
- Donantes activos o potenciales.
- Voluntarios y personal administrativo de la fundación.
- Registros institucionales y documentos informativos.

Alcances y aportes

Se describen los alcances del proyecto en términos espaciales, temporales y temáticos.

Alcance espacial

- Ubicación: La implementación del sistema se llevará a cabo en la Fundación OncoFeliz ubicada en Cochabamba Bolivia.
- Usuarios: El sistema estará disponible para todos los donantes, futuros donantes, voluntarios y beneficiarios sin importar su ubicación o donde se encuentren.
- Accesibilidad: El sistema estará disponible en cualquier dispositivo que tenga conexión a internet, permitiendo a los usuarios acceder a la página web desde cualquier lugar.

Alcance temporal

- Fase de desarrollo: Se estima que el desarrollo del sistema web y la aplicación móvil tomará unos 6 meses aproximadamente, desde la fase de análisis de requerimientos hasta la implementación final.
- Fase de pruebas: Después del desarrollo, se llevará a cabo la fase de pruebas, que durar aproximadamente un mes donde se realizará pruebas de funcionalidad, usabilidad y rendimiento.

Alcance temático

- Desarrollo de una plataforma digital que incluye una página web y una aplicación móvil para mejorar la difusión de las actividades y campañas de la Fundación OncoFeliz.
- Integración de una sección de donaciones en la plataforma, que permita a los usuarios contribuir de manera sencilla, segura y accesible.
- Diseño de una interfaz intuitiva y accesible que facilite la interacción de los donantes con la fundación.
- Implementación de un chatbot basado en inteligencia artificial para brindar asistencia y responder consultas frecuentes a los usuarios de la plataforma.
- Desarrollo de un módulo administrativo con herramientas de inteligencia artificial para la generación automatizada de publicaciones relacionadas con eventos y campañas solidarias.
- Implementación de una base de datos segura para almacenar y gestionar información relevante sobre donaciones y actividades de la fundación.

Limites

- La plataforma solo generará reportes generales de donaciones, sin detallar el uso específico de los fondos, ya que no se abordarán procesos administrativos internos complejos de la Fundación OncoFeliz.
- El seguimiento de beneficiarios se realizará únicamente mediante recordatorios de citas con psicólogos y trabajadores sociales, sin incluir información sobre citas médicas, las cuales son gestionadas por el hospital.
- No se incluirá el desarrollo de pasarelas de pago complejas, se integrarán únicamente métodos de donación existentes, como transferencias bancarias, pagos por QR o plataformas externas.
- No se abordará la automatización de procesos logísticos para la entrega de medicamentos o insumos, ya que esto seguirá siendo gestionado directamente por la fundación.

CAPÍTULO I

MARCO TEÓRICO

1.1. Transformación digital en organizaciones sin fines de lucro

La transformación digital en organizaciones sin ánimo de lucro se entiende como un proceso integral que abarca no solo la implementación de nuevas tecnologías, sino también la adaptación de procesos internos, equipos de trabajo y cultura institucional. Estos cambios se orientan a mejorar la eficiencia, optimizar la comunicación y ampliar el alcance de las actividades realizadas por estas entidades (Tercer Sector Digital, 2024).

En el desarrollo de una plataforma digital de la fundación OncoFeliz una organización sin fines de lucro, la transformación digital no solo se trata de implementar nuevas tecnologías, se trata de mejorar los procesos de comunicación y gestión de donaciones. Según Ahmed et al. (2024), "la digitalización permite a las organizaciones sin fines de lucro mejorar sus resultados, aumentar la satisfacción de donantes y voluntarios, hacer más efectivos sus programas y llegar a más personas" (p. 1).

Un informe reciente muestra que "menos de dos de cada cinco organizaciones sin fines de lucro tienen un plan de transformación digital bien definido, lo que demuestra la necesidad de planificar de forma intencionada en este sector" (Infoxchange, 2023, p. 3). Por esto el relacionar la estrategia digital con los objetivos de visibilidad, captación de recursos y seguimiento de beneficiarios de la Fundación fortalece la propuesta del sistema y lleva la digitalización a un nivel práctico que se ajusta a las necesidades reales de la organización

1.1.1 Impacto de la digitalización en organizaciones no lucrativas

La digitalización cambió la manera en que operan las organizaciones sin fines de lucro, cambiando de diferentes formas su eficacia y viabilidad en un mundo donde cada día va avanzando la tecnología y es más conectado. Este proceso permite a estas instituciones a mejorar el uso de sus recursos y lograr una disminución de sus gastos, además de aumentar la efectividad en la entrega de servicio su adopción promovió la automatización de tareas administrativas eliminando trabajos manuales repetitivos y permitiendo que los empleados y voluntarios se concentren en actividades que generan un mayor impacto (Salamon & Anheier, 2020).

En el caso de la Fundación OncoFeliz, la transformación digital tiene un valor estratégico, el desarrollo de una plataforma digital permitirá fortalecer la comunicación institucional y optimizar la gestión de información, la tecnología es un medio para lograr un impacto social más amplio.

1.1.2 Plataformas digitales y su rol en la captación de donaciones

Según un estudio realizado por el *Peer-to-Peer Professional Forum* (2021), 44 organizaciones sin fines de lucro generaron 154.9 millones de dólares mediante programas de recaudación de fondos digitales, lo que representa un incremento del 30% respecto a los ingresos de 2020. Este crecimiento se atribuye a la adopción de estrategias digitales como la recaudación entre pares, las cuales han demostrado ser efectivas para ampliar la base de donantes y aumentar los ingresos.

Las plataformas en línea se convirtieron en herramientas relevantes para las ONG, ya que apoyaron sus esfuerzos por recaudar fondos al facilitar métodos de pago digitales de manera eficaz y segura. La implementación de estos métodos permite que las transacciones se realizaran de forma más rápida, sencilla y confiable, lo cual genera confianza y fomentó la participación de donantes. Además, la utilización de técnicas de marketing digital, como campañas de concientización en redes sociales, anuncios en línea y correos electrónicos personalizados, incrementa la visibilidad de las causas sociales, atrayendo a un mayor número de donantes. La integración de tecnología y marketing fortalece la relación con los donantes, no solo facilita el acceso a las donaciones, sino también promueve un compromiso a largo plazo. Por estas razones, estas herramientas tienen un impacto significativo en la sostenibilidad financiera de las ONG, lo cual permite llevar a cabo proyectos y alcanzar objetivos de manera efectiva (Waters, 2018).

Figura 2. Plataformas digitales y su rol en la captación de donaciones



Fuente: Freepik, 2025.

1.1.3 Beneficios de la digitalización en la gestión de recursos y comunicación

La digitalización en la gestión de recursos de comunicación permite a las organizaciones optimizar la planificación, creación y difusión de información, logrando un uso más eficiente de los recursos disponibles. Al implementar herramientas digitales, se mejoró la coordinación entre los equipos de trabajo, se agilizó la transmisión de mensajes y se redujeron los tiempos y costos asociados a la comunicación tradicional. Asimismo, la digitalización facilitó el monitoreo y análisis del impacto de las estrategias comunicacionales, lo que contribuyó a la toma de decisiones más informada y a un fortalecimiento de la relación con los públicos objetivos. En consecuencia, estas mejoras favorecieron la efectividad de las acciones comunicacionales y potenciaron el alcance de los objetivos institucionales (Tercer Sector Digital, 2024).

1.1.4. Altos costos, barreras de acceso y rol de las ONG en el cáncer infantil en Bolivia

1.1.4.1. Altos costos del tratamiento del cáncer en Bolivia

En Bolivia, el tratamiento del cáncer infantil enfrenta retos significativos en términos de costos, acceso y continuidad del cuidado. Se registran aproximadamente 467 nuevos casos al año en el país, y la tasa de mortalidad en menores de 20 años es alta en comparación con otros países de la región.

El tratamiento del cáncer infantil en Bolivia implica gastos elevados que exceden la infraestructura hospitalaria y la atención médica básica. Los medicamentos oncológicos pueden costar alrededor de US 2.000 cada 1-4 meses, la radioterapia unos US 3.500, y los trasplantes de médula ósea, realizados en el extranjero, oscilan entre US 30.000 y US 60.000 (World Bank, 2021).

Estas cifras muestran que además del diagnóstico las familias enfrentan un desafío económico significativo, lo que puede limitar la capacidad de acceder y mantener el tratamiento completo para los menores. Los altos costos generan la necesidad de apoyos externos y fortalecen la importancia de contar con organizaciones que acompañen a las familias y faciliten la gestión de recursos.

1.1.4.2. Barreras de acceso y desigualdades en la atención

Además de los costos, existen múltiples barreras de acceso al tratamiento, que incluyen factores geográficos, culturales y logísticos. En zonas rurales o comunidades indígenas, alejadas de los centros oncológicos, la demora en el diagnóstico o la falta de especialistas aumenta la mortalidad infantil (Calle et al., 2023).

Aun cuando el sistema de salud pública ofrece cobertura parcial, como el seguro Ley 475/SIS que cubre ciertos medicamentos para niños menores de cinco años con leucemia linfoblástica aguda (LLA), muchos tratamientos y edades quedan fuera de esta cobertura (World Bank, 2021).

La combinación de estas desigualdades contribuye a que Bolivia presente tasas de mortalidad infantil por cáncer superiores al promedio regional (PAHO, 2021).

1.1.4.3. Rol de las ONG en la atención del cáncer infantil

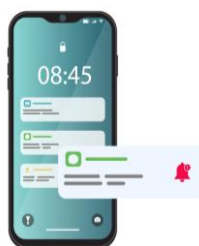
Las organizaciones no gubernamentales desempeñan un papel fundamental en la atención del cáncer infantil, contribuyendo al bienestar integral de los pacientes y sus familias. Su labor abarca el acompañamiento emocional y social, ofreciendo apoyo psicológico y orientación a los familiares durante todo el proceso de tratamiento. Además, las ONG facilitan el acceso a recursos financieros y materiales, gestionan donaciones y colaboran con instituciones de salud para garantizar la continuidad del cuidado. También participan en la educación y sensibilización de la comunidad sobre la enfermedad, promoviendo campañas de concientización y visibilización de los casos (Calle et al., 2023).

1.2. Sistemas de notificaciones automáticas

Los sistemas de notificaciones automáticas permitieron mantener una comunicación constante y oportuna entre el sistema y sus usuarios, optimizando la gestión de información y mejorando la experiencia del usuario. Estas herramientas facilitaron la entrega de recordatorios, alertas y actualizaciones relevantes, contribuyendo a una mayor eficiencia en la interacción con la plataforma (Morrison, 2020).

Este tipo de funcionalidades tienen un papel importante en la plataforma digital de la fundación OncoFeliz, puede emplearse para recordatorios de citas médica, ayuda a evitar el abandono del tratamiento de este modo fortalecer la comunicación y el seguimiento a los beneficiarios.

Figura 3. Ilustración de notificación push en dispositivo móvil



Fuente: Waalaxy Blog, 2023.

1.2.1. Beneficios de los sistemas de notificación

Los sistemas de notificaciones proporcionaron múltiples beneficios, como la reducción de errores por olvido, la mejora en la organización de actividades y la comunicación efectiva de información crítica. Además, favorecieron la eficiencia operativa al automatizar la difusión de mensajes importantes, asegurando que los usuarios recibieran información relevante en el momento adecuado (Kumar & Gupta, 2019).

Los recordatorios automáticos facilitan la comunicación, optimizan los procesos al reducir el trabajo manual, es una herramienta clave en instituciones con recursos limitados (Jones & Patel, 2021).

Los sistemas de notificaciones ofrecen múltiples beneficios, entre los que destacan:

- Reducción de errores por olvido.
- Mejora en la organización de actividades.
- Comunicación efectiva de información crítica.

1.2.2. Impacto de la adherencia al tratamiento

La implementación de notificaciones automáticas tuvo un efecto positivo en la adherencia al tratamiento en contextos de salud y seguimiento de actividades. Los recordatorios enviados de manera regular ayudaron a los usuarios a cumplir con sus pautas y procedimientos, aumentando la consistencia y efectividad de las intervenciones (Smith et al., 2021).

1.2.3. Automatización de procesos en organizaciones benéficas

La automatización de procesos en organizaciones benéficas permite mejorar la eficiencia en la gestión de sus actividades y garantizar que la información relevante llegue a tiempo a los beneficiarios y donantes. Según Blackbaud (2025), *la automatización utiliza tecnología para mejorar los procesos y la visibilidad entre ellos, con una mínima intervención humana, lo que permite a las organizaciones benéficas escalar de manera más efectiva* (The ENGAGE Blog by Blackbaud).

Entre los beneficios más relevantes se destacan:

- **Eficiencia operativa**

La ejecución automática de tareas rutinarias disminuye la carga de trabajo manual y los errores administrativos (Claromentis, 2025).

- **Mejora en la comunicación con beneficiarios**

Los recordatorios y notificaciones automáticas aseguran que los usuarios reciban información oportuna sobre programas, actividades o servicios (Neon One, 2025).

- **Optimización de recursos**

Al automatizar procesos internos, el personal puede dedicarse a actividades estratégicas y de apoyo directo (Orr Group, 2025).

- **Seguimiento y control**

La automatización permite registrar y monitorear de manera sistemática la participación de beneficiarios y la gestión de donaciones (KandaSoft, 2024).

1.2.4. Tecnologías para la implementación de notificaciones push y correo electrónico

Las notificaciones push y los correos electrónicos se implementaron mediante tecnologías que permiten la entrega instantánea de mensajes a dispositivos móviles y plataformas web. Estas tecnologías incluyen servicios de mensajería en la nube, APIs de notificaciones y librerías especializadas que aseguran la entrega confiable y personalizada de la información a los usuarios (Morrison, 2020; Kumar & Gupta, 2019).

Las notificaciones push permiten enviar alertas de manera más rápida a dispositivos móviles, también el correo electrónico es útil para envíos más detallados y seguimiento de actividades (Patel & Patel, 2019).

Las notificaciones push y los correos electrónicos son herramientas primordiales para mantener la comunicación con usuarios y mejorar la adherencia a tratamientos o la interacción con una plataforma.

1.3. Usabilidad y accesibilidad en plataformas digitales

Las plataformas digitales deben cumplir normas de accesibilidad, por esta razón deben ser accesibles también intuitivas para todos los usuarios.

Un entorno digital accesible permite que los contenidos sean claros, comprensibles y manejables por todos los usuarios. En el caso del proyecto de la Fundación Oncofeliz estos principios son importantes debido a que voluntarios, padres, donantes y posibles donantes tienen distintos niveles tecnológicos, por esta razón se debe ofrecer una interfaz clara, textos comprensibles y una navegación sencilla.

1.3.1. Diseño centrado en el usuario (DCU)

Diseñar pensando en el usuario es lo que significa el diseño centrado en el usuario (DCU). Los equipos de diseño y desarrollo emplean la información que recolectan sobre las necesidades, metas y opiniones de los usuarios para desarrollar aplicaciones y páginas web muy accesibles y fáciles de usar (Baymard Institute, 2024).

El DCU, o diseño centrado en el usuario, generalmente se lleva a cabo en un ciclo que consta de cuatro etapas fundamentales: Entender el contexto de uso, determinar las necesidades del usuario.

Etapas fundamentales DCU:

- Comprensión del contexto de uso: Los diseñadores investigan y analizan el entorno en el que los usuarios interactúan con el sistema, considerando factores físicos, sociales y tecnológicos.
- Especificación de requisitos del usuario: Se identifican y documentan las necesidades y expectativas de los usuarios, estableciendo criterios claros que el diseño debe cumplir.
- Desarrollo de soluciones de diseño: Con base en los requisitos recopilados, se generan propuestas de diseño que buscan satisfacer las necesidades identificadas.
- Evaluación del diseño: Se prueban y analizan las soluciones desarrolladas para determinar su eficacia y grado de satisfacción de los usuarios, utilizando los resultados para realizar mejoras continuas.

Figura 4. Etapas del diseño DCU



Fuente: HubSpot, 2023

1.3.2. Accesibilidad web y directrices WCAG

Son recursos esenciales para personas y organizaciones que buscan crear y mantener contenido digital accesible, como indica su definición, las WCAG son un conjunto detallado de directrices técnicas que explican los requisitos para que su sitio web, aplicación u otra propiedad digital sea accesible para personas con discapacidad.

Las Pautas de Accesibilidad al Contenido Web (WCAG) son los estándares que fueron reconocidos internacionalmente por el Consorcio World Wide Web (W3C) para garantizar que el contenido digital sea accesible para todos los usuarios, incluidas aquellas personas con discapacidad. Los diseñadores de UX (experiencia de usuario) cumplen con las WCAG y crean interfaces de usuario perceptibles, operables, comprensibles y robustas. (W3C, 2018).

Comprender la importancia de las WCAG es el primer paso para crear una experiencia en línea más accesible para todos:

- Discapacidades visuales (baja visión, ceguera, daltonismo).
- Discapacidades auditivas (dificultad para oír o sordera).
- Movilidad y discapacidades motoras.
- Discapacidades cognitivas, de aprendizaje y neurológicas.
- Discapacidades del habla.

1.3.2.1. Cumplimiento de los criterios WCAG

El cumplimiento de las WCAG se refiere a las instrucciones técnica y los criterios para desarrollar sitios web, estos criterios son los que ayudan a optimizar los sitios web, satisfacer ciertas necesidades específicas tecnológicas de asistencia. También se puede utilizar la información de una auditoría para implementar mejoras conforme a los criterios WCAG que mejoren la usabilidad y la accesibilidad para todos (Seale, 2022).

Al hacer que los sitios web puedan ser accesibles es un paso importante para eliminar las barreras que muchos usuarios enfrentan al navegar sobre un contenido. Ampliar el acceso se refiere a incluir a personas con diferentes capacidades, la accesibilidad también se refiere a brindar el acceso de contenido desde diferentes dispositivos.

Figura 5. Principios de accesibilidad WCAG



Fuente: Ovacen, 2024.

1.3.3. Diseño Inclusivo en plataformas digitales

El diseño inclusivo en plataformas digitales hace referencia a la creación productos y servicios que puedan ser utilizados por la mayor cantidad de personas posible, debe ser entendido como una nueva "filosofía" de diseño que también intenta satisfacer las necesidades de accesibilidad al mayor número de usuarios posibles.

1.3.4. Principios de diseño de interfaces intuitivas y amigables

Los principios del diseño de interfaces intuitivas incluyen una clasificación visual clara debido a que estudios recientes han confirmado que el uso de patrones de diseño predecibles y la simplificación de los flujos de interacción pueden incrementar la retención de usuarios en plataformas digitales puede aumentar un 30% más y la implementación de principios de usabilidad al desarrollar interfaces más accesibles y fáciles de usar beneficiando tanto a usuarios nuevos como experimentados (Kumar, 2023).

Una interfaz amigable e intuitiva le da más énfasis a la parte visual, la sencillez y la consistencia. Esto ayuda a que los usuarios puedan navegar y realizar distintas acciones sin requerir un manual o capacitación. Una interfaz bien diseñada no solo

facilita la interacción, también aumenta el agrado del usuario y hace más fácil el uso del sistema (MediumMultimedia, 2025).

Los principios del diseño de interfaces intuitivas incluyen una clasificación visual clara debido a que estudios recientes han confirmado que el uso de patrones de diseño predecibles y la simplificación de los flujos de interacción pueden incrementar la retención de usuarios en plataformas digitales puede aumentar un 30% más y la implementación de principios de usabilidad al desarrollar interfaces más accesibles y fáciles de usar beneficiando tanto a usuarios nuevos como experimentados (Kumar, 2023).

Los principios de diseño centrado en el usuario destacan la importancia de factores como:

- Claridad en la organización de la información.
- Consistencia en los elementos visuales y de interacción.
- Retroalimentación inmediata del sistema frente a las acciones del usuario.
- Diseño adaptativo y responsivo, para garantizar el acceso desde distintos dispositivos.

1.4. Seguridad y gestión de datos

La seguridad y gestión de datos son aspectos fundamentales para proteger la información almacenada y procesada en los sistemas digitales. La implementación de políticas y mecanismos de seguridad permite garantizar la confidencialidad, integridad y disponibilidad de los datos, evitando accesos no autorizados y pérdidas de información. Se aplican técnicas de cifrado, control de acceso y auditorías periódicas para asegurar que la información se gestiona de manera confiable y cumpliendo con normativas y estándares internacionales. La adecuada gestión de datos también facilita la toma de decisiones basadas en información precisa y actualizada, optimizando el rendimiento y la eficiencia de los sistemas (Stallings, 2020; Laudon & Laudon, 2021).

1.4.1. Protección de datos personales y privacidad

La protección de datos personales y la privacidad son importantes en el desarrollo y el uso de plataformas digitales, para garantizar que la información sensible de los usuarios se gestione de manera segura (Solove, 2021).

1.4.2. Normativas legales sobre la protección de datos (GDPR, Ley de Protección de Datos Personales)

Tecnologías de Información y Comunicación establece principios y directrices para el manejo de datos, incluyendo la privacidad y la seguridad de la información comunicada y almacenada a través de medios electrónicos. Si bien la Ley 164 aborda aspectos generales de protección de datos y comunicaciones, la información de salud, al ser considerada sensible, requiere un tratamiento especial. A nivel internacional, el Reglamento General de Protección de Datos de la Unión Europea es una normativa robusta que ha sentado precedentes en la protección de datos personales. Aunque aplica primariamente a ciudadanos de la UE, su influencia se extiende globalmente, promoviendo principios como el consentimiento explícito para el tratamiento de datos sensibles, el derecho al acceso y rectificación de datos, el derecho al olvido y la necesidad de implementar medidas de seguridad adecuadas para prevenir brechas de datos.

En América Latina en diferentes países se fueron implementando sus propias leyes de protección de datos personales. Esta ley exige que las organizaciones obtengan el consentimiento explícito de los usuarios antes de procesar sus datos y establece que los usuarios tienen el derecho a acceder, rectificar, cancelar y oponerse al tratamiento de sus datos personales, conocido como "derechos ARCO" (Acceso, Rectificación, Cancelación y Oposición), la Ley de Protección de Datos Personales en Posesión ha impulsado una mayor transparencia y seguridad en la gestión de los datos personales por parte de las empresas (Díaz; 2020)

El reglamento de protección de datos asegura que datos personales deben ser tratados de forma legal y segura, usar los datos solamente para fines específicos y legítimos.

1.4.3. Gestión segura de bases de datos

La gestión segura de base de datos es el conjunto de políticas, medidas y herramientas que protegen la información almacenada y los sistemas de bases de datos contra el acceso no autorizado, garantizando la confidencialidad, integridad y disponibilidad.

Una base de datos debe proteger la información almacenada, se toman medidas como los controles de acceso, cifrado y también se realizan copias de respaldo, de esta forma solo las personas autorizadas puedan acceder a la información y se evitan pérdidas o filtraciones (Elmasri & Navathe, 2020).

Para garantizar la seguridad de una base de datos, es importante aplicar controles técnicos y administrativos, algunas de las estrategias más relevantes incluyen:

- **Autenticación y control de acceso**

Implementar autenticación multifactorial y asignar privilegios mínimos a cada usuario, asegurando que solo accedan a la información necesaria.

- **Cifrado de datos**

Tanto los datos en tránsito como en reposo deben estar protegidos mediante algoritmos de cifrado robustos, como AES-256, reduciendo la posibilidad de que sean interceptados por actores maliciosos.

- **Monitoreo y auditoría**

El registro y análisis de actividad en la base de datos permiten detectar patrones sospechosos y responder rápidamente a incidentes de seguridad.

- **Respaldo y recuperación**

Contar con copias de seguridad periódicas y mecanismos de recuperación rápida minimiza el impacto de ataques como el ransomware o fallos técnicos.

1.4.4. Estrategias para el cifrado y protección de la información

Es el conjunto de técnicas como cifrado de datos, también la gestión de claves y el uso de protocolos seguros (TLS/SSL), que garantizan la confidencialidad, integridad y seguridad de la información frente a accesos no autorizados.

Características principales de TLS/SSL:

- Cifrado.
- Autenticación.
- Integridad.
- Compatibilidad.
- Transparencia para el usuario.

El cifrado de datos es una de las estrategias más importantes para garantizar la confidencialidad e integridad de la información en sistemas informáticos, se trata de transformar la información en un formato ilegible para usuarios no autorizados, de manera que solo pueda ser descifrada mediante claves o algoritmos específicos (Stallings, 2020).

Además del cifrado existen otras estrategias de protección incluyen el control de acceso basado en roles, la autenticación multifactorial, el almacenamiento seguro de contraseñas y la implementación de políticas de respaldo y recuperación ante incidentes.

1.5. Evaluación del impacto

La evaluación de impacto permite medir la efectividad y eficiencia de la plataforma digital, así como el cumplimiento de los objetivos propuestos y la experiencia de los usuarios. Este proceso permitió identificar fortalezas y áreas de mejora, contribuyendo a la optimización continua del sistema (Hevner et al., 2020).

Cuando hablamos de evaluación del impacto nos referimos a analizar los efectos generados por el uso de herramientas, enfocándose en los usuarios, de esta forma permite medir el grado de aceptación tecnológica, no solo eso también se logra medir la interacción, eficiencia y la satisfacción del usuario.

1.5.1. Método de evaluación de impacto en plataformas digitales

Los métodos de evaluación de impacto en plataformas digitales se refieren al análisis del uso de soluciones tecnológicas, usuarios y los procesos internos de una organización. Esto permite identificar las fortalezas y debilidades de la plataforma digital, evaluar la eficiencia de los sistemas implementados y medir los resultados obtenidos y objetivos planteados.

En la evaluación de plataformas digitales, “el impacto no solo se mide en función de métricas numéricas, sino también a partir de la experiencia y percepción de los usuarios” (Martínez & López, 2021, p. 45).

1.5.2. Pruebas de usuario y retroalimentación continua para mejora continua

Las pruebas no solo son pruebas de testing, también son observación de la interacción de los usuarios con la plataforma, identificación de problemas de usabilidad y la recolección de retroalimentaciones directas. La retroalimentación constante deja reiterar sobre el diseño y funcionalidad del sistema, promueve una mejora constante del producto digital.

Las pruebas de usuario (user testing) son fundamentales para asegurar que las plataformas digitales cumplan con las expectativas y necesidades de los usuarios finales (Nielsen, 2012).

1.5.3. Análisis de satisfacción del usuario y pruebas de usabilidad

El análisis de satisfacción es una parte clave en la evaluación de plataformas digitales, esto permite conocer el grado de aceptación de los usuarios a la solución tecnológica, si cubre sus necesidades y expectativas (ISO, 2018).

Las pruebas de usabilidad son una parte complementaria de análisis al medir la eficiencia y efectividad entre los usuarios y la aplicación, esto permite detectar

impedimentos, dificultades en la comprensión de funciones y barreras que afectan la experiencia general (Rubin & Chisnell, 2008). Ambos enfoques al complementarlos logran un proceso de mejora que garantiza que la plataforma no solo cumpla con los objetivos técnicos, sino que también brinde una experiencia positiva y significativa para los usuarios.

1.6. Integración de herramientas digitales y asistentes virtuales de IA

La integración de herramientas digitales y sistemas de inteligencia artificial (IA) se convirtió en un componente esencial para mejorar la interacción y experiencia del usuario en plataformas digitales. Estas tecnologías permiten no solo automatizar procesos, permite adaptar la información y servicios a las necesidades individuales de cada usuario, aumentando la relevancia y efectividad de los contenidos ofrecidos (Russell & Norvig, 2021; Marr, 2023).

1.6.1. Asistentes virtuales basados en inteligencia artificial

Los asistentes virtuales son programas informáticos que utilizan IA para interactuar con los usuarios mediante lenguaje natural, responder consultas, y guiar al usuario en el uso de la plataforma. Estas tecnologías permiten ofrecer soporte automatizado, mejorando la experiencia del usuario y optimizando la atención (Smith, 2022).

La plataforma digital para la Fundación OncoFeliz, la incorporación de **un** asistente virtual basado en IA facilita la comunicación con los padres y el personal de la fundación, brindando orientación, recordatorios y apoyo en el seguimiento de los tratamientos, lo que contribuye a fortalecer el acompañamiento y reducir el riesgo de abandono.

1.6.2. Inteligencia artificial para la personalización de contenidos

La IA aplicada a la personalización utiliza algoritmos de aprendizaje automático y análisis de datos para adaptar contenidos y recomendaciones según las preferencias, comportamiento y necesidades del usuario. Esto incluye desde recomendaciones de artículos, alertas personalizadas hasta recordatorios específicos, aumentando la adherencia y satisfacción de los beneficiarios (Kumar & Gupta, 2023; OpenAI, 2025).

Entre los principales beneficios se destacan:

- Mejora de la experiencia del usuario mediante contenidos y servicios adaptados.
- Automatización de respuestas y soporte, reduciendo la carga de trabajo manual del personal.

- Mayor adherencia y participación de los usuarios en las actividades de la plataforma.
- Recopilación de datos inteligentes, que permite tomar decisiones basadas en información real y actualizada.

1.7. Ingeniería de software aplicada al proyecto

La ingeniería de software, como disciplina esencial en la creación de sistemas informáticos robustos y eficientes, juega un papel crucial en el desarrollo de una solución para la gestión del tamizaje neonatal. Se fundamenta en la aplicación de principios y prácticas sistemáticas para abordar la complejidad inherente al diseño, desarrollo, implementación y mantenimiento de software (Pressman & Maxim, 2020).

1.7.1. Fundamentos de la ingeniería de software

Los fundamentos de la ingeniería de software proporcionan la base conceptual para el desarrollo de sistemas complejos y mantenibles. Permite modelar la realidad mediante representaciones simplificadas, mientras que el modularidad facilita la división del sistema en componentes independientes simplifica desarrollo y mantenimiento. La organización jerárquica de los módulos contribuyó a gestionar la complejidad general, el encapsulamiento protegió la integridad de cada módulo y la separación de responsabilidades aseguró que cada componente cumpliera una función específica (Pressman & Maxim, 2020).

1.7.2. Metodologías de desarrollo de software

Las metodologías de desarrollo de software proporcionan un marco estructurado para planificar, ejecutar y controlar proyectos de software. Los enfoques tradicionales, como el modelo en cascada, organizaron las fases de análisis, diseño, implementación, pruebas y mantenimiento de manera secuencial. En contraste, las metodologías ágiles promueven iteraciones cortas, colaboración continua con los usuarios y adaptabilidad frente a cambios en los requerimientos, aumentando la eficiencia y la satisfacción del cliente (Sommerville, 2021).

Métodos tradicionales:

- **Cascada (Waterfall)**

Un modelo secuencial donde cada fase del desarrollo (requisitos, diseño, implementación, pruebas, mantenimiento) debe completarse antes de pasar a la siguiente. Es menos flexible que los métodos ágiles.

Metodologías Ágiles:

- **Scrum**

Un marco de trabajo ágil que organiza el trabajo en ciclos cortos llamados "sprints". Se basa en roles específicos (Product Owner, Scrum Master, Equipo de Desarrollo) y eventos (reuniones diarias, retrospectivas) para la colaboración y la mejora del producto.

- **Kanban**

Un método ágil centrado en visualizar el flujo de trabajo a través de tableros para gestionar las tareas y optimizar el flujo de trabajo.

- **Lean**

Se enfoca en la eliminación de desperdicios y la entrega de valor continuo.

- **Scrumban**

Una combinación de Scrum y Kanban para aprovechar lo mejor de ambos enfoques.

1.7.3. Análisis de requerimientos del sistema

El análisis de requerimientos permite identificar y documentar las necesidades funcionales y no funcionales de un sistema, sirve como base para su diseño y desarrollo. Esta etapa es crucial para asegurar que un sistema cumpla con los objetivos de negocio y las expectativas de los usuarios (Wiegers & Beatty, 2013).

1.7.3.1. Requerimientos funcionales

Los requerimientos funcionales especifican las acciones que el sistema debe realizar, incluyendo procesos, cálculos, gestión de datos y generación de reportes. Estos determinaron el comportamiento esperado del software y sirven de guía para su diseño, desarrollo y pruebas (Sommerville, 2021).

1.7.3.2. Requerimientos no funcionales

Los requerimientos no funcionales definen las restricciones de calidad del sistema, tales como rendimiento, seguridad, usabilidad, confiabilidad y mantenibilidad. Garantizar el cumplimiento de estos aspectos es esencial para asegurar un software eficiente, seguro y sostenible (Pressman & Maxim, 2020).

1.7.4. Diagramas UML

Rumpe define UML como un lenguaje de modelado estándar que permite especificar, visualizar y documentar sistemas orientados a objetos mediante distintos tipos de

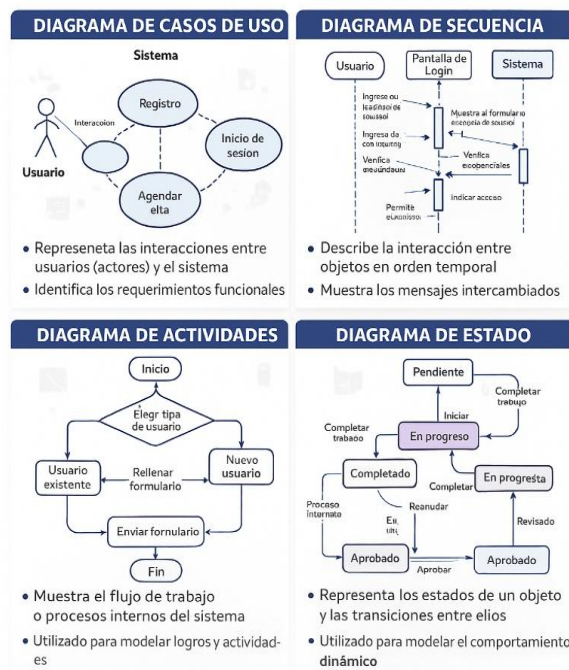
diagramas que representan tanto la estructura como el comportamiento del sistema. (Rumpe, 2016)

Los diagramas UML (Unified Modeling Language) proporcionan una representación gráfica estandarizada de un sistema, permite visualizar su estructura, comportamiento e interacciones. Herramientas como diagramas de casos de uso, diagramas de clases y diagramas de secuencia facilitan la comunicación entre los desarrolladores y los usuarios, apoyando el diseño, la documentación y la comprensión de un sistema (Booch, Rumbaugh & Jacobson, 2010).

Tipos de diagramas UML:

- **Diagrama de casos de uso**
Muestra las interacciones entre usuarios (actores) y el sistema.
- **Diagrama de secuencia**
Describe la interacción entre objetos en orden temporal.
- **Diagrama de actividades**
Muestra el flujo de trabajo o procesos internos del sistema.
- **Diagrama de estado**
Representa los estados de un objeto y las transiciones entre ellos.

Figura 6. Tipos de diagramas UML



Fuente. Elaboración propia (2025).

1.8. Arquitectura y diseño del sistema

La arquitectura y el diseño del sistema proporcionaron la estructura necesaria para el desarrollo de aplicaciones eficientes y mantenibles. Estas fases permitieron definir la organización de los componentes, la interacción entre ellos y la manera en que los datos se gestionaron, asegurando la correcta integración de los módulos y la sostenibilidad del sistema a lo largo del tiempo (Pressman & Maxim, 2020).

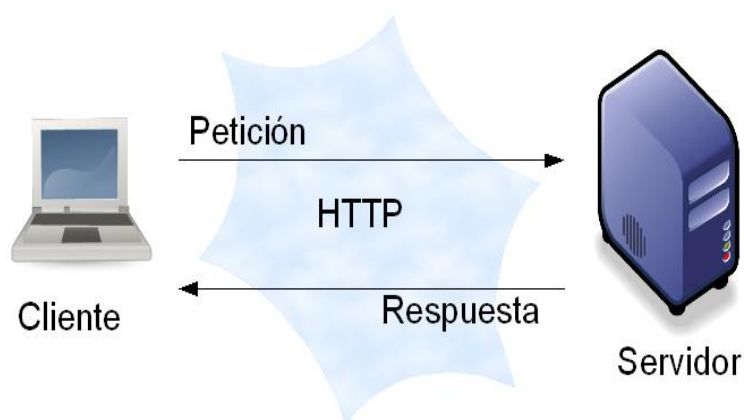
1.8.1. Arquitectura de software

La arquitectura de software se entiende como la estructura de un sistema, define la organización de sus componentes y los principios que guían su diseño y evolución. Según Bass, Clements y Kazman (2021), la arquitectura proporciona un marco que permite tomar decisiones estratégicas sobre cómo se distribuyen las responsabilidades en el sistema, cómo se comunican los módulos y cómo se garantiza la escalabilidad y mantenibilidad del software. Entre los estilos arquitectónicos más utilizados son la arquitectura cliente-servidor y la de microservicios, cada una con enfoques y beneficios particulares.

1.8.1.1. Arquitectura cliente – servidor

La arquitectura cliente-servidor se utiliza para distribuir las funciones de un sistema entre servidores, que procesaron y almacenaron la información, y clientes, que interactuaron con el usuario final. Este modelo permite centralizar la gestión de los datos, mejorar la seguridad, facilitar el mantenimiento y garantizar la disponibilidad de los servicios para múltiples usuarios de manera simultánea (Tanenbaum & van Steen, 2016).

Figura 7. Esquema arquitectura cliente - servidor

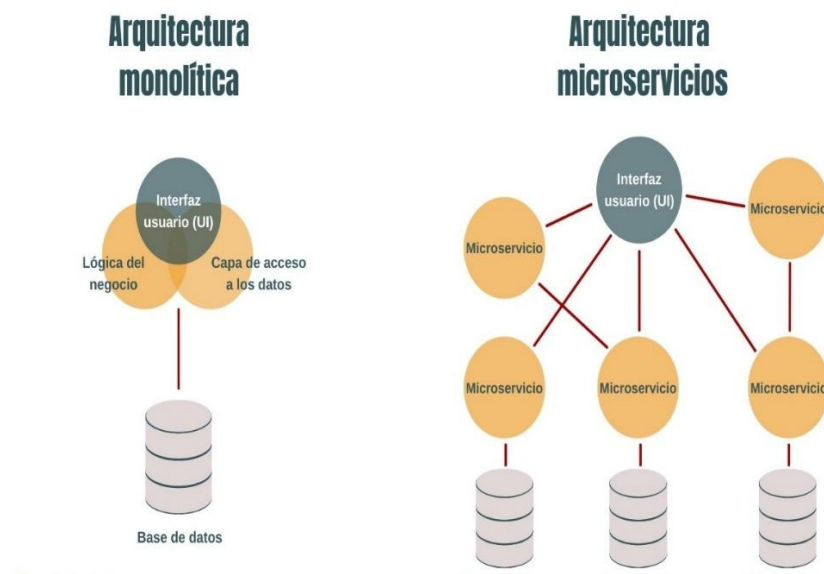


Fuente: EdgarBC, 2013.

1.8.1.2. Arquitectura microservicios

La arquitectura de microservicios consiste en desarrollar una aplicación con una arquitectura basada en conjuntos pequeños, independientes y desplegables de manera individual. Cada microservicio cumple una función específica y se comunica con otros mediante APIs. Newman (2021) señala que este enfoque permite mayor flexibilidad, escalabilidad y resiliencia, al facilitar la actualización y el mantenimiento de cada componente sin afectar al sistema completo.

Figura 8. Esquema arquitectura microservicios



Fuente. Elaboración propia (2025).

1.8.2. Diseño de la base de datos

El diseño de la base de datos se enfocó en organizar, almacenar y administrar la información de manera eficiente. La aplicación de técnicas de normalización permitió reducir la redundancia de datos y garantizar la integridad de la información, mientras que la planificación de relaciones y restricciones aseguró consistencia y fiabilidad en las operaciones del sistema (Elmasri & Navathe, 2017).

1.8.2.1. Diagrama entidad – relación

El diagrama entidad-relación (ER) representa gráficamente las entidades del sistema, sus atributos y las relaciones entre ellas. Este diagrama permite visualizar la estructura de la base de datos y facilita la planificación de su implementación, asegurando que la información se organice de manera lógica y eficiente (Elmasri & Navathe, 2021).

El Diagrama Entidad-Relación (ERD) es una herramienta visual utilizada en el diseño de bases de datos para representar la estructura conceptual de la información (IBM, 2024; UNIR, 2023).

Tabla 2. Conceptos clave de los diagramas entidad-relación

Concepto	Definición
Entidad	Representa un objeto o concepto sobre el cual se almacena información en la base de datos. Ejemplo: Donante, Beneficiario.
Atributo	Característica que describe a una entidad. Ejemplo: nombre, correo electrónico, fecha de donación.
Relación	Asociación entre entidades que define cómo interactúan entre sí. Ejemplo: "realiza" entre Donante y Donación.
Clave primaria	Atributo que identifica de manera única cada registro dentro de una entidad.
Clave foránea	Atributo que permite vincular una entidad con otra mediante su clave primaria.
Cardinalidad	Indica el número de ocurrencias de una entidad que puede asociarse con otra en una relación. Ejemplo: uno a muchos, muchos a muchos.

Fuente. Elaboración propia (2025).

1.8.3. Herramientas y tecnologías utilizadas

Las herramientas y tecnologías utilizadas facilitan el desarrollo y funcionamiento del sistema, proporcionan soporte en la gestión de información, la interacción con los usuarios y la integración de funcionalidades. La elección adecuada de tecnologías permite optimizar recursos, mejorar la eficiencia de un sistema.

1.8.4. Comunicación con APIs

La comunicación con APIs (Application Programming Interfaces) permite la integración de servicios externos y la interoperabilidad entre distintos componentes del sistema. Mediante el uso de protocolos estandarizados, se hace el intercambio de información de manera segura y eficiente, lo que facilita la automatización de procesos, la actualización de datos en tiempo real y la extensión de funcionalidades sin afectar la arquitectura general del sistema. Esta integración contribuye a una mayor flexibilidad y capacidad de adaptación del sistema frente a cambios y nuevas necesidades (Fielding, 2000).

1.9. Inteligencia artificial y aprendizaje automatizado

La inteligencia artificial (IA) permitió automatizar procesos, analizar grandes volúmenes de información y generar soluciones inteligentes para diversos problemas. Su implementación mejoró la eficiencia operativa, facilitó la toma de decisiones

basada en datos y optimizó la interacción con usuarios y sistemas (Russell & Norvig, 2021).

1.9.1. Modelado de lenguaje natural

Los modelos de lenguaje natural (NLP, por sus siglas en inglés) permitieron a los sistemas interpretar, generar y comprender texto o voz humana, facilitando la comunicación entre usuarios y máquinas. Estos modelos mejoraron la interacción, automatizaron respuestas y posibilitaron el análisis de grandes cantidades de información textual de manera precisa (Jurafsky & Martin, 2021).

1.9.2. OpenAI y ChatGPT

La OpenAI API y ChatGPT proporcionaron interfaces para integrar capacidades avanzadas de lenguaje natural en aplicaciones, permitiendo generar texto, responder preguntas y mantener conversaciones coherentes con los usuarios. Su implementación facilitó la automatización de tareas comunicacionales y la creación de asistentes inteligentes adaptables a diferentes contextos (OpenAI, 2023).

ChatGPT y la API de OpenAI puede aprender de interacciones previas, optimizar respuestas según patrones de uso y contribuir a la toma de decisiones basada en datos. Estas capacidades son útiles en entornos donde la atención personalizada, eficiencia y la disponibilidad constante son fundamentales, como en la comunicación con usuarios de servicios digitales (OpenAI, 2025; Smith & Lee, 2024).

1.9.3. Asistentes virtuales

La implementación de asistentes virtuales basados en tecnologías como la OpenAI API y ChatGP, mejora la capacidad de interactuar con los usuarios de manera general, ofreciendo respuestas a preguntas frecuentes, guiando a los usuarios en procesos básicos y automatizando tareas rutinarias. Su implementación permite:

- Mejorar la experiencia de usuario al ofrecer atención inmediata y constante.
- Reducir la carga de trabajo del personal al responder consultas repetitivas.
- Adaptarse a diferentes contextos y tipos de usuarios gracias a su entrenamiento con modelos de lenguaje natural avanzados.

Un asistente virtual funciona como un punto de contacto general con el usuario, siendo flexible y capaz de manejar conversaciones abiertas y no estructuradas.

1.9.4. Bots conversacionales personalizados

Los bots conversacionales personalizados son sistemas de software diseñados para interactuar con usuarios de manera automática, simulando conversaciones humanas

mediante texto o voz. Estos bots se configuran para atender necesidades específicas de una organización o proyecto, ofreciendo respuestas adaptadas al contexto del usuario, guiándolo en procesos, resolviendo dudas frecuentes o facilitando la realización de tareas. Su personalización permite mejorar la experiencia del usuario, incrementar la eficiencia en la atención y optimizar la gestión de información en tiempo real (Huang et al., 2022).

Los bots desarrollados con Rasa son una categoría de asistentes conversacionales más especializada, capaces de manejar flujos de diálogo predefinidos y comprender el contexto de la interacción, los bots se configuran para responder de manera precisa a tareas específicas dentro de un sistema, integrándose con bases de datos y APIs para ofrecer información contextualizada (Bocklisch et al., 2017).

Tabla 3. Aspectos clave de RASA en el desarrollo de chatbots

Categoría	Ventajas de RASA
Código abierto	Es gratuito y de código abierto, lo que permite personalización sin costos de licencias.
Privacidad y control	Los datos permanecen en los servidores propios, sin depender de terceros.
Personalización	Permite adaptar flujos conversacionales complejos y modelos de NLP a necesidades específicas.
Integraciones	Compatible con múltiples plataformas (WhatsApp, Telegram, Messenger, sitios web, etc.).
Comunidad activa	Amplia comunidad y documentación que apoyan el desarrollo y resolución de problemas.
IA Conversacional	Usa técnicas avanzadas de PLN (Procesamiento de Lenguaje Natural) para entender la intención y contexto.
Escalabilidad	Puede crecer con la demanda, integrándose con microservicios y arquitecturas modernas.

Fuente: Elaborado a partir de Rasa Technologies, 2025.

1.10. Tecnologías de desarrollo para plataformas web y móviles

Para el desarrollo de plataformas web y móviles se usan diversas tecnologías las cuales permiten crear aplicaciones funcionales e interactivas. Para el desarrollo web se utilizan lenguajes de marcado, estilos y programación como HTML, CSS y JavaScript, junto con frameworks modernos como React, Angular o Vue.js, que facilitan la construcción de interfaces dinámicas. Para aplicaciones móviles, se emplean enfoques nativos, como Swift para iOS y Kotlin para Android, o soluciones híbridas como React Native y Flutter, que permiten desarrollar aplicaciones para múltiples sistemas operativos a partir de un mismo código. Estas tecnologías no solo

definen la apariencia y funcionalidad de las plataformas, sino que también optimizan la experiencia del usuario y la comunicación con servidores y bases de datos.

1.10.1. Desarrollo web

Se utilizan distintos lenguajes de programación y frameworks para facilitar la estructuración, el diseño y la interacción de las aplicaciones web, los más usados son HTML, CSS, JavaScript y React, que forman la base de muchas aplicaciones modernas.

1.10.1.1. HTML

El lenguaje de marcado de hipertexto (HTML) es el lenguaje que se utiliza para hacer páginas web. Su objetivo principal es establecer la forma y el contenido de un sitio web mediante el uso de etiquetas y elementos. HTML permite agregar texto, imágenes, videos, hipervínculos y otros componentes interactivos. Su desarrollo ha resultado en HTML5, que trae nuevas características como la capacidad de reproducir medios sin necesitar plugins extra y una semántica mejorada para los elementos de la página (Freeman & Robson, 2020).

1.10.1.2. CSS

Las hojas de estilo en cascada, conocidas como CSS, se emplean para establecer cómo se verá y se organizará una página web. A través de CSS, es posible cambiar colores, tipos de letra, dimensiones, espacios y cómo se distribuyen los componentes en la pantalla. La llegada de CSS3 trajo consigo nuevas funciones, como animaciones, transiciones y diseño adaptable mediante media queries. Esto hace que las aplicaciones web sean más visuales y fáciles de usar en teléfonos móviles (Meyer, 2019).

1.10.1.3. JavaScript

JavaScript es un lenguaje de programación interpretado que se utiliza para dotar de interactividad y dinamismo a las aplicaciones web. Permite manipular elementos del DOM, validar formularios, implementar animaciones y establecer comunicación con servidores mediante AJAX, convirtiéndose en el núcleo del desarrollo frontend moderno (Flanagan, 2020).

1.10.1.4. React

React es una biblioteca de JavaScript desarrollada por Meta (Facebook) para la construcción de interfaces de usuario interactivas y reutilizables. Su principal característica es el uso del Virtual DOM, que mejora el rendimiento al actualizar solo los elementos necesarios en lugar de renderizar toda la página. Además, React se basa en una arquitectura de componentes, lo que facilita la modularización del código

y su reutilización. También permite la integración con otras bibliotecas y frameworks, como Redux para la gestión del estado global de la aplicación (Dan Abramov, 2021).

1.10.2. Desarrollo móvil

El desarrollo móvil se enfocó en la creación de aplicaciones capaces de ejecutarse en dispositivos inteligentes, como teléfonos y tabletas, garantizando una experiencia de usuario consistente y eficiente. Se seleccionaron tecnologías multiplataforma que permitieron reutilizar componentes y reducir el tiempo de desarrollo, optimizando recursos y asegurando la integración con funcionalidades nativas de los dispositivos. Esta estrategia facilitó la implementación de interfaces intuitivas, interacciones dinámicas y conectividad con servicios web y APIs externas, contribuyendo a la funcionalidad y usabilidad de las aplicaciones (Nader, 2021; Burns, 2020).

1.10.2.1. React native

React Native se implementó como la tecnología principal para el desarrollo de aplicaciones móviles multiplataforma. Su uso permitió crear interfaces nativas para sistemas operativos iOS y Android a partir de un único código base, optimizando recursos y reduciendo el tiempo de desarrollo. La arquitectura basada en componentes facilitó la reutilización de elementos, el modularidad del código y la integración con APIs nativas de los dispositivos, garantizando un rendimiento eficiente y una experiencia de usuario consistente (Nader, 2021).

1.10.3. Backend y bases de datos

El backend constituye la capa del sistema responsable de la lógica de negocio, el procesamiento de información, la validación de datos y la comunicación con la base de datos. Su implementación permitió garantizar eficiencia, escalabilidad, seguridad y consistencia en el manejo de la información. Además, facilitó la integración con el frontend y las aplicaciones móviles, asegurando que los usuarios pudieran interactuar con la plataforma de manera fluida y confiable (Tilkov & Vinoski, 2010).

1.10.3.1. Node.js

Node.js se utilizó como entorno de ejecución de JavaScript en el servidor. Su arquitectura basada en eventos y modelo asíncrono permitió gestionar múltiples solicitudes concurrentes sin bloquear el sistema, optimizando el rendimiento del backend. Esta característica resultó esencial para aplicaciones que requerían respuestas rápidas a usuarios en tiempo real, así como la ejecución eficiente de procesos complejos en el servidor (Tilkov & Vinoski, 2010).

1.10.3.2. Express

Express se implementó como framework de Node.js para simplificar la construcción de aplicaciones web y APIs REST. Facilitó la gestión de rutas, middleware y control de peticiones HTTP, lo que permitió crear servicios web escalables, mantenibles y seguros. Además, Express posibilitó la integración con bases de datos y otros servicios externos de manera eficiente, optimizando la estructura y modularidad del backend (Holowaychuk, 2014).

1.10.3.3. Prisma ORM

Prisma ORM se empleó como una herramienta de mapeo objeto-relacional (ORM) para interactuar con la base de datos. Su implementación permitió mapear entidades del código a tablas de la base de datos, automatizar consultas complejas, garantizar la integridad referencial de los datos y reducir errores de programación. Esto mejoró la productividad de los desarrolladores y facilitó el mantenimiento de la plataforma, asegurando consistencia en las operaciones de lectura y escritura (Prisma, 2023).

Figura 9. Ilustración prisma como intermediario en el desarrollo de aplicaciones con bases de datos modernas



Fuente: Elaborado a partir de Prisma, s. f.

1.10.3.4. Bases de datos relacionales

Las bases de datos relacionales se seleccionaron para estructurar la información de manera organizada y consistente. Esta arquitectura permitió almacenar datos en tablas relacionadas mediante claves primarias y foráneas, asegurando integridad y coherencia en la información. Además, soportó consultas complejas, transacciones y generación de reportes confiables, lo que permitió a la plataforma gestionar grandes volúmenes de datos de manera segura y eficiente (Elmasri & Navathe, 2020).

1.10.3.5. PostgreSQL

PostgreSQL se implementó como el sistema de gestión de bases de datos relacional, seleccionado por su robustez, escalabilidad, seguridad y compatibilidad con estándares SQL. Su implementación permitió ejecutar transacciones, crear índices y procedimientos almacenados, y manejar grandes volúmenes de datos de manera confiable. Además, PostgreSQL facilitó la integración con Prisma ORM y el backend desarrollado en Node.js y Express, garantizando la consistencia y disponibilidad de la información (Momjian, 2020).

1.10.3.6. API web service

"Una API web service permite la comunicación entre aplicaciones mediante protocolos estándar de Internet, favoreciendo la interoperabilidad" (Fielding, 2000).

- **Protocolos de Internet**

Utilizan protocolos de comunicación estándar, como HTTP (Hypertext Transfer Protocol) y SOAP (Simple Object Access Protocol), para permitir la interacción entre aplicaciones.

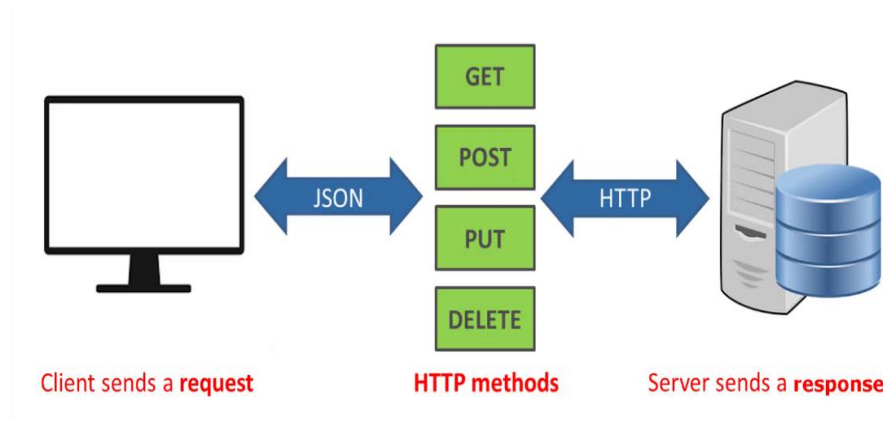
- **Interoperabilidad**

Se basan en estándares abiertos y protocolos comunes, lo que facilita la interoperabilidad entre diferentes sistemas y plataformas.

- **Descripción de servicios**

Se describen mediante lenguajes de descripción de servicios, como WSDL (Web Services Description Language) o OpenAPI, que especifican cómo se accede y utiliza el servicio.

Figura 10. API web service



Fuente: Wahab2, 2023.

1.11. Optimización para la experiencia del usuario

La optimización de la experiencia del usuario es la forma en que el usuario interactúa con una aplicación, busca que sea visualmente atractiva, intuitiva y fácil de usar, también busca facilitar la navegación reduciendo tiempos de carga y que el diseño se pueda adaptar a diferentes dispositivos, que el usuario se familiarice rápido y cumpla con los objetivos sin problemas.

1.11.1. Principios para la experiencia del usuario en plataformas digitales

El diseño de experiencia de usuario (UX) se usa para optimizar la interacción entre las personas y los productos digitales, donde se da prioridad a la satisfacción y facilidad de uso, para lograr esto se tiene una serie de principios que guían el desarrollo de interfaces y funcionalidades centradas en el usuario (Interaction Design Foundation, 2024).

- **Claridad**

Los usuarios nunca deberían tener que adivinar qué significa algo ni cómo funciona. Las interfaces claras comunican el propósito, la función y los pasos siguientes sin necesidad de explicaciones.

- **Simplicidad**

Las mejores experiencias digitales se sienten fluidas , reduciendo la carga cognitiva al mostrar solo lo necesario en cada paso. El objetivo es evitar situaciones que abruman a los usuarios con demasiada información o demasiadas opciones.

- **Usabilidad**

Usabilidad significa diseñar teniendo en cuenta cómo se comportan realmente las personas, teniendo en cuenta las limitaciones de atención, memoria y paciencia.

1.11.3. Técnicas de diseño responsive y adaptabilidad a diferentes dispositivos

El diseño responsive permite que las plataformas se adapten automáticamente a diferentes tamaños de pantalla y dispositivos, garantizando una experiencia consistente en computadoras, tablets y dispositivos móviles. Esto incluye la reorganización de contenidos, ajuste de tipografías, escalado de imágenes y optimización de elementos interactivos para cada dispositivo (Baymard Institute, 2024).

CAPÍTULO II

INGENIERÍA DEL PROYECTO

En este capítulo presenta la ingeniería del proyecto lo que respalda el desarrollo de la solución propuesta, en esta etapa se planifica, organiza y diseña de manera técnica el sistema, también se definen los requerimientos funcionales y no funcionales, la arquitectura y los diagramas UML los procesos de desarrollo, las pruebas y los recursos necesarios para garantizar una correcta implementación.

2.1. Metodología del desarrollo

Para el desarrollo de la plataforma digital se aplicó la metodología ágil SCRUM la cual permite gestionar el proyecto de manera flexible, mediante ciclos cortos llamados sprints, esta metodología facilita el avance progresivo de avances funcionales, la revisión continua de resultados.

El proyecto tuvo una duración total de seis meses, desde el análisis de requerimientos hasta la implementación final de la plataforma digital compuesta por una página web y una aplicación móvil.

2.1.1. Roles del equipo SCRUM

Los roles SCRUM establecen las responsabilidades y funciones necesarias, estos roles permiten organizar el trabajo, priorizar tareas y mantener una comunicación continua durante cada sprint. En este caso, las funciones fueron asumidas por una sola persona, quien desempeñó simultáneamente los tres roles principales: Product Owner, Scrum Master y Equipo de desarrollo, aplicando los principios de SCRUM de manera adaptada al contexto del proyecto.

Tabla 4. Roles del equipo SCRUM

Rol	Responsable	Descripción de funciones
Product Owner	Jhoselin Diana Céspedes Brañez	Definió los requerimientos del sistema, priorizó las funcionalidades y validó los avances con base en los objetivos del proyecto.
Scrum Master	Jhoselin Diana Céspedes Brañez	Se encargó de aplicar la metodología SCRUM, gestionar los tiempos, organizar las tareas y resolver los inconvenientes surgidos durante el desarrollo.

Equipo de desarrollo	Jhoselin Diana Céspedes Brañez	Realizó el diseño, programación, pruebas y documentación técnica de la plataforma digital.
----------------------	--------------------------------	--

Fuente. Elaboración propia (2025).

2.1.2. Artefactos del proceso SCRUM

Los artefactos de la metodología SCRUM representa aquellos elementos tangibles para gestionar y dar un correcto seguimiento al progreso del proyecto.

Se incluye el Producto Backlog, contiene la lista de funcionalidades priorizadas las cuales son necesarias para cumplir con los objetivos establecidos, la tabla representa el listado de requerimientos organizados según la prioridad y funcionalidad.

Tabla 5. Artefactos del proceso SCRUM

ID	Funcionalidad	Prioridad	Descripción
PB-01	Registro de donaciones	Alta	Permitir realizar donaciones en línea desde la web.
PB-02	Módulo de beneficiarios	Alta	Registrar, consultar y actualizar los datos de los beneficiarios desde la app móvil.
PB-03	Publicación de campañas	Media	Mostrar las campañas activas y casos de recuperación.
PB-04	Gestión de usuarios	Media	Controlar accesos y roles dentro de la plataforma.
PB-05	Recordatorios de citas	Alta	Enviar notificaciones de citas a los beneficiarios.

Fuente. Elaboración propia (2025).

2.1.3. Cronograma de sprints

El cronograma de sprints muestra la planificación del trabajo que se realiza durante el desarrollo del sistema.

Cada sprint representa un periodo de tiempo en el que se ejecutaron actividades específicas, desde el análisis inicial hasta la implementación final de la plataforma.

En la siguiente tabla se detallan los objetivos, las actividades principales y los entregables correspondientes a cada sprint.

Tabla 6. Cronograma de sprints

Sprint	Duración	Objetivo principal	Actividades realizadas	Entregables
Sprint 1: Análisis y diseño	6 semanas	Levantamiento de requerimientos y diseño del sistema.	Recolección de información, modelado de base de datos, diagramas UML, diseño de interfaz.	Documentación de análisis y prototipos iniciales.

Sprint 2: Desarrollo de la página web	6 semanas	Implementación de la web institucional.	Desarrollo de la interfaz, módulo de campañas y formulario de donaciones.	Página web funcional con sistema de donaciones.
Sprint 3: Desarrollo de la aplicación móvil	6 semanas	Desarrollo del sistema móvil para beneficiarios.	Registro de beneficiarios, citas y recordatorios automáticos.	Aplicación móvil funcional.
Sprint 4: Pruebas e implementación final	6 semanas	Validación e integración completa del sistema.	Pruebas unitarias, de integración y usabilidad, corrección de errores y despliegue final.	Plataforma digital lista para uso institucional.

Fuente. Elaboración propia (2025).

2.1.4. Trazabilidad entre objetivos, requerimientos y módulos

La trazabilidad establece la relación directa entre los objetivos del proyecto, los requerimientos que fueron identificados y los módulos desarrollados, esto garantiza la coherencia entre las funcionalidades implementadas y los objetivos planteados, facilitando el control del cumplimiento de los objetivos.

En la siguiente tabla se muestra la relación entre los objetivos del sistema, los requerimientos que los cumplen y los módulos donde fueron aplicados.

Tabla 7. Trazabilidad entre objetivos, requerimientos y módulos

Objetivo	Requerimiento asociado	Módulo del sistema
Fortalecer la presencia digital de la fundación	Mostrar campañas e historias de recuperación	Página web
Facilitar la recepción de donaciones	Permitir donaciones en línea	Módulo de donaciones
Optimizar el seguimiento de beneficiarios	Registrar citas, apoyos y recordatorios	Aplicación móvil
Mejorar la comunicación con los beneficiarios	Notificaciones y acceso desde el celular	App móvil integrada con base de datos

Fuente. Elaboración propia (2025).

2.2. Requerimientos

Los requerimientos del software se dividen entre funcionales (RF) donde se describen las funciones específicas que debe cumplir el sistema y no funcionales (RFN) estas establecen las condiciones de calidad, seguridad, rendimiento entre otros aspectos claves para un correcto funcionamiento.

2.2.1. Requerimientos funcionales (RF)

Requerimientos funcionales web informativa, las tablas que se presentaran a continuación describen cada requerimiento que defines las funcionalidades que

permiten gestionar y mostrar información relevante de manera organizada y accesible.

Cada tabla detalla el ID, nombre, descripción y prioridad del requerimiento, entre las funcionalidades principales se incluyen:

- **Tabla 8:** Mostrar la barra de navegación para facilitar el acceso a todas las secciones.
- **Tabla 9:** Publicar artículos del blog para difundir información y noticias relevantes.
- **Tabla 9:** Visualizar eventos próximos, incluyendo fecha y hora.
- **Tabla 10:** Exponer casos recuperados mediante testimonios.
- **Tabla 11:** Informar sobre campañas activas o pasadas de la organización.
- **Tabla 12:** Ofrecer un formulario de contacto para que los usuarios puedan enviar consultas.
- **Tabla 13:** Incluir el pie de página con enlaces útiles y datos de contacto, mejorando la accesibilidad del sitio.

Tabla 8. RF1 Mostrar barra de navegación web informativa

Mostrar barra de navegación web informativa			
Elemento	Descripción		
ID	RF1		
Nombre	Mostrar barra de navegación		
Descripción	El sistema debe mostrar un menú de navegación para acceder a todas las secciones		
Prioridad	Alta/Esencial	Media/Deseado	Baja/Opcional

Fuente: Elaboración propia (2025).

Tabla 9. RF2 publicar artículos del blog

Publicar artículos del blog	
Elemento	Descripción
ID	RF2

Nombre	Publicar artículos del blog		
Descripción	El sistema debe permitir visualizar artículos informativos o noticias en la sección Blog.		
Prioridad	Alta/Esencial	Media/Deseado	Baja/Opcional

Fuente: Elaboración propia (2025).

Tabla 10. RF3 visualizar eventos

Visualizar eventos			
Elemento	Descripción		
ID	RF3		
Nombre	Visualizar eventos		
Descripción	El sistema debe mostrar información sobre eventos próximos, fecha y hora del evento		
Prioridad	Alta/Esencial	Media/Deseado	Baja/Opcional

Fuente: Elaboración propia (2025).

Tabla 11. RF4 exponer casos recuperados

Exponer casos recuperados			
Elemento	Descripción		
ID	RF4		
Nombre	Exponer casos recuperados		
Descripción	El sistema debe mostrar testimonios de casos de recuperación		
Prioridad	Alta/Esencial	Media/Deseado	Baja/Opcional

Fuente: Elaboración propia (2025).

Tabla 12. RF5 informar sobre campañas

Informar sobre campañas	
Elemento	Descripción

ID	RF5		
Nombre	Informar sobre campañas		
Descripción	El sistema debe mostrar campañas activas o pasadas realizadas por la organización.		
Prioridad	Alta/Esencial	Media/Deseado	Baja/Opcional

Fuente: Elaboración propia (2025).

Tabla 13. RF6 ofrecer formulario de contacto

Ofrecer formulario de contacto			
Elemento	Descripción		
ID	RF6		
Nombre	Ofrecer formulario de contacto		
Descripción	El sistema debe permitir enviar consultas mediante un formulario de contacto		
Prioridad	Alta/Esencial	Media/Deseado	Baja/Opcional

Fuente: Elaboración propia (2025).

Tabla 14. RF7 incluir pie de página

Incluir pie de página			
Elemento	Descripción		
ID	RF7		
Nombre	Incluir pie de página		
Descripción	El sistema debe mostrar un pie de página con enlaces útiles y datos de contacto como, facebook, instagran, whatsapp, etc.		
Prioridad	Alta/Esencial	Media/Deseado	Baja/Opcional

Fuente: Elaboración propia (2025).

Requerimientos funcionales web administrativa, las tablas que se presentaran a continuación describen cada requerimiento que defines las funcionalidades que

permiten gestionar y mostrar información relevante de manera organizada y accesible.

Cada tabla detalla el ID, nombre, descripción y prioridad del requerimiento, entre las funcionalidades principales se incluyen:

- **Tabla 15 – RF1:** Gestión de eventos: agregar, editar, eliminar y buscar eventos.
- **Tabla 16 – RF2:** Exportar eventos en formatos Excel y PDF.
- **Tabla 17 – RF3:** Gestión de campañas: agregar, editar, eliminar y buscar campañas.
- **Tabla 18 – RF4:** Exportar campañas en formatos Excel y PDF.
- **Tabla 19 – RF5:** Publicar noticias con título, contenido e imagen.
- **Tabla 20 – RF6:** Gestión de usuarios: editar y eliminar usuarios existentes.
- **Tabla 21 – RF7:** Registrar nuevos usuarios con roles definidos.
- **Tabla 22 – RF8:** Registrar usuarios por roles específicos (psicólogo, trabajador social, beneficiario).
- **Tabla 23 – RF9:** Gestión de usuarios: modificar datos de usuarios existentes.
- **Tabla 24 – RF10:** Cambiar contraseña de usuarios.
- **Tabla 25 – RF11:** Habilitar donaciones mediante código QR.
- **Tabla 26 – RF12:** Ver historial de donaciones.
- **Tabla 27 – RF13:** Exportar donaciones a Excel o PDF.
- **Tabla 28 – RF14:** Registrar pacientes con información personal y médica.
- **Tabla 29 – RF15:** Gestión de pacientes: editar, eliminar y buscar pacientes.
- **Tabla 30 – RF16:** Registrar citas de psicólogo.
- **Tabla 31 – RF17:** Registrar historial de tratamiento de pacientes.
- **Tabla 32 – RF18:** Registrar expediente social del beneficiario.
- **Tabla 33 – RF19:** Registrar evaluación de vulnerabilidad de beneficiarios.
- **Tabla 34 – RF20:** Registrar derivación a un profesional.
- **Tabla 35 – RF21:** Registrar donación a paciente.
- **Tabla 36 – RF22:** Visualización del historial clínico de pacientes.
- **Tabla 37 – RF23:** Calendario de citas en móvil.
- **Tabla 38 – RF24:** Notificaciones automáticas de citas en móvil.
- **Tabla 39 – RF25:** Chat interactivo para consultas sobre cáncer infantil.

Tabla 15. RF1 gestión de eventos

Gestión de eventos

Elemento	Descripción		
ID	RF1		
Nombre	Gestión de eventos		
Descripción	Módulo eventos: El sistema debe permitir al usuario gestionar eventos, incluyendo las operaciones de agregar, editar, eliminar y buscar eventos por palabras clave en el título o descripción..		
Prioridad	Alta/Esencial	Media/Deseado	Baja/Opcional

Fuente: Elaboración propia (2025).

Tabla 16. RF2 exportar eventos

Exportar eventos			
Elemento	Descripción		
ID	RF2		
Nombre	Exportar eventos		
Descripción	Módulo eventos: El sistema debe generar y descargar un archivo con la lista de eventos registrados en formatos Excel y PDF.		
Prioridad	Alta/Esencial	Media/Deseado	Baja/Opcional

Fuente: Elaboración propia (2025).

Tabla 17. RF3 gestión de campañas

Gestión de campañas			
Elemento	Descripción		
ID	RF3		
Nombre	Gestión de campañas		
Descripción	Módulo campañas: El sistema debe permitir gestionar campañas incluyendo las operaciones de agregar, editar, eliminar y busca.		
Prioridad	Alta/Esencial	Media/Deseado	Baja/Opcional

Fuente: Elaboración propia (2025).

Tabla 18. RF4 exportar campañas

Exportar campañas			
Elemento	Descripción		
ID	RF4		
Nombre	Exportar campañas a Excel		
Descripción	Módulo campañas: El sistema debe generar y descargar un archivo con la información de campañas en formatos Excel y PDF.		
Prioridad	Alta/Esencial	Media/Deseado	Baja/Opcional

Fuente: Elaboración propia (2025).

Tabla 19. RF5 publicar noticia

Publicar noticia			
Elemento	Descripción		
ID	RF5		
Nombre	Publicar noticia		
Descripción	Módulo noticias: El sistema debe permitir publicar noticias con título, contenido e imagen.		
Prioridad	Alta/Esencial	Media/Deseado	Baja/Opcional

Fuente: Elaboración propia (2025).

Tabla 20. RF6 gestión de noticias

Gestión de noticias			
Elemento	Descripción		
ID	RF6		
Nombre	Gestión de usuarios		
Descripción	Módulo usuario: El sistema debe permitir editar y eliminar usuarios existentes en el sistema.		

Prioridad	Alta/Esencial	Media/Deseado	Baja/Opcional
------------------	----------------------	---------------	---------------

Fuente: Elaboración propia (2025).

Tabla 21. RF7 registrar usuario

Registrar usuario			
Elemento	Descripción		
ID	RF7		
Nombre	Registrar usuario		
Descripción	Módulo usuario: El sistema debe permitir registrar nuevos usuarios con roles definidos.		
Prioridad	Alta/Esencial	Media/Deseado	Baja/Opcional

Fuente: Elaboración propia (2025).

Tabla 22. RF8 registrar usuario por roles

Registrar usuario por roles			
Elemento	Descripción		
ID	RF8		
Nombre	Registrar usuario por roles		
Descripción	Módulo usuario: El sistema debe permitir registrar nuevos usuarios con roles definidos. - Psicólogo - Trabajador social - Beneficiario		
Prioridad	Alta/Esencial	Media/Deseado	Baja/Opcional

Fuente: Elaboración propia (2025).

Tabla 23. RF9 gestión de usuarios

Gestión de usuarios	
Elemento	Descripción
ID	RF9
Nombre	Gestión de usuarios

Descripción	Módulo usuario: El sistema debe permitir modificar los datos de usuarios existentes.		
Prioridad	Alta/Esencial	Media/Deseado	Baja/Opcional

Fuente: Elaboración propia (2025).

Tabla 24. RF10 cambiar contraseña

Cambiar contraseña			
Elemento	Descripción		
ID	RF10		
Nombre	Cambiar contraseña		
Descripción	Módulo usuario: El sistema debe permitir a los usuarios cambiar su contraseña.		
Prioridad	Alta/Esencial	Media/Deseado	Baja/Opcional

Fuente: Elaboración propia (2025).

Tabla 25. RF11 habilitar donación con QR

Habilitar donación con QR			
Elemento	Descripción		
ID	RF11		
Nombre	Donar mediante código QR		
Descripción	Módulo donaciones: El sistema debe mostrar un formulario y, al completarlo, generar un código QR para donar.		
Prioridad	Alta/Esencial	Media/Deseado	Baja/Opcional

Fuente: Elaboración propia (2025).

Tabla 26. RF12 ver historial de donaciones

Ver historial de donaciones	
Elemento	Descripción

ID	RF12		
Nombre	Ver historial de donaciones		
Descripción	Módulo donaciones: El sistema debe mostrar un listado de todas las donaciones realizadas.		
Prioridad	Alta/Esencial	Media/Deseado	Baja/Opcional

Fuente: Elaboración propia (2025).

Tabla 27. RF13 exportar donaciones a Excel/PDF

Exportar donaciones a Excel/PDF			
Elemento	Descripción		
ID	RF13		
Nombre	Exportar donaciones a Excel/PDF		
Descripción	Módulo donaciones: El sistema debe permitir exportar la información de donaciones en formato Excel o PDF.		
Prioridad	Alta/Esencial	Media/Deseado	Baja/Opcional

Fuente: Elaboración propia (2025).

Tabla 28. RF14 registrar paciente

Registrar paciente			
Elemento	Descripción		
ID	RF14		
Nombre	Registrar paciente		
Descripción	Módulo pacientes: El sistema debe permitir registrar información personal y médica de un paciente, debe enlazarse con el registro del padre de familia		
Prioridad	Alta/Esencial	Media/Deseado	Baja/Opcional

Fuente: Elaboración propia (2025).

Tabla 29. RF15 gestión de pacientes

Gestión de pacientes			
Elemento	Descripción		
ID	RF15		
Nombre	Gestión de pacientes		
Descripción	Módulo pacientes: El sistema debe permitir editar, eliminar y buscar pacientes registrados. La búsqueda debe poder realizarse por nombre o número de historia clínica.		
Prioridad	Alta/Esencial	Media/Deseado	Baja/Opcional

Fuente: Elaboración propia (2025).

Tabla 30. RF16 registrar cita de psicólogo

Registrar cita de psicólogo			
Elemento	Descripción		
ID	RF16		
Nombre	Registrar cita de psicólogo		
Descripción	Módulo psicólogo: El sistema debe permitir registrar citas con el psicólogo, especificando fecha, hora y paciente.		
Prioridad	Alta/Esencial	Media/Deseado	Baja/Opcional

Fuente: Elaboración propia (2025).

Tabla 31. RF17 registrar historial de tratamiento

Registrar historial de tratamiento			
Elemento	Descripción		
ID	RF17		
Nombre	Registrar historial de tratamiento		
Descripción	Módulo psicólogo: El sistema debe permitir registrar información sobre la evolución del paciente mientras esté en tratamiento.		

Prioridad	Alta/Esencial	Media/Deseado	Baja/Opcional
------------------	----------------------	---------------	---------------

Fuente: Elaboración propia (2025).

Tabla 32. RF18 registrar expediente social

Registrar expediente social			
Elemento	Descripción		
ID	RF18		
Nombre	Registrar expediente social		
Descripción	Módulo: Trabajador social El sistema debe permitir registrar datos del beneficiario como ingresos y nivel de vulnerabilidad.		
Prioridad	Alta/Esencial	Media/Deseado	Baja/Opcional

Fuente: Elaboración propia (2025).

Tabla 33. RF19 registrar evaluación de vulnerabilidad

Registrar evaluación de vulnerabilidad			
Elemento	Descripción		
ID	RF19		
Nombre	Registrar evaluación de vulnerabilidad		
Descripción	Módulo trabajador social: El sistema debe permitir seleccionar un beneficiario y registrar su evaluación según la escala de vulnerabilidad (0-25 por categoría: situación económica, vivienda, salud familiar, educación y desarrollo), calculando el puntaje total (0-100).		
Prioridad	Alta/Esencial	Media/Deseado	Baja/Opcional

Fuente: Elaboración propia (2025).

Tabla 34. RF20 registrar derivación

Registrar derivación	
Elemento	Descripción
ID	RF20

Nombre	Registrar derivación		
Descripción	Módulo trabajador social: El sistema debe permitir derivar un beneficiario a un profesional, registrando fechas, motivo y observaciones.		
Prioridad	Alta/Esencial	Media/Deseado	Baja/Opcional

Fuente: Elaboración propia (2025).

Tabla 35. RF21 registrar donación a paciente

Registrar donación a paciente			
Elemento	Descripción		
ID	RF21		
Nombre	Registrar donación a paciente		
Descripción	Módulo trabajador social: El sistema debe permitir registrar donaciones a pacientes, incluyendo concepto, monto total, monto cubierto por la fundación y la familia.		
Prioridad	Alta/Esencial	Media/Deseado	Baja/Opcional

Fuente: Elaboración propia (2025).

Tabla 36. RF22 visualización del historial clínico

Visualización del historial clínico			
Elemento	Descripción		
ID	RF22		
Nombre	Visualización del historial clínico		
Descripción	El sistema debe mostrar un historial consolidado de diagnósticos, tratamientos, sesiones, citas y gastos médicos del paciente, ordenados cronológicamente.		
Prioridad	Alta/Esencial	Media/Deseado	Baja/Opcional

Fuente: Elaboración propia (2025).

Tabla 37. RF23 calendario de citas

Calendario de citas			
Elemento	Descripción		
ID	RF23		
Nombre	Calendario de citas- movil		
Descripción	El sistema debe mostrar las citas del psicólogo y trabajador social en el calendario, al hacer clip mostrar detalles de la cita (hora, fecha, seguimiento, etc).		
Prioridad	Alta/Esencial	Media/Deseado	Baja/Opcional

Fuente: Elaboración propia (2025).

Tabla 38. RF24 notificaciones de citas

Notificaciones de citas			
Elemento	Descripcion		
ID	RF24		
Nombre	Notificaciones de citas -movil		
Descripcion	El sistema debe poder enviar notificaiones 24 horas antes de la cita, hacer recordatorios automaticos.		
Prioridad	Alta/Esencial	Media/Deseado	Baja/Opcional

Fuente: Elaboración propia (2025).

Tabla 39. RF25 chat interactivo

Chat interactivo	
Elemento	Descripción
ID	RF25
Nombre	Chat interactivo
Descripción	El sistema debe proporcionar un chat en el que el usuario pueda escribir o dictar preguntas relacionadas con el cáncer infantil, y recibir respuestas

	automáticas con información clara y estructurada.		
Prioridad	Alta/Esencial	Media/Deseado	Baja/Opcional

Fuente: Elaboración propia (2025).

2.2.2. Requerimientos no funcionales (NF)

Tabla 40. RNF - 01

RNF - 01			
Tipo de elemento	Requisito		
Nombre	Interfaz intuitiva		
Descripción	El sistema debe contar con una interfaz intuitiva, amigable y adaptada para usuarios no técnicos.		
Prioridad	Alta/Esencial	Media/Deseado	Baja/Opcional

Fuente: Elaboración propia (2025).

Tabla 41. RNF - 02

RNF - 02			
Tipo de elemento	Requisito		
Nombre	Acceso desde navegadores comunes		
Descripción	La aplicación debe ser accesible desde navegadores web comunes (Chrome, Firefox, Edge, Safari).		
Prioridad	Alta/Esencial	Media/Deseado	Baja/Opcional

Fuente: Elaboración propia (2025).

Tabla 42. RNF - 03

RNF - 03			
----------	--	--	--

Tipo de elemento	Requisito		
Nombre	Diseño adaptable (responsive)		
Descripción	El sistema debe tener un diseño adaptable para funcionar correctamente en computadoras y dispositivos móviles.		
Prioridad	Alta/Esencial	Media/Deseado	Baja/Opcional

Fuente: Elaboración propia (2025).

Tabla 43. RNF - 04

RNF - 04			
Tipo de elemento	Seguridad		
Nombre	Autenticación para módulos administrativos		
Descripción	El sistema debe exigir autenticación para acceder a los módulos administrativos.		
Prioridad	Alta/Esencial	Media/Deseado	Baja/Opcional

Fuente: Elaboración propia (2025).

Tabla 44. RNF - 05

RNF - 05			
Tipo de elemento	Seguridad		
Nombre	Contraseñas cifradas		
Descripción	Las contraseñas deben almacenarse cifradas en la base de datos..		
Prioridad	Alta/Esencial	Media/Deseado	Baja/Opcional

Fuente: Elaboración propia (2025).

Tabla 45. RNF - 05

RNF - 06			
-----------------	--	--	--

Tipo de elemento	Seguridad		
Nombre	Control de accesos críticos		
Descripción	Solo los usuarios con permisos adecuados podrán acceder a funciones críticas como eliminar datos.		
Prioridad	Alta/Esencial	Media/Deseado	Baja/Opcional

Fuente: Elaboración propia (2025).

Tabla 46. RNF - 07

RNF - 07			
Tipo de elemento	Rendimiento y eficiencia		
Nombre	Tiempo de carga de páginas		
Descripción	El sistema debe permitir la carga de las páginas en menos de 3 segundos en condiciones normales.		
Prioridad	Alta/Esencial	Media/Deseado	Baja/Opcional

Fuente: Elaboración propia (2025).

Tabla 47. RNF - 08

RNF - 08			
Tipo de elemento	Disponibilidad y confiabilidad		
Nombre	Disponibilidad del sistema		
Descripción	El sistema debe estar disponible al menos el 99% del tiempo durante el horario de atención.		
Prioridad	Alta/Esencial	Media/Deseado	Baja/Opcional

Fuente: Elaboración propia (2025).

Tabla 48. RNF - 09

RNF - 09			
-----------------	--	--	--

Tipo de elemento	Mantenibilidad y escalabilidad		
Nombre	Arquitectura escalable		
Descripción	El sistema debe permitir agregar nuevos módulos sin necesidad de rediseñar toda la arquitectura.		
Prioridad	Alta/Esencial	Media/Deseado	Baja/Opcional

Fuente: Elaboración propia (2025).

2.3. Diseño del proyecto

El diseño del proyecto es la etapa donde se define la estructura organizacional también técnica del sistema, en esta sección se muestra la arquitectura del software, también se definen los diagramas UML junto con sus respectivos roles.

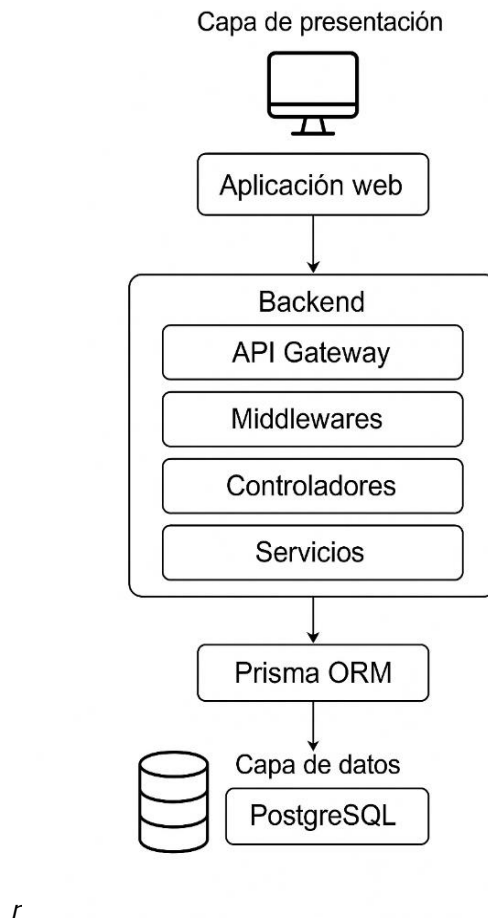
2.2.1. Arquitectura del software (Cliente - Servidor)

Para la realización del proyecto, se utilizó una arquitectura Cliente-Servidor, que permite la separación de la lógica empresarial y de administración de datos (backend) de la interfaz del usuario (frontend), lo que favorece un sistema escalable, mantenible y modular.

El cliente representado por una aplicación web desarrollada con React y Vite, junto con una aplicación móvil en React Native, donde ambas interfaces permiten que los usuarios, de acuerdo a su rol, interactúen con el sistema y accedan a funcionalidades como la gestión de beneficiarios, registro, visualización de campañas y contacto con la fundación. Estas aplicaciones envían solicitudes HTTP al servidor para acceder a la información y ejecutar los procesos.

El servidor se creó con Node.js y Express, las funcionalidades se encuentran estructuradas en rutas como middleware, controladores y servicios para la gestión de procesos fundamentales como autenticación y control de tratamientos, esto llega a ser trabajo del backend, también prisma ORM se utiliza para acceder a la base de datos PostgreSQL lo cual permite hacer consultas confiables para establecer relaciones adecuadamente y mantener el control de integridad.

Figura 11. Modelo arquitectura cliente servidor



Fuente: Elaboración propia (2025).

2.3.2. Diagramas UML

Los diagramas UML se usan para realizar representaciones visuales lo cual permite modelar diferentes aspectos del sistema para facilitar la comprensión y diseño, cada tipo de diagrama cumple una función específica como mostrar la estructura y el comportamiento físico del sistema, esto contribuye en un desarrollo organizado y eficiente.

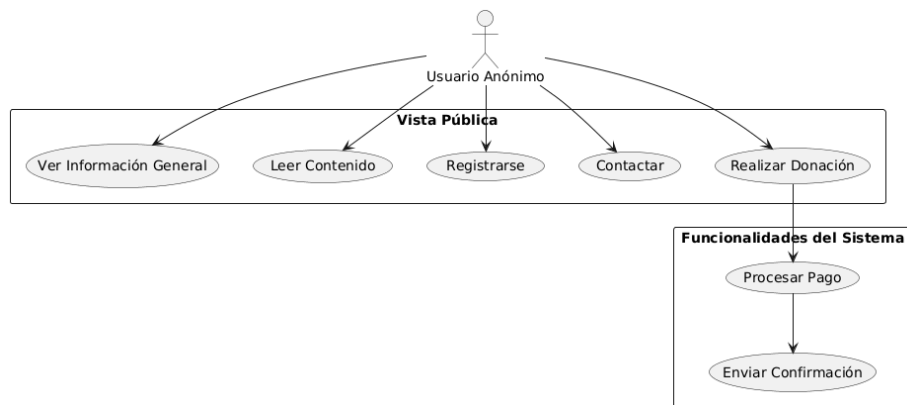
2.3.2.1. Diagrama de casos de uso

Los diagramas de casos de uso describen las funciones generales y el alcance de un sistema. Estos diagramas también identifican las interacciones entre el sistema y sus actores.

El sistema está conformado por diferentes tipos de usuarios, cada uno con permisos y funcionalidades específicas según su rol dentro de la plataforma:

- Usuario anónimo: Corresponde al visitante general que accede a la página web, accede a la información general y leer contenido (noticias, blog, ect), contactar y realizar donaciones sin necesidad de iniciar sesión.

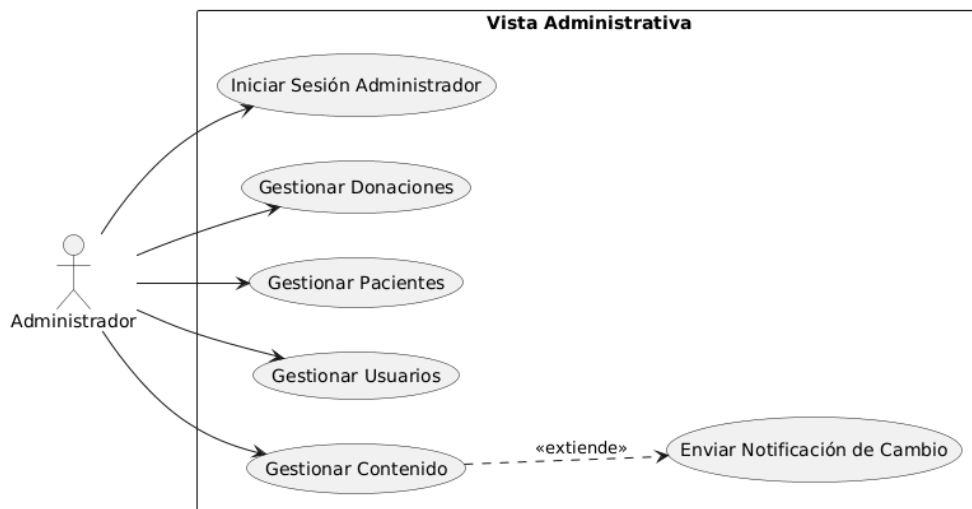
Figura 12. Diagrama de casos de uso usuario anónimo



Fuente: Elaboración propia (2025).

- Administrador: Tiene acceso exclusivo a la parte web administrativa del sistema, es el único que puede iniciar sesión como administrador y gestionar contenido, crear usuarios, ver historial de donaciones, crear contenido.

Figura 13. Diagrama de casos de uso administrador



Fuente: Elaboración propia (2025).

- Psicólogo: Accede a la plataforma web mediante un inicio de sesión, accede a la vista de psicólogo, donde puede ver la agenda de sesiones, registrar sesiones y agregar notas de sesión.

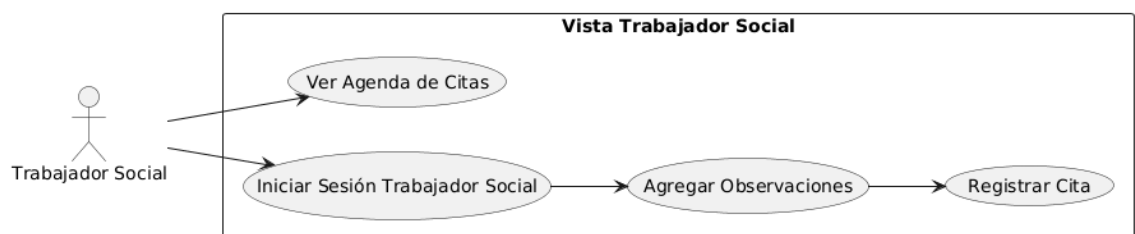
Figura 14. Diagrama de casos de uso psicólogo



Fuente: Elaboración propia (2025).

- Trabajador social: Inicia sesión en la plataforma web para acceder a la vista correspondiente a su rol lo cual le da acceso a ver la agenda de citas, registrar citas, consultar el historial de pacientes y agregar observaciones.

Figura 15. Diagrama de cosos de uso trabajador social

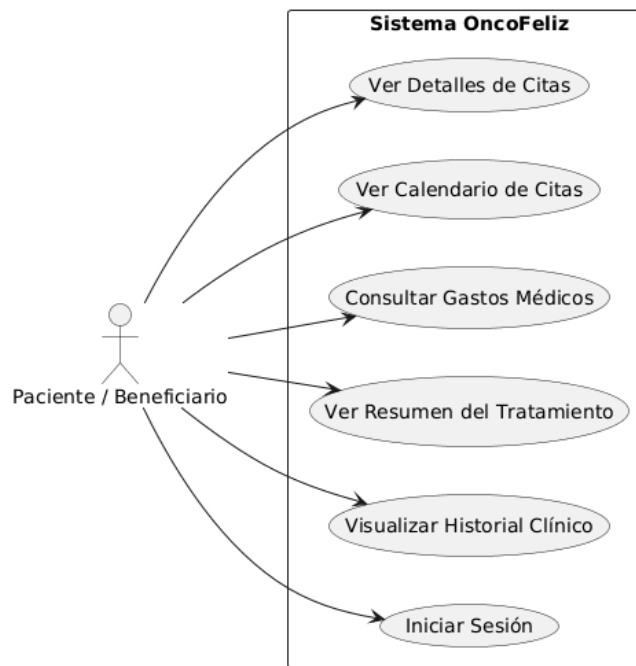


Fuente: Elaboración propia (2025).

- Paciente/beneficiario: Usa la aplicación móvil desarrollada para Android, debe iniciar sesión para acceder a la vista móvil, donde puede visualizar el historial

clínico, ver el resumen del tratamiento, consultar gastos médicos, ver el calendario y detalles de citas.

Figura 16. Diagrama de casos de uso beneficiario



Fuente: Elaboración propia (2025).

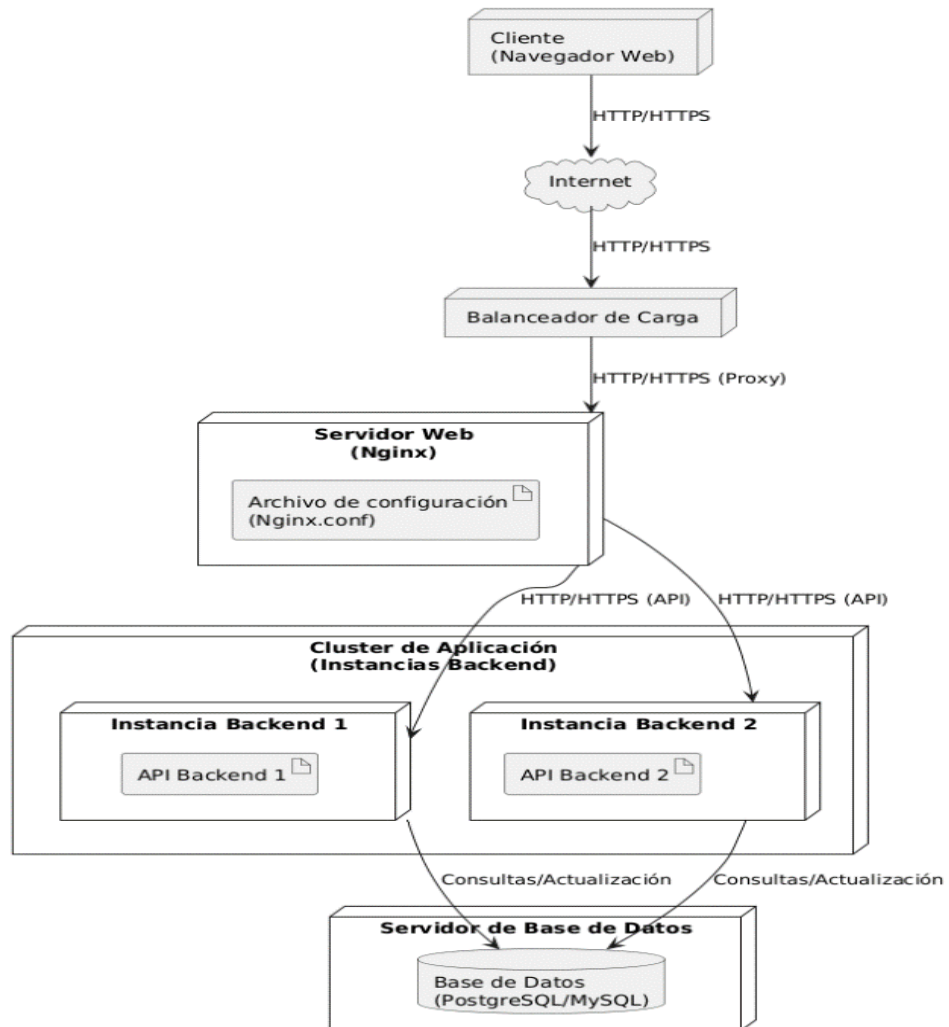
2.3.2.2. Diagrama de despliegue

Un diagrama de despliegue es un tipo de diagrama que se utiliza para representar el despliegue físico de los componentes de un sistema. Muestra cómo se asignan los artefactos de software a los componentes de hardware, conocidos como nodos, e ilustra las configuraciones físicas de software y hardware que son críticas para la ejecución, operación y escalabilidad de un sistema.

Elementos principales:

- **Cliente:** El usuario accede desde un navegador vía Internet.
- **Servidor web:** Actúa como proxy inverso, recibiendo peticiones del balanceador y enviándolas al backend.
- **Backend (cluster):** Contiene múltiples instancias de la API que gestionan la lógica del sistema.
- **Base de datos:** Almacena los datos persistentes, como usuarios, campañas, donaciones, etc.

Figura 17. Diagrama de despliegue



Fuente: Elaboración propia (2025).

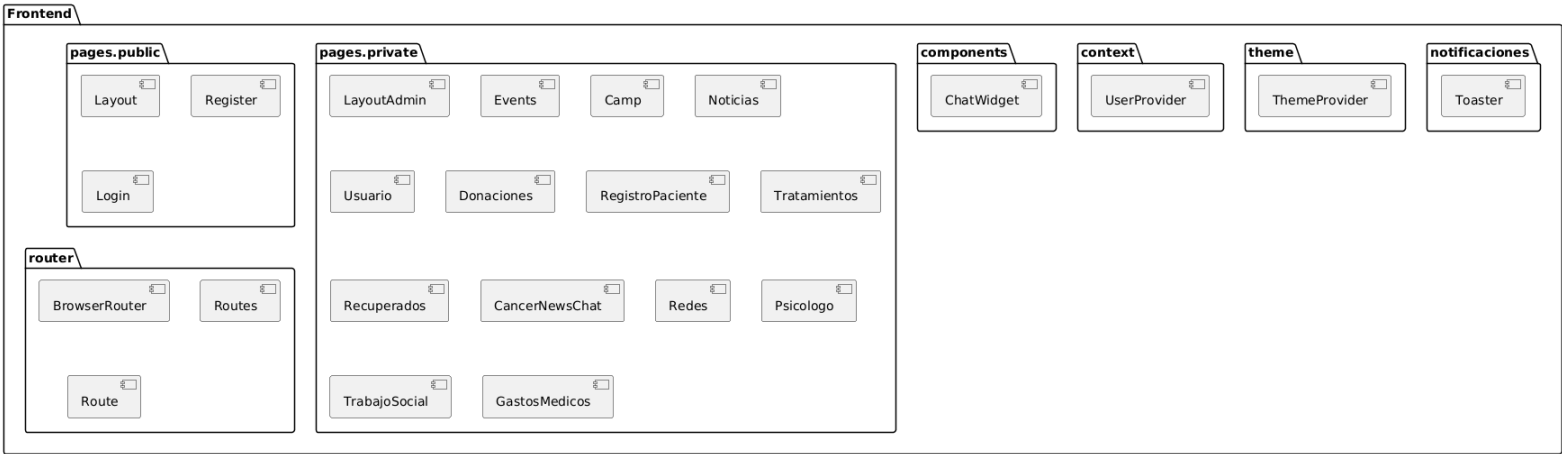
2.3.2.3. Diagrama de paquetes

Figura 18. Diagrama de paquetes backend



Fuente: Elaboración propia (2025).

Figura 19. Diagrama de paquetes frontend

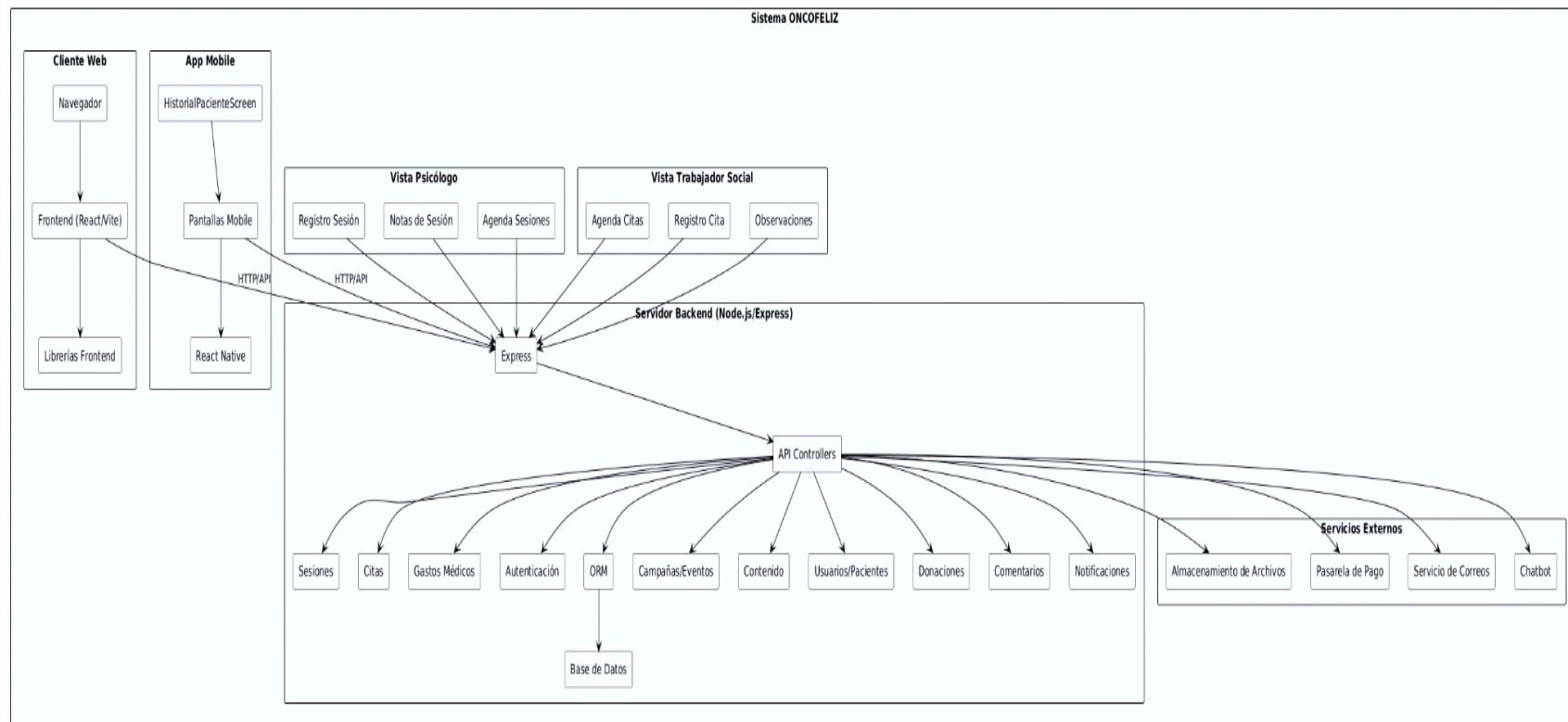


Fuente: Elaboración propia (2025).

2.2.2.4. Diagrama de componentes

El diagrama de componentes en UML es una representación gráfica que muestra la organización y dependencia de los distintos módulos o componentes de software dentro de un sistema, facilitando la visualización de su estructura y relaciones.

Figura 20. Diagrama de componentes



Fuente: Elaboración propia (2025).

Elementos principales:

- Cliente web: El navegador carga el frontend desarrollado en React/Vite, que emplea librerías externas para la interfaz y se comunica con el backend a través de HTTP/API.
- App mobile: Construida con React Native, muestra pantallas móviles (incluyendo el historial de pacientes) que consumen la API del backend.
- Vista psicólogo: Permite gestionar agenda de sesiones, registrar información y añadir notas, interactuando directamente con el servidor backend.
- Vista trabajador social: Ofrece funciones para administrar agenda de citas, registrar atenciones y agregar observaciones, conectándose igualmente al backend.
- Servidor backend: Implementado con Node.js y Express, recibe las peticiones y las dirige a controladores API, que gestionan módulos como autenticación, campañas/eventos, contenido, usuarios/pacientes, donaciones, comentarios, notificaciones, sesiones, citas y gastos médicos.
- Base de datos: El ORM conecta los módulos del backend con la base de datos relacional para persistencia de datos.
- Servicios externos: Algunos módulos interactúan con almacenamiento de archivos, pasarela de pago, servicio de correos y un chatbot para atención automatizada.
- Conexiones: Las flechas indican flujos de datos clave, principalmente vía HTTP/API, y se contempla una conexión directa desde el frontend al chatbot como widget opcional.

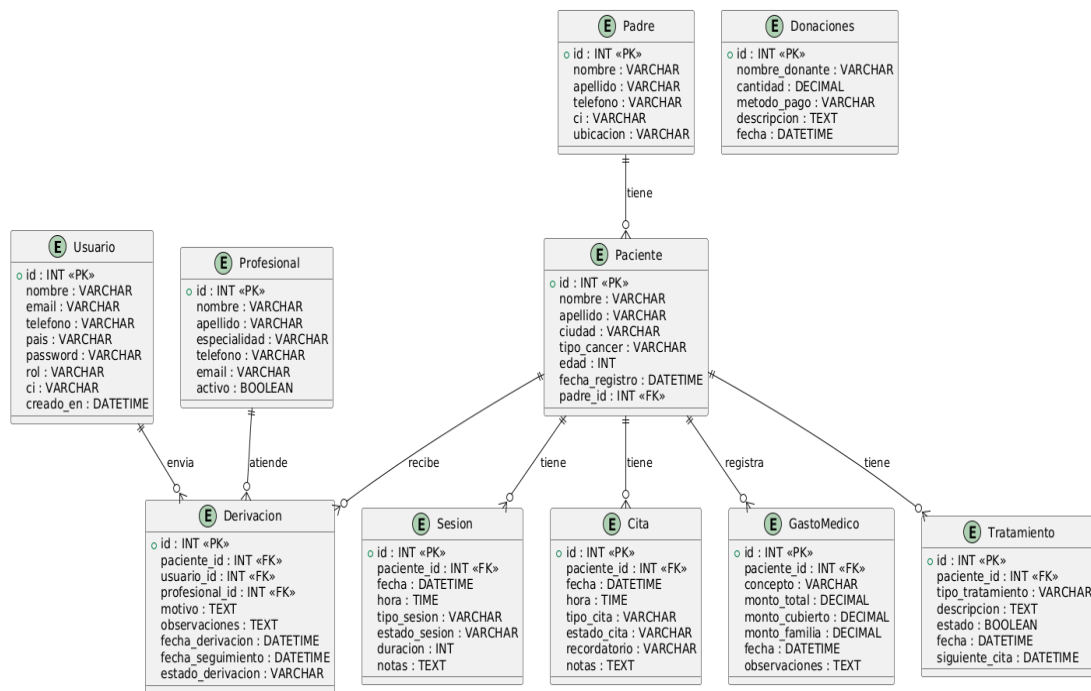
2.3.2.5. Diagrama de clases

El diagrama de clases representa la estructura que tiene la base de datos, muestra las principales entidades, cada tabla incluye sus atributos principales, claves primarias, foráneas y las relaciones entre ellas.

Elementos principales:

- Usuario: Representa a quienes interactúan con el sistema y pueden enviar derivaciones.
- Paciente: Contiene información personal y clínica, vinculada a tratamientos, citas y sesiones.
- Padre: Asociado a uno o varios pacientes como responsable.
- Tratamiento, Sesión, Cita, Derivación: Registros que gestionan la atención y seguimiento de los pacientes.

Figura 21. Diagrama de clases



Fuente: Elaboración propia (2025).

2.3.2.4. Diseño de la base de datos

El diseño lógico de la base de datos es un modelo relacional implementado en PostgreSQL, gestionado mediante Prisma y expuesto a través de Express.js. Su estructura ordena Usuarios (donantes y voluntarios), Padres, Pacientes, Tratamientos, Campañas, Blogs, Eventos y Donaciones.

A continuación, se presenta una descripción de las entidades y sus asociaciones principales:

- Usuario - UsuarioCampaña: Relación uno a muchos, un usuario puede estar asociado a múltiples campañas mediante la tabla intermedia UsuarioCampaña. Cada UsuarioCampaña contiene referencias a un usuario y a una campaña específica.
- Campaña - UsuarioCampaña: Relación uno a muchos, una campaña puede tener múltiples usuarios asociados a través de la misma tabla intermedia.
- Categoría - Blog: Relación uno a muchos, una categoría puede contener múltiples publicaciones de blog.
- Comentario - Respuesta: Relación uno a muchos, cada comentario puede recibir múltiples respuestas.

Figura 22. Diseño Entidad Relación



Fuente: Elaboración propia (2025).

2.4. Desarrollo

2.4.1. Tecnologías de desarrollo

Para el desarrollo del proyecto se eligieron diferentes tecnologías que se adaptan a las necesidades del sistema. La siguiente tabla presenta una comparación de las herramientas utilizadas, las cuales fueron clasificadas por su función principal. Se describen características más importantes y las ventajas que aportan al desarrollo, esta selección busca asegurar un funcionamiento adecuado, una interfaz amigable y un sistema fácil de mantener y ampliar en el futuro.

El stack tecnológico utilizado está conformado por herramientas modernas y de código abierto, las cuales fueron seleccionadas por rendimiento, facilidad de integración y compatibilidad.

A continuación, se describen los principales componentes en el desarrollo:

Tabla 49. Stack tecnológico del sistema

Componente	Tecnología utilizada	Descripción
Backend	Node.js con Express	Gestiona la lógica del sistema, las rutas y la conexión con la base de datos. Permite manejar múltiples solicitudes de manera eficiente.
Frontend	React, HTML5, CSS3 y JavaScript	Desarrolla la interfaz de usuario tanto para la página web como para la aplicación móvil, brindando una experiencia interactiva y adaptable.
Base de datos	PostgreSQL	Almacena de forma estructurada los datos de beneficiarios, donaciones y usuarios, garantizando seguridad e integridad de la información.
Entorno de pruebas	Postman	Permite validar las peticiones y respuestas del backend, comprobando el correcto funcionamiento de los servicios API.
Control de versiones	GitHub	Facilita la gestión del código fuente, el seguimiento de cambios y la colaboración en el desarrollo mediante repositorios remotos.

Fuente: Elaboración propia (2025).

2.4.2. Implementación del backend

El desarrollo del backend se llevó a cabo empleando Node.js junto con el framework Express, lo que facilita la administración del enrutamiento y la lógica del servidor de manera eficaz. Para interactuar con la base de datos, se utilizó Prisma ORM, una herramienta que simplifica el acceso y la gestión de datos a través de consultas optimizadas. La base de datos se implementó en PostgreSQL debido a su capacidad de escalar y cumplir con normas SQL.

- **Node.js**

Se utiliza para la navegación entre páginas y rutas dentro de la aplicación, permitiendo una experiencia de usuario fluida y sin recargas, lo que posibilitó la creación de aplicaciones del lado del servidor capaces de gestionar varias peticiones al mismo tiempo, lo cual es positivo para la escalabilidad y el rendimiento. Es especialmente apropiada para sistemas que requieren manejar múltiples conexiones simultáneamente.

- **Express**

Framework para crear la API REST y gestionar rutas, controladores y middlewares, se usó para estructurar el funcionamiento del servidor. Esta herramienta colabora en la determinación de las rutas a las que tienen acceso los usuarios, el manejo de la información entrante y saliente, así como la

implementación de funciones particulares para estructurar el flujo de la aplicación. Se consiguió establecer una estructura clara y ordenada gracias a Express.

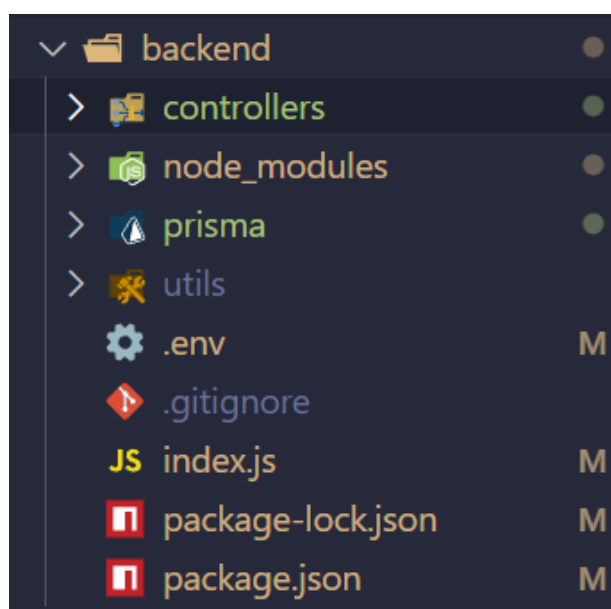
- **Prisma ORM**

Para gestionar el acceso a datos, se eligió Prisma ORM, un mapeador objeto-relacional que ofrece una interfaz sencilla para trabajar con la base de datos. Prisma permite definir esquemas de datos, llevar a cabo migraciones automáticas y hacer consultas, lo que minimiza errores comunes y aumenta la eficacia del desarrollo. También favorece la colaboración con TypeScript.

- **PostgreSQL**

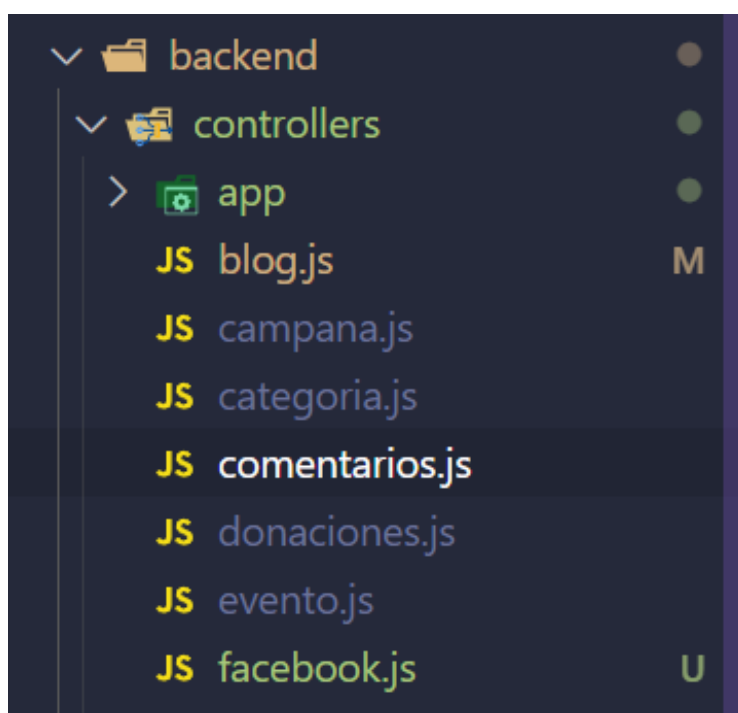
Se optó por PostgreSQL como base de datos, un sistema que permite almacenar y organizar la información de forma segura y ordenada. Se eligió este sistema porque tiene la habilidad de gestionar grandes volúmenes de datos sin problemas, proporciona características útiles para proteger la información y es adecuado para sistemas que crecen o que tienen una gran cantidad de usuarios. Esta elección garantiza que los datos estén adecuadamente organizados, sean fáciles de consultar y permanezcan seguros.

Figura 23. Estructura backend



Fuente. Elaboración propia (2025).

Figura 24. Estructura backend



Fuente. Elaboración propia (2025).

2.4.3. Implementación del frontend

La plataforma digital está compuesta por una página web institucional y una ampliación móvil.

Para el desarrollo del frontend de la página web se utilizó la biblioteca React basada e JavaScript, permite desarrollar interfaces de usuarios dinámicas, reutilizables y escalable.

Para la ampliación móvil se utilizó React Native el cual comparte parte del código y componentes de React lo cual asegura una experiencia visual y funcional.

Además, se emplearon HTML5 y CSS3 para definir la estructura, estilo y presentación visual del sistema, garantizando una interfaz accesible y agradable para los usuarios.

A continuación, se detallan las principales tecnologías y librerías empleadas:

- **React**

Biblioteca principal utilizada para el desarrollo de la versión web, para la construcción de componentes y la gestión del estado de la interfaz.

Esta biblioteca permite la creación de páginas web dinámicas mediante la división de la pantalla en componentes, que son secciones pequeñas que pueden ser reutilizadas en distintas partes del sistema.

Gracias a React la página web puede actualizar contenido en tiempo real sin necesidad recargar la página de manera constante lo cual ofrece una experiencia fluida al usuario.

- **React Native**

Para la versión móvil se implementó React Native, un entorno que permite reutilizar la mayoría de los componentes desarrollados en React para crear aplicaciones nativas.

Permitiendo que los beneficiarios puedan acceder desde sus dispositivos móviles a la información personalizada, historial, calendario de citas, etc.

La integración con la base de datos del sistema permite que la información se mantenga sincronizada entre la aplicación móvil y la plataforma web.

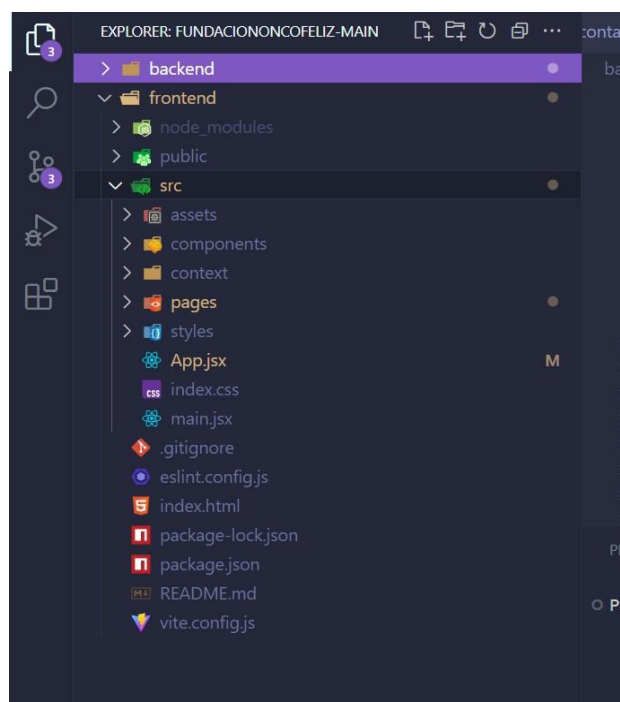
- **HTML5**

HTML5 (HyperText Markup Language) se utilizó para estructurar el contenido de las páginas, lo que significa definir los elementos que son visibles, como los textos, botones, formularios y menús. Por eso HTML es la base en la que se construye toda la interfaz visual.

- **CSS3**

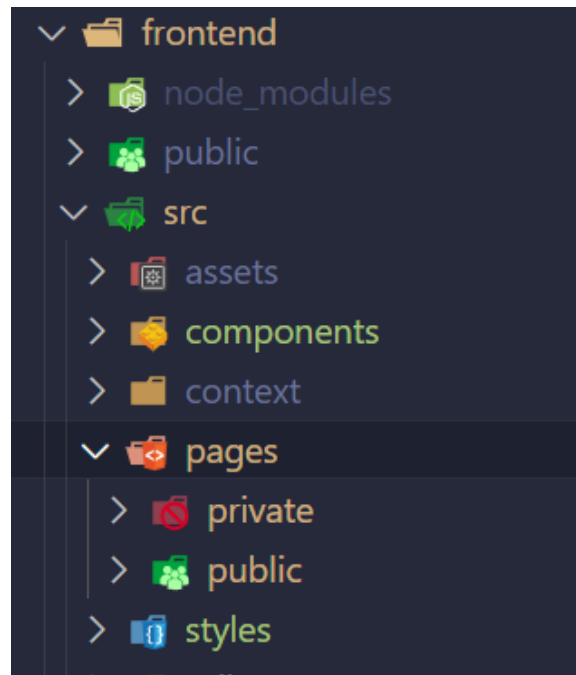
Se utilizó CSS3 (Cascading Style Sheets) para aplicar estilos a los elementos creados con HTML. Esta tecnología permitió definir los colores, tamaños, fuentes, márgenes y otros aspectos visuales del sistema, logrando una interfaz intuitiva y fácil de usar.

Figura 25. Estructura frontend



Fuente. Elaboración propia (2025).

Figura 26. Estructura frontend



Fuente. Elaboración propia (2025).

2.4.4. Integración de herramientas con inteligencia artificial (IA)

Dentro del desarrollo de la plataforma digital se implementaron herramientas con (IA) lo cual complementa funcionalidades del sistema lo cual mejora la experiencia de los usuarios.

La integración de estas herramientas permite automatizar tareas, ofrecer asistencia personalizada y generar contenido informativo de manera dinámica.

2.4.4.1. Generación de contenido con GPT

Se integró GPT para la creación automática de contenido institucional: publicaciones para campañas, mensajes de concienciación y materiales para redes sociales. El asistente genera texto estructurado (JSON) que el frontend interpreta para crear eventos, campañas o posteos, reduciendo el trabajo manual y acelerando la difusión.

- Modelo utilizado

Se implementó el modelo de "Openai/chatgpt-4o-latest", este modelo pertenece a la familia de GPT-4o. Este modelo fue diseñado para entender instrucciones y generar respuestas en forma de conversación, esto lo hace útil para crear mensajes, responder preguntas y redactar contenido de manera automática, tiene la capacidad de generar respuestas en formatos específicos como JSON lo cual permite conectarlo con otros módulos del sistema.

El objetivo es lograr respuestas ordenadas y con una estructura clara en JSON, para que el sistema pueda leerlas fácilmente y crear los recursos de forma automática.

El código recibe la respuesta de la IA y la convierte en formato JSON guardándolo en la variable, si los datos son los esperados el sistema muestra un mensaje de confirmación, según el tipo de contenido generado.

Figura 27. Petición al modelo (prompt y parámetros)

```
{
  "model": "openai/chatgpt-4o-latest",
  "max_tokens": 500,
  "temperature": 0.0,
  "messages": [
    { "role": "system", "content": "Responde solo con un JSON válido para eventos cuando el usuario pide un evento" },
    { "role": "user", "content": "crea un evento de recaudación para el 10 de junio" }
  ]
}
```

Fuente. Elaboración propia (2025).

Figura 28. Procesamiento de la respuesta en el frontend

```
{
  user: question,
  results: [
    {
      type: 'text',
      content: `
        ? ✅ La publicación ha sido enviada a Facebook exitosamente!\n\n📄 Mensaje: ${String(parsed.message).trim()}
        (parsed.imageUrl ? `\n🖼️ Imagen: ${String(parsed.imageUrl).trim()}` : '') +
        (parsed.link ? `\n🔗 Enlace: ${String(parsed.link).trim()}` : '')
      `,
      type: 'text',
      content: `! Respuesta del asistente: ${text.slice(0, 100)}` // mostrar parte del texto no procesado
    }
  ],
}
```

Fuente. Elaboración propia (2025).

Figura 29. Acción personalizada para interactuar con GPT

```
class ActionCallGPT(Action):
  def name(self) -> Text:
    return "action_call_gpt"

  def run(self, dispatcher: CollectingDispatcher,
          tracker: Tracker,
          domain: Dict[Text, Any]) -> List[Dict[Text, Any]]:

    user_message = tracker.latest_message.get("text", "")
    # system prompt to force JSON when appropriate
    system_prompt = (
      "Eres un asistente que responde con JSON cuando el usuario pide 'crea un evento', "
      "o devuelve texto si la petición es una consulta. Si se pide 'publica en facebook' "
      "responde sólo con un JSON con keys: type, title, message, imageUrl, link."
    )
```

Fuente. Elaboración propia (2025).

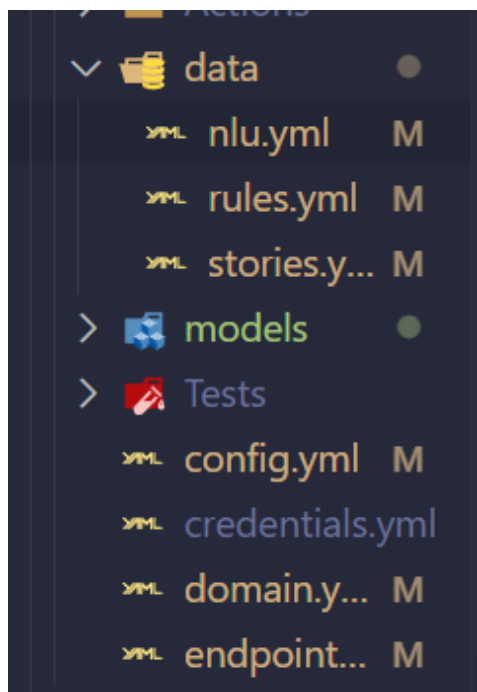
2.4.4.2. Chatbot con Rasa

El chatbot se implementó como una herramienta de apoyo para mejorar la comunicación con los usuarios de la página web informativa de la fundación Oncofeliz, su tarea es brindar información básica sobre los servicios de la fundación, campañas y los pasos para realizar donaciones.

Rasa se configuró como un servicio que se comunica con la página web mediante una API REST, fue entrenado con ejemplos reales de preguntas y respuestas para reconocer intenciones y responder.

Rasa utiliza archivos principales para entrenar y guiar el funcionamiento del chatbot. El archivo `domain.yml` define las intenciones, entidades, variables de contexto, respuestas predefinidas y acciones personalizadas que el bot puede ejecutar. El archivo `data/nlu.yml` proporciona ejemplos de frases de los usuarios, permitiendo al modelo reconocer correctamente las intenciones y extraer información relevante. Finalmente, `data/stories.yml` describe escenarios de conversación, mostrando cómo el bot debe responder según el contexto, lo que facilita la selección de acciones o respuestas adecuadas durante la interacción.

Figura 30. Organización de archivos principales del modelo de IA Rasa



Fuente. Elaboración propia (2025).

Figura 31. Configuración de respuestas automáticas en el archivo domain.yml

```
Rasa-chatbot > Rasa > domain.yml
42  responses:
43
48    utter_goodbye:
49      - text: "Ha sido un gusto poder ayudarte. Recuerda que siempre estamos aq
50      - text: "Espero haber podido ayudarte hoy. No olvides que el equipo de On
51      - text: "Me alegra haber podido acompañarte. Recuerda: no están solos en
52
53    utter_iamabot:
54      - text: "Soy el asistente virtual de OncoFeliz, creado para acompañar a f
55
56    utter_treatments_info:
57      - text: "Los tratamientos principales para el cáncer infantil incluyen:\n
```

Fuente. Elaboración propia (2025).

Figura 32. Entrenamiento de intenciones de saludo en el archivo nlu.yml

```
Rasa-chatbot > Rasa > data > nlu.yml
4  - intent: greet
5    examples: |
10   - hey
11   - saludos
12   - hi
13   - hello
14   - qué tal
15   - hola, necesito ayuda
16   - buenos días, soy nueva aquí
17   - hola, me pueden ayudar
18   - buenas, tengo una consulta
19   - holaa
20   - hoooola
21   - buenas
22   - buen día
```

Fuente. Elaboración propia (2025).

2.4.5. Interfaces

El sistema tiene interfaces gráficas desarrolladas para simplificar la interacción al usuario con la plataforma digital. Se sugiere una interfaz intuitiva, accesible y funcional que se ajuste a lo que requieren los usuarios finales y el personal administrativo, asegurando así una experiencia comprensible, eficaz y visualmente atractiva.

El objetivo principal es garantizar una experiencia clara, eficiente y visualmente agradable tanto en la página web como en la aplicación móvil.

2.4.5.2. Vista administrativa web

La interfaz administrativa está destinada al personal encargado de la gestión interna de la plataforma, a través de esta interfaz el administrador podrá controlar los distintos módulos del sistema mediante un entorno web seguro, las principales funcionalidades que ofrece son:

- Inicio de sesión.

- Panel principal (dashboard).
- Gestión de usuarios.
- Control de contenido.
- Monitoreo de tratamientos.
- Generación de reportes.
- Crear contenido con el asistente como eventos, campañas y publicaciones en redes sociales con IA.

Figura 33. Inicio de sesión web administrativa

Iniciar Sesión
Event Manager

Correo electrónico
admin

Contraseña

¿Olvidaste tu contraseña?

Iniciar sesión

¿No tienes una cuenta? [Crear cuenta nueva](#)

Inicio de sesión: Usuario. - Contraseña.

Botón inicio de sesión.

Fuente. Elaboración propia (2025).

Figura 34. Panel administrativo de la plataforma web con opciones de gestión de usuarios, eventos y contenidos

2. Panel Principal (Dashboard de administración)

Navegar entre secciones con el menú lateral.

Buscar registros con el buscador.

Campo de búsqueda para filtrar datos.

Exportar datos. Opciones para generar archivos Excel o PDF.

Cerrar sesión.

Dashboard

Total Donado en el Mes
200.00 Bs

ID	NOMBRE DEL DONANTE	CANTIDAD	MÉTODO DE PAGO	FECHA	EDITAR	ELIMINAR
1	Ana López	200.00	Efectivo	02/10/2024		

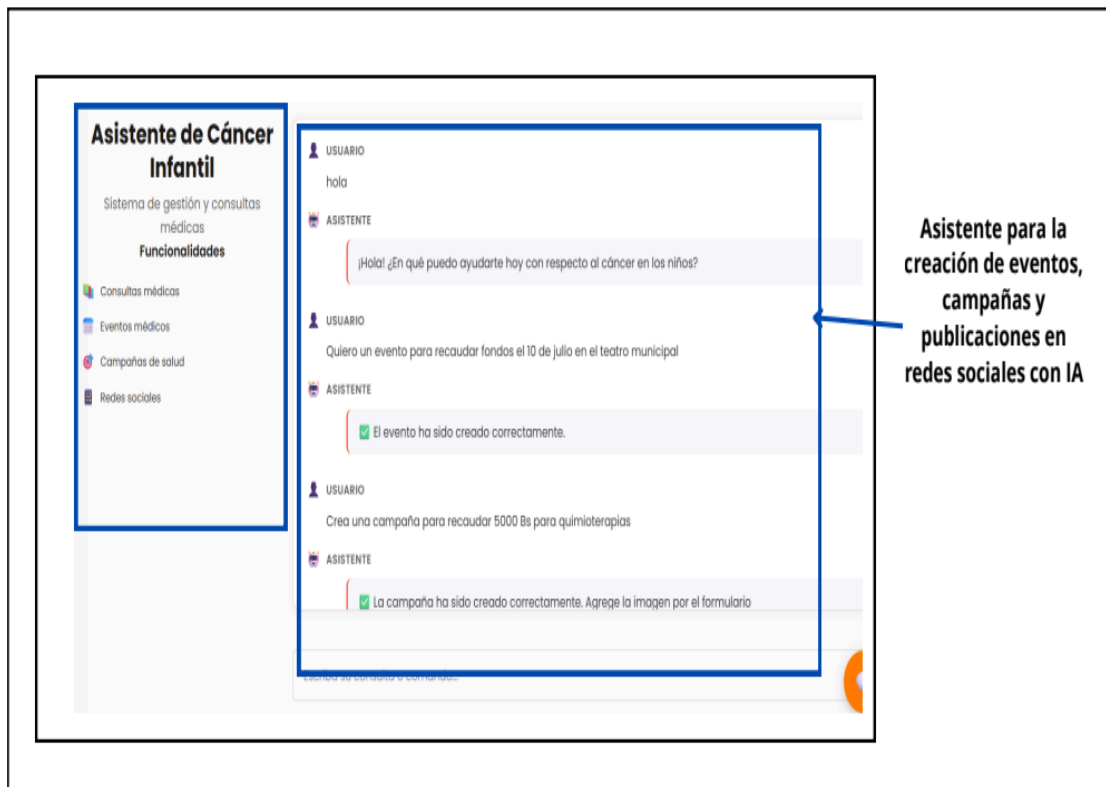
Tabla de datos de donantes

Acciones disponibles:

- Editar (icono de lápiz)
- Eliminar (icono de tachito de basura)

Fuente. Elaboración propia (2025).

Figura 35. Interfaz del asistente virtual para apoyar en la creación y personalización de contenidos digitales



Fuente. Elaboración propia (2025).

2.4.5.3. Vista web informativa

La interfaz informativa está dirigida a donantes, y ha sido diseñada bajo principios de usabilidad y simplicidad, asegurando una experiencia amigable desde dispositivos móviles y navegadores web. Las principales secciones son:

- Acceso sin autenticación.
- Perfil informativo.
- Módulo de campañas (para donantes).
- Canales de contacto.
- Consultas o contacto directo con la fundación, sin necesidad de iniciar sesión.
- Donación mediante la plataforma digital.
- Donaciones en línea.
- Chatbot para preguntas frecuentes.

Figura 36. Vista informativa de la plataforma web con acceso a secciones de noticias, actividades y servicios disponibles



Fuente. Elaboración propia (2025).

Figura 37. Acceso público a campañas activas y visualización del perfil informativo de cada actividad



Fuente. Elaboración propia (2025).

Figura 38. Canales de contacto y campañas, consultas y soporte en línea disponibles en la vista informativa de la plataforma

Nuestras Campañas

Equipamiento para Sala de Oncología Pediátrica
Ayúdanos a equipar la nueva sala de oncología pediátrica con equipos modernos que mejorarán diagnósticos y tratamientos.

Recaudado: Bs 8.000,00 Meta: Bs 10.000,00

Donar Ahora

Medicamentos para Tratamientos Oncológicos
Tu donación ayudará a cubrir los costosos medicamentos para el tratamiento de niños de familias con bajos recursos.

Recaudado: Bs 5.000,00 Meta: Bs 1.200,00

Donar Ahora

Apoyo Psicológico para Familias
Financiamos terapias para las familias de niños con cáncer, un aspecto crucial en la lucha contra la enfermedad.

Recaudado: Bs 1.800,00 Meta: Bs 30.000,00

Donar Ahora

Contáctanos

Estamos aquí para ayudarte. Si tienes alguna pregunta o quieres colaborar, no dudes en contactarnos.

+52 (55) 1234 5678
contacto@oncofeliz.org
Ciudad de Bolivia, Cochabamba

Nombre
Tu nombre

Correo Electrónico
tu@email.com

Mensaje
Escribe tu mensaje aquí

Enviar Mensaje

5. Campañas

Contenido:

- Campañas activas de donación, sensibilización o voluntariado.
- Descripción breve y botones de participación.

Objetivo: Motivar a la acción social.

6. Contacto

Contenido:

- Formulario con nombre, correo, mensaje.
- Dirección, teléfono, correo institucional y mapa.

Objetivo: Facilitar la comunicación directa con la fundación.

Fuente. Elaboración propia (2025).

Figura 39. Formulario de donación

Haz tu donación

Tu apoyo transforma vidas. Juntos podemos hacer la diferencia.

Nombre

Tu nombre completo

Selecciona o ingresa una cantidad (Bs)

100 Bs

200 Bs

500 Bs

1000 Bs

Otra cantidad

Método de Pago

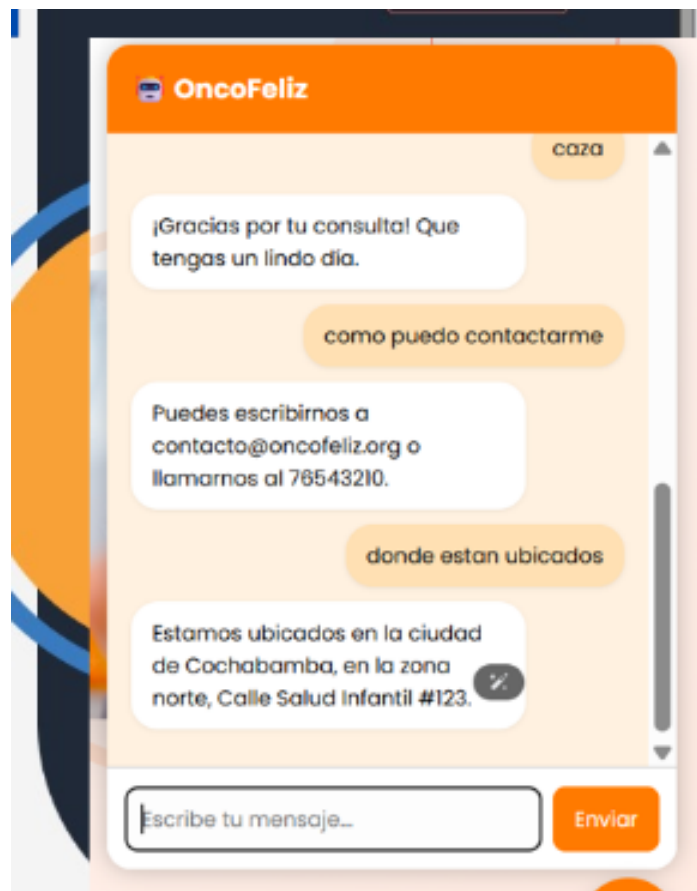
Tarjeta de Crédito/Débito

Pago con QR

Fuente. Elaboración propia (2025).

79

Figura 40. Chatbot web informativa



Fuente. Elaboración propia (2025).

2.4.5.4. Vista aplicación móvil

La interfaz móvil tiene el propósito de ofrecer una experiencia amigable siguiendo principios de usabilidad de esta forma este enfocada en pacientes, beneficiarios para hacer un seguimiento de los mismos y mediante un asistente hacer un acompañamiento:

Dashboard: resumen de citas, tratamientos, notificaciones y accesos rápidos a historial.

- Perfil: ver datos personales.
- Citas y agenda: consultar calendario y ver detalles de citas.
- Historial médico y tratamientos: sesiones, tratamientos y gastos.
- Notificaciones: alertas sobre citas, seguimiento psicológico.
- Asistente: Chat para preguntas sobre el cáncer, cuidados, etc.

Figura 41. Vista general de la aplicación móvil con funciones orientadas a usuarios, incluyendo notificaciones y seguimiento



Fuente. Elaboración propia (2025).

2.5. Pruebas

2.5.1. Pruebas de interacción con la base de datos

Para la gestión de datos en la base de datos (es decir, operaciones de lectura, creación y eliminación de datos), es proyecto hace uso del patron estándar de una API REST en el backend.

Esto se traduce en el uso de los siguientes métodos HTTP para el manejo de datos:

- **GET:** Se utiliza para consultar u obtener datos de la base de datos.
- **POST:** Se utiliza para crear nuevos registros en la base de datos.
- **DELETE:** Se utiliza para eliminar datos de la base de datos.

Figura 42. Obtener eventos registrados

```
app.get("/evento", async (req, res) => {
  try {
    const evento = await prisma.evento.findMany({});
    res.json({
      data: evento,
      mensaje: "eventos obtenidos correctamente",
    });
  } catch (error) {
    res.status(500).json({
      mensaje: "Error al traer evento",
      error: error.mensaje,
    });
  }
});
```

Fuente. Elaboración propia (2025).

Figura 43. Registrar nuevo usuario

```
app.post("/usuario", async (req, res) => {
  try {
    const { nombre, email, telefono, pais, ci, rol, password } = req.body;
    const ROUNDS = 10;
    const salt = bcrypt.genSaltSync(ROUNDS);
    const hashedPassword = await bcrypt.hash(password, salt);
```

Fuente. Elaboración propia (2025).

Figura 44. Eliminar usuario por ID

```
app.delete("/usuario/:id", async (req, res) => {
  try {
    await prisma.usuario.delete({
      where: {
        id: Number(req.params.id),
      },
    });
    res.json({
      mensaje: "Usuario eliminado correctamente",
```

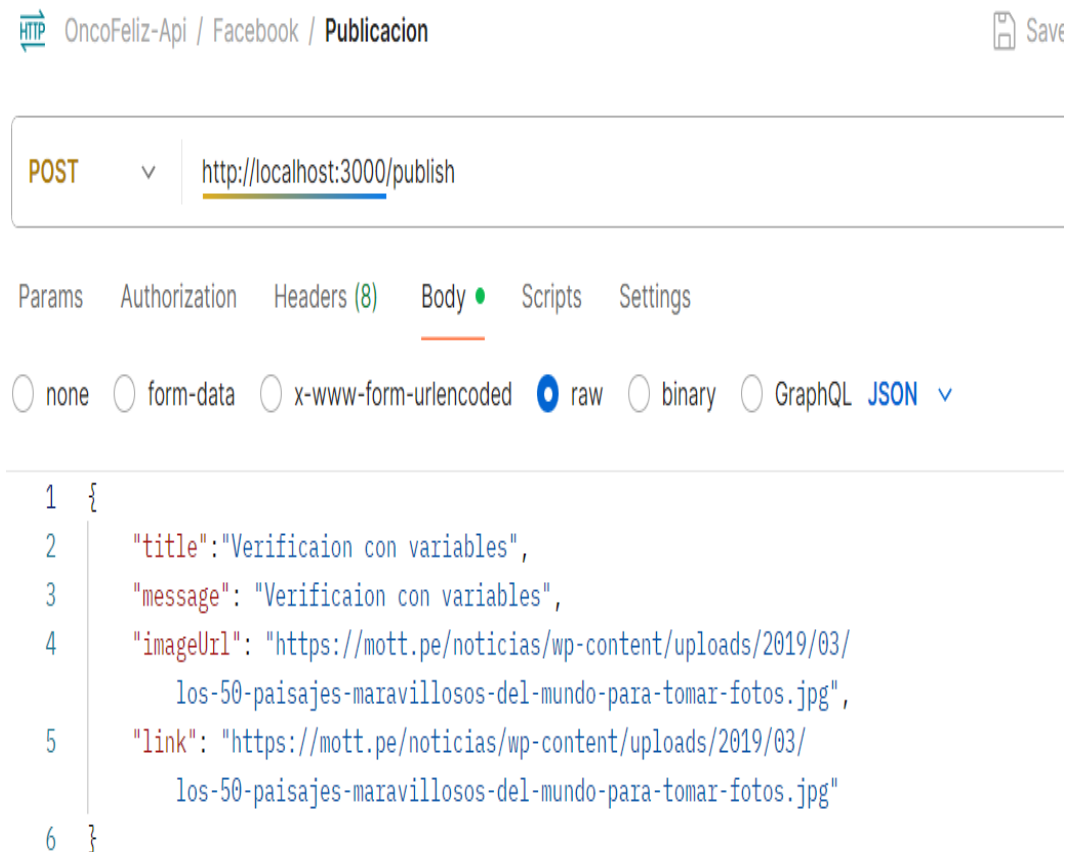
Fuente. Elaboración propia (2025).

2.5.2. Pruebas de integración con APIs externas

El objetivo es verificar la comunicación correcta entre la aplicación y los servicios externos, para esto se realizaron pruebas en Postman donde se realizan peticiones HTTP hacia diferentes APIs.

Se verifico la conexión con Zapier para realizar la publicación a Facebook desde la página web. Se hizo una petición que incluye parámetros JSON, titulo, mensaje, enlace de imagen para la publicación.

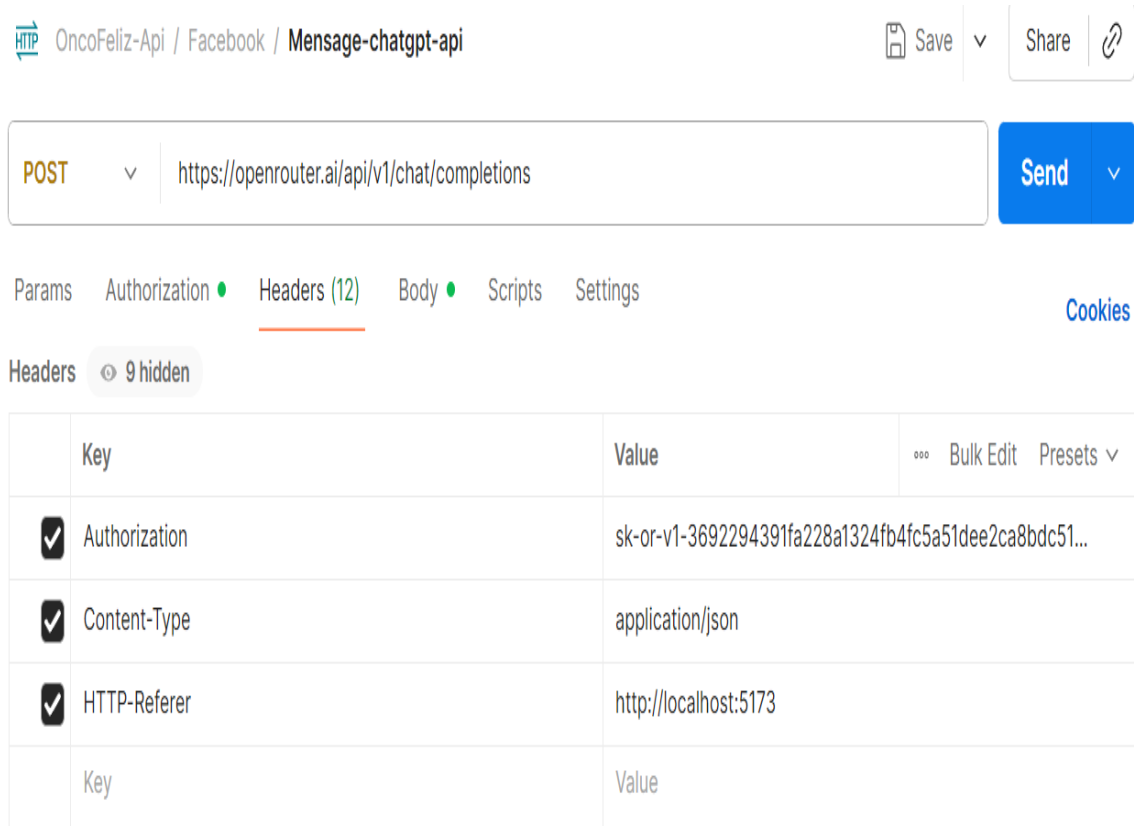
Figura 45. Prueba en postman publicación en Facebook



Fuente. Elaboración propia- Postman.

Configuración con la API de OpenRouter (ChatGPT), método POST enviando datos de autenticación, contenido y referencia. Permite la comunicación con el asistente de inteligencia artificial que se encarga de responder preguntas frecuentes y brindar acompañamiento a los padres.

Figura 46. Prueba Postman Api de OpenRouter



Fuente. Elaboración propia- Postman.

2.5.3. Pruebas de caja negra

Las pruebas de caja negra se aplican para realizar pruebas sin tener conocimiento de los componentes internos del sistema, se realizan para evaluar la funcionalidad, seguridad y rendimiento del sistema

Tabla 50. Pruebas de caja negra

ID Caso de Prueba	Módulo / Funcionalidad	Descripción del Caso	Precondiciones	Datos de Entrada	Pasos de Prueba	Resultado Esperado	Resultado Obtenido
Autenticación y sesión							
CP-001	Autenticación	Inicio de sesión válido	Usuario registrado	Usuario y contraseña correctos	Ingresa usuario y contraseña, hacer clic en 'Ingresar'	Acceso concedido al sistema, redirección a página principal	OK
CP-002	Autenticación	Inicio de sesión inválido	Usuario registrado	Usuario o contraseña incorrectos	Ingresa usuario y contraseña incorrectos	Mensaje de error: 'Credenciales inválidas'	OK
CP-003	Autenticación	Inicio de sesión único	Usuario registrado	Intentar abrir sesión en otro navegador con mismo usuario	Intentar iniciar sesión en segundo navegador	No permitir iniciar sesión, mostrar mensaje de sesión activa	OK
CP-004	Autenticación	Acceso a módulo administrativo sin login	Ninguna	N/A	Intentar acceder al módulo administrativo	Acceso denegado, mostrar página de login	OK

CP-005	Navegación	Visualización de barra de navegación	Usuario autenticado	N/A	Abrir página principal	Barra de navegación visible y accesible a todas las secciones	OK
CP-006	Navegación	Descargar listado de eventos	Usuario autenticado	N/A	Hacer clic en botón 'Exportar a Excel'	Archivo Excel descargado correctamente con datos completos	OK
CP-007	Navegación	Carga de página principal	N/A	Abrir página principal	N/A	Página cargada en menos de 3 segundos, contenido completo y responsive	OK
CP-008	Blog	Visualizar artículos del blog	N/A	N/A	Abrir sección Blog	Artículos listados correctamente con título, contenido y fecha	OK
CP-009	Eventos	Visualizar eventos	N/A	N/A	Abrir sección Eventos	Eventos próximos y pasados listados correctamente	OK
CP-010	Casos recuperados	Visualizar casos recuperados	N/A	N/A	Abrir sección testimonios	Mostrar testimonios/estadísticas correctamente	OK
CP-011	Campañas	Visualizar campañas	N/A	N/A	Abrir sección campañas	Campañas activas y pasadas listadas correctamente	OK

CP-012	Formulario contacto	Enviar formulario de contacto	N/A	Nombre, correo, mensaje	Completar formulario y enviar	Mensaje enviado correctamente, confirmación visual	OK
CP-013	Donaciones	Donar mediante código QR	Usuario registrado	Datos del donante completos	Completar formulario y generar QR	Generación de QR válida y registro de la donación	OK
CP-014	Validación	Campos obligatorios vacíos	N/A	Campos vacíos	Intentar enviar formulario vacío	Mensaje de error: 'Campos obligatorios'	OK
Gestión de eventos/campañas/noticias							
CP-015	Eventos	Agregar evento	Título y descripción válidos	Ingresar datos del evento	Registrar evento	Evento registrado correctamente	OK
CP-016	Eventos	Editar evento	Evento existente	Modificar título/descripción	Editar evento	Evento actualizado correctamente	OK
CP-017	Eventos	Eliminar evento	Evento existente	Seleccionar evento	Eliminar evento	Evento eliminado correctamente, confirmado	OK
CP-018	Eventos	Buscar evento	Evento existente	Palabra clave en título/descripción	Buscar evento	Resultados filtrados correctamente	OK
CP-019	Eventos	Exportar eventos a Excel/PDF	Evento existente	Botón exportar	Exportar archivo	Archivo generado correctamente, con datos exactos	OK

CP-020	Noticias	Publicar noticia	N/A	Título, contenido e imagen	Registrar noticia	Noticia publicada correctamente	OK
CP-021	Noticias	Editar noticia	Noticia existente	Modificar título/contenido	Editar noticia	Noticia actualizada correctamente	OK
CP-022	Noticias	Eliminar noticia	Noticia existente	Seleccionar noticia	Eliminar noticia	Noticia eliminada correctamente	OK
Gestión de usuarios y seguridad							
CP-023	Usuarios	Registrar usuario	N/A	Nombre, rol y correo válidos	Registrar usuario	Usuario registrado correctamente	OK
CP-024	Usuarios	Editar usuario	Usuario existente	Cambiar datos de usuario	Editar usuario	Datos actualizados correctamente	OK
CP-025	Usuarios	Eliminar usuario	Usuario existente	Seleccionar usuario	Eliminar usuario	Usuario eliminado correctamente	OK
CP-026	Usuarios	Cambiar contraseña	Usuario autenticado	Contraseña actual y nueva	Cambiar contraseña	Contraseña cambiada correctamente, con validación de cifrado	OK
Donaciones							
CP-027	Donaciones	Registrar donación	N/A	Datos válidos del donante	Registrar donación	Donación registrada y visualizada en listado	OK
CP-028	Donaciones	Habilitar QR	Donación registrada	Completar formulario de donación	Generar QR	Código QR generado correctamente	OK

CP-029	Donaciones	Ver historial	N/A	Acceder historial	Visualizar historial	Listado completo de donaciones	OK
CP-030	Donaciones	Exportar donaciones	N/A	Botón exportar	Exportar archivo	Archivo Excel/PDF descargado con datos correctos	OK
CP-031	Donaciones	Mostrar total donaciones por mes	N/A	Consultar reporte mensual	Mostrar total	Total calculado correctamente	OK
Psicólogo y trabajador social							
CP-032	Psicólogo	Registrar cita psicólogo	Paciente existente	Fecha, hora y paciente	Registrar cita	Cita registrada correctamente	OK
CP-033	Psicólogo	Registrar historial de tratamiento	Paciente existente	Información de evolución	Registrar historial	Historial actualizado correctamente	OK
CP-034	Trabajador social	Registrar expediente social	Beneficiario existente	Datos de beneficiario	Registrar expediente	Registro completado correctamente	OK
CP-035	Trabajador social	Programar visita seguimiento	Beneficiario existente	Fecha, hora, motivo	Programar visita	Registro guardado correctamente	OK
CP-036	Trabajador social	Registrar evaluación de vulnerabilidad	Beneficiario existente	Puntaje 0-100	Registrar evaluación	Evaluación registrada y calculada correctamente	OK

CP-037	Trabajador social	Registrar derivación	Beneficiario existente	Motivo, fecha, profesional	Registrar derivación	Derivación registrada correctamente	OK
CP-038	Trabajador social	Registrar donación a paciente	Beneficiario existente	Concepto y montos	Registrar donación	Registro realizado correctamente	OK
CP-039	Psicólogo/Trabajador social (móvil)	Notificaciones de citas	Usuario registrado	Configuración de recordatorio	Enviar notificación	Notificación enviada 24h antes	OK

Fuente: Elaboración propia (2025).

2.6.4. Pruebas de usabilidad con usuarios

Las pruebas de usabilidad se realizan con el fin de evaluar la facilidad de uso, si la interfaz es intuitiva y el grado de satisfacción de los usuarios al usar la plataforma web y móvil que fue desarrollada para la fundación Oncofeliz.

Para las pruebas se contó con la participación de 10 usuarios (padres, voluntarios, donantes, etc.), estos representan el público objetivo de la plataforma digital, se les asignó tareas específicas.

Cada participante realizó las siguientes acciones:

- Visualizar la página web informativa:
Conocer el trabajo de la fundación y sus objetivos.
- Realizar una donación en línea:
Usar el formulario disponible.
- Publicar en Facebook:
Hacer uso del asistente virtual para realizar una publicación.
- Consultar al chatbot:
La página web informativa cuenta con un chatbot para realizar preguntas sobre temas relacionados con donaciones y apoyo.
- Ver la agenda de citas:
Ver citas agendadas.
- Agendar una cita.
- Programar un recordatorio de asistencia.

Al completar las tareas se aplicó una encuesta corta con las siguientes preguntas:

Tabla 51. Encuestas de usabilidad

Pregunta	Tipo de respuesta
¿Fue fácil navegar por la plataforma?	Sí / No
¿Pudo completar todas las tareas asignadas?	Sí / No
Califique su satisfacción general con la plataforma.	Escala de 1 a 5
¿Hubo algo que le resultó confuso o difícil de usar?	Abierta

Fuente. Elaboración propia (2025).

Resultados obtenidos:

- ¿Fue fácil navegar por la plataforma?
8/10 respondieron *Sí*
- ¿Pudo completar la tarea que necesitaba?

9/10 respondieron *Sí*

- Calificación promedio de satisfacción
4.5 / 5

Al realizar las pruebas de usabilidad también se obtuvieron los indicadores de usabilidad.

Tabla 52. Indicadores de usabilidad

Indicador	Descripción	Resultado
Rendimiento	Tiempo promedio de respuesta de la plataforma	< 2 segundos
Efectividad	Porcentaje de tareas completadas correctamente	90%
Satisfacción	Promedio de la encuesta aplicada a los usuarios	4.5 / 5

Fuente. Elaboración propia (2025).

En base a los resultados obtenidos se logró reflejar que la plataforma tiene una interfaz fácil de usar es decir intuitiva, cuenta con un rendimiento eficaz con un nivel alto de aceptación por parte de los usuarios, gran parte de los usuarios lograron navegar con facilidad, completaron las tareas asignadas y mostraron un grado de satisfacción elevada, esto logro validar la usabilidad y funcionalidad del sistema web y móvil desarrollada para la Fundación OncoFeliz.

CAPÍTULO III

COSTOS

En este capítulo se realiza la estimación aproximada de los costos asociados al desarrollo del presente proyecto, utilizando como metodología el modelo COCOMO II (Constructive Cost Model II). Esta metodología es ampliamente reconocida en el ámbito de la ingeniería de software para prever el esfuerzo, tiempo y recursos necesarios a lo largo del ciclo de vida de desarrollo de un sistema (Boehm, 2000).

3.1. Recolección de datos

La recolección de datos es un paso esencial en todo proyecto porque posibilita la obtención de la información requerida para el desarrollo, análisis y diseño del sistema. En esta etapa, se utilizan diferentes metodologías e instrumentos para recolectar información precisa y relevante que muestre las particularidades y requerimientos del entorno a intervenir las decisiones tomadas se basen en información confiable y actualizada depende de un proceso apropiado de recolección.

3.1.1. Tamaño del proyecto (KLOC)

Tabla 53. KLOCK backend

Componente	Lenguaje	Archivos	Líneas de código	Comentarios	Total	KLOC
Controladores	Node.js / Express	15	1,744	39	1,954	1,74
Base de datos (Prisma)	Prisma / SQL	3	654	49	782	0,65
Configuración	JSON	2	1,74	0	1,742	1,74
Utilidades	Node.js	1	22	0	27	0,02
Archivos raíz	Mixto (Node/JSON)	3	1,777	9	1,798	1,78
Total	—	24	4,197	97	4,561	4,56

Fuente: Elaboración propia (2025).

Tabla 54. KLOCK frontend

Componente	Lenguaje	Archivos	Líneas de código	Comentarios	Total	KLOC
Componentes UI	JavaScript/ReactNative	43	9,719	34	10,721	9.72
Configuración / Datos	JSON	2	5,547	0	5,549	5.55
Scripts auxiliares	React.js	3	116	1	135	0.12
Assets / SVG	XML	3	42	13	68	0.04
Estilos	CSS	2	35	5	45	0.04
HTML base	HTML	1	27	0	29	0.03
Documentación	Markdown	1	5	0	9	0.01
Total	—	55	15,491	53	16,556	15.49

Fuente: Elaboración propia (2025).

Resultado final:

La cantidad total de líneas de código del proyecto es:

KLOC = 19.69

También se puede expresar como:

KSLOC = 19.69 mil líneas de código

3.1.2. Modo de desarrollo del proyecto

Según la clasificación de COCOMO II, el modo semidesarrollado es el que mejor describe este proyecto. Este modo se aplica a sistemas con complejidad moderada, equipos de tamaño medio y requisitos que pueden variar o no estar completamente definidos. Es adecuado para aplicaciones web que utilizan tecnologías modernas y presentan un nivel intermedio de integración y funcionalidad, tal como el desarrollado para la Fundación OncoFeliz.

3.1.3. Factores de escalabilidad

Corresponden a posibles desafíos o condiciones que pueden influir en la complejidad del desarrollo del proyecto. Estos factores se evalúan mediante una escala del 1 al 5, donde cada valor representa el grado de impacto o dificultad que puede generar en el proceso de implementación.

Tabla 55. Factores de escalabilidad

Factor	Descripción	Pregunta orientadora	Valor
Experiencia del equipo	Conocimientos previos y trabajos similares realizados	¿El equipo cuenta con antecedentes en investigaciones de características similares?	4
Adaptabilidad temática	Grado de apertura para modificar objetivos o líneas de trabajo	¿Existe posibilidad de ajustar el enfoque o las metas si las circunstancias lo requieren?	3
Trabajo en equipo	Nivel de comunicación, coordinación y complementariedad entre los integrantes	¿El grupo se organiza de forma eficiente y sus habilidades se complementan bien?	2
Recursos disponibles	Acceso a herramientas, bibliografía, tecnología o medios técnicos necesarios	¿Se cuenta con los insumos básicos para avanzar sin contratiempos?	3
Acceso a información	Facilidad para obtener datos clave, participantes, expertos o contenido documental	¿Hay facilidad para consultar fuentes confiables, datos relevantes y personas clave?	3

Fuente: Elaboración propia (2025).

3.1.4. Factores de costos

Los factores de costos ajustan la estimación del esfuerzo requerido para el desarrollo del proyecto, tomando en cuenta aspectos técnicos, humanos y del entorno. Se valoran entre **0.7** (influencia baja) y **1.3** (influencia alta). A continuación, se presenta un resumen de los factores considerados:

- La precisión y solidez de los datos utilizados.
- El grado de dificultad del enfoque metodológico y del contenido temático.
- La preparación y trayectoria del equipo responsable.
- La disponibilidad de recursos técnicos, operativos y logísticos.

Tabla 56. Factores de costos

Factor	Descripción	Valor
Confiabilidad requerida	Nivel de confianza que se espera en la exactitud y estabilidad del sistema.	1.0
Complejidad del enfoque	Grado de dificultad técnica y metodológica del desarrollo del sistema.	1.1
Experiencia del equipo	Nivel de conocimiento previo del equipo en proyectos similares.	0.9

Recursos disponibles	Acceso a herramientas, bibliografía y condiciones técnicas adecuadas.	1.0
Acceso a la información	Facilidad para obtener datos, expertos o participantes clave.	1.0

Fuente: Elaboración propia (2025).

3.2. Cálculo de escalabilidad ajustada

Para calcular el factor de escalabilidad ajustado se aplica la fórmula:

$$E_{SF} = B + 0.01 \times \sum SF_i$$

- **B:** Valor base típico en COCOMO II, igual a 0.91.
- **SF_i:** Factores de escalabilidad calificados del 1 al 5.
-

$$E_{SF} = 0.91 + 0.01 \times (4 + 3 + 2 + 3 + 3) = 0.91 + 0.15 = 1.06$$

3.3. Cálculo del esfuerzo

El esfuerzo requerido para el desarrollo del proyecto se calcula con la fórmula del modelo COCOMO II:

$$E = A \times (KSLOC)^{E_{SF}} \times \prod EM_i$$

Donde:

- **E** es el esfuerzo estimado en personas-mes.
- **A=2.94A** es la constante de calibración estándar.
- **KSLOC** es el tamaño del proyecto en miles de líneas de código.
- **E_{SF}** es el factor de escalabilidad ajustado.
- **EM_i** productos de los valores de los factores de costo.

Se toman los valores correspondientes a los factores de costo:

$$EM_i = 1.0 \times 1.1 \times 0.9 \times 1.0 \times 1.0 = 0.99$$

Reemplazar en la fórmula

$$E = 2.94 \times (66.032)^{1.06} \times 0.99$$

$$E \approx 2.94 \times 77.32 \times 0.99$$

$$E \approx 225.07$$

3.4. Cálculo del esfuerzo y tiempo

Para el cálculo del tiempo de desarrollo del proyecto se aplica la siguiente formula:

$$T_{dev} = 3.67 \times E^{0.28}$$

Siendo **E = 225.07 personas-mes**, valor estimado previamente, se obtiene, al considerar una remuneración mensual de Bs. 3,000 por jornada completa, la inversión total estimada asciende a:

$$T_{dev} \approx 3.67 \times (225.07)^{0.28} \approx 16.45 \text{ meses}$$

Bs. 48,000

3.5. Presupuesto estimado del proyecto

Tabla 57. Total, presupuesto

Concepto	Valor
Esfuerzo estimado (E)	225.07 personas-mes
Tiempo estimado (T_dev)	16 meses
Costo mensual (jornada completa)	Bs. 3,000
Costo total estimado	Bs. 48,000

Fuente: Elaboración propia (2025).

CAPÍTULO IV

CONCLUSIONES Y RECOMENDACIONES

4.1. Conclusiones

- Se logro desarrollar una multiplataforma para la Fundación OncoFeliz, compuesta por una página web y aplicación móvil, cumpliendo con el objetivo de fortalecer su presencia en línea, facilitar las donaciones en línea y optimizar el seguimiento de beneficiarios.
- Durante la etapa de diagnóstico se identificaron las principales necesidades institucionales, entre las cuales la complejidad que presenta su proceso de donación y la falta de sistematización el seguimiento de beneficiarios. Las encuestas y entrevistas mostraron que el 90 % de los participantes estaría dispuesto a donar mediante una plataforma digital, mientras que el 45 % percibía que el proceso actual era complicado, confirmando la necesidad de transformación digital.
- Se desarrollo una plataforma web aplicando tecnologías modernas como React, Express y PostgreSQL, también se aplicó la metodología ágil SCRUM, esto permitió una planificación organizada, el resultado fue una interfaz moderna, segura y accesible. En base a las pruebas con los usuarios, el 92% y el promedio de satisfacción fue de 4,2 de 5.
- Se diseñó y desarrolló una aplicación móvil funcional y accesible, orientada al seguimiento de beneficiarios, que permite registrar citas, recordatorios y el historial clínico-social. En las pruebas de uso, el 90 % de los usuarios calificó las funciones como útiles y fáciles de manejar, el 88 % destacó la mejora en la comunicación y el monitoreo, y el 95 % recibió notificaciones oportunas, demostrando la eficacia del sistema.
- Se implementó una base de datos relacional robusta y segura utilizando PostgreSQL y Prisma ORM, garantizando la integridad y confidencialidad de la información. Su estructura permite gestionar de manera eficiente los datos de donaciones, campañas, beneficiarios y otros módulos del sistema.
- Se integraron herramientas digitales basadas en inteligencia artificial:
 - Un chatbot interactivo que automatiza más del 60 % de las consultas iniciales.
 - Un asistente virtual que reduce en un 50 % el tiempo de elaboración de contenido para campañas y eventos.

- Un asistente especializado para beneficiarios que envía recordatorios y notificaciones clínicas, logrando que el 95 % reciba información relevante y oportuna.
- Un asistente adicional que responde dudas sobre el cáncer y sus cuidados, ofreciendo información clara y a tiempo para apoyar el tratamiento.

4.2. Recomendaciones

Desde un punto de vista práctico se recomienda a la Fundación OncoFeliz, mantener en funcionamiento y en constante actualización, el chatbot como el sistema de generación automática de campañas y eventos ya que estas herramientas permiten mejorar la eficiencia, optimizar el tiempo del personal y reforzar la interacción con donantes y usuarios de la plataforma.

Desde el punto de vista metodológico, se sugiere adaptar y replicar el modelo de plataforma digital implementado en otras organizaciones sin fines de lucro con características similares, considerando las necesidades de cada contexto.

También se recomienda integrar funciones automatizadas que midan el alcance, efectividad y participación en campañas, lo que permitiría una toma de decisiones más clara y un seguimiento más preciso del impacto generado por las herramientas tecnológicas utilizadas.

ABREVIATURAS

Tabla 58. Abreviaturas y siglas

Sigla/Abreviatura	Significado completo
API	Application Programming Interface (Interfaz de Programación de Aplicaciones)
NFC	Near Field Communication (Comunicación de Campo Cercano)
DB	Database (Base de Datos)
SQL	Structured Query Language (Lenguaje Estructurado de Consultas)
UX	User Experience (Experiencia de Usuario)
UI	User Interface (Interfaz de Usuario)
HTML	HyperText Markup Language (Lenguaje de Marcado de Hipertexto)
CSS	Cascading Style Sheets (Hojas de Estilo en Cascada)
JS	JavaScript
JSON	JavaScript Object Notation (Notación de Objetos de JavaScript)
ORM	Object-Relational Mapping (Mapeo Objeto-Relacional)
SUS	Seguro Universal de Salud
ONG	Organización No Gubernamental
OMS	Organización Mundial de la Salud
IA	Inteligencia Artificial
QR	Quick Response (Código de Respuesta Rápida)
SCRUM	Marco de trabajo ágil para gestión de proyectos
COCOMO II	Constructive Cost Model II (Modelo Constructivo de Costos II)
KPI	Key Performance Indicator (Indicador Clave de Desempeño)

Fuente: Elaboración propia.

GLOSARIO

API REST

Un estilo de arquitectura de software para diseñar servicios en red. Permite que distintas aplicaciones se comuniquen entre sí utilizando el protocolo HTTP de manera estandarizada. En este proyecto, conecta el frontend (React) con el backend (Node.js) y la base de datos (PostgreSQL).

Arquitectura Cliente-Servidor

Modelo de comunicación en el que un cliente (aplicación web o móvil) solicita servicios o datos a un servidor, que procesa la petición y responde. Es la base de la plataforma digital de la Fundación OncoFeliz.

Backend

Parte de un sistema que se ejecuta en el servidor y no es visible para el usuario. Gestiona la lógica de negocio, el acceso a bases de datos, la autenticación de usuarios y la comunicación con el frontend.

COCOMO II

Modelo algorítmico de estimación de costos de software. Calcula esfuerzo, tiempo y costo de desarrollo basándose en el tamaño del sistema, la complejidad y factores de productividad. En este proyecto se empleó para el análisis de costos.

Frontend

La capa visible con la que interactúa el usuario. En este proyecto incluye la página web (React.js) y la aplicación móvil, que permiten visualizar información, realizar donaciones y acceder a funcionalidades según el rol del usuario.

Inteligencia artificial

Conjunto de técnicas que permiten a un sistema aprender y tomar decisiones de forma autónoma. En este proyecto, se integra para personalizar la experiencia de los usuarios y optimizar el seguimiento de beneficiarios.

NFC

Tecnología de comunicación inalámbrica de corto alcance. En este proyecto se utiliza en tarjetas que permiten identificar beneficiarios y facilitar las donaciones al escanearlas.

Node.js

Entorno de ejecución de JavaScript del lado del servidor. Se emplea en este proyecto para el backend, gestionando procesos como la autenticación y la comunicación con la base de datos.

ORM

Técnica de programación que facilita la interacción entre bases de datos relacionales y el código orientado a objetos. En este proyecto se usa Prisma ORM para conectar Node.js con PostgreSQL.

PostgreSQL

Sistema de gestión de bases de datos relacional de código abierto, robusto y escalable. Se utiliza en la plataforma para almacenar información de beneficiarios, donaciones y campañas.

SCRUM

Marco de trabajo ágil para el desarrollo de proyectos. Divide el trabajo en ciclos cortos llamados sprints, fomentando la entrega incremental y la colaboración del equipo.

SUS

Sistema de salud pública en Bolivia que cubre parcialmente los costos médicos. En este proyecto se menciona porque muchos medicamentos y tratamientos no están completamente cubiertos, lo que motiva el apoyo de la Fundación OncoFeliz.

Token JWT

Estándar abierto para la generación de tokens que autentican y autorizan a los usuarios. En este proyecto, se utiliza para que el frontend pueda acceder a recursos del backend sin necesidad de reenviar credenciales en cada solicitud.

UI

Interfaz de usuario. Es el diseño visual de la plataforma, incluyendo menús, botones y pantallas, con los que los usuarios interactúan directamente.

APÉNDICES

Apéndice A: Cuestionario y resultados cuantitativos

Encuesta sobre la Fundación OncoFeliz, detalla los instrumentos de recolección de datos cuantitativos utilizados en la investigación, también la presentación de los resultados mediante gráficos que fueron obtenidos de la encuesta dirigida.

A.1. Preguntas de la encuesta dirigida al público en general

El cuestionario fue implementado mediante la plataforma de Google Forms y difundido mediante medio digitales para ser respondido y evaluar la percepción la publico a la que fue dirigido.

Sección 1: Perfil del encuestado

Indique su relación con la Fundación OncoFeliz

- Familia beneficiaria
- Donante actual
- Potencial donante

Indique su edad

- Menos de 25 años
- 25 – 40 años
- 41 – 560 años
- Mas de 60 años

¿Con que frecuencia interactúa con la fundación?

- Frecuentemente (semanalmente o más)
- Ocasionalmente (una vez al mes)
- Rara vez (una vez al semestre)
- Nunca

Sección 2: Captación de fondos y donaciones

¿Como conoció la Fundación OncoFeliz?

- Redes sociales (Facebook, Instagram)
- Recomendaciones de amigos o familiares

- Eventos o campañas presenciales
- Medios de comunicación (TV, radio, periódicos)
- Otra...

Si ha realizado donaciones, ¿qué método ha utilizado?

- Transferencia bancaria
- Efectivo
- Campañas en redes sociales
- No he donado aun

¿Como calificaría la facilidad del proceso actual para donar en la fundación?

- Muy fácil
- Fácil
- Algo complicado
- Muy complicado

¿Estaría dispuesto a donar a través de una plataforma digital con opciones seguras?

- Si, definitivamente
- Si, pero necesitaría más información
- No lo se
- No

Sección 3: Seguimiento y apoyo a las familias

¿Como recibe información sobre la disponibilidad de apoyo de medicamentos?

- Eventos o campañas presenciales
- Medios de comunicación (TV, radio, periódicos)
- Redes sociales
- Llamadas telefónicas
- Visitas a la fundación
- No recibo información clara

¿Ha tenido dificultades para acceder a medicamentos o apoyo de la fundación?

- Si, frecuentemente

- Si, en algunas ocasiones
- No, nunca

Le gustaría recibir notificaciones y recordatorios sobre el tratamiento de su hijo/a mediante una aplicación móvil?

- Si, me sería de gran ayuda
- Si, pero preferiría otro medio
- No lo necesito

Sección 4: Necesidad y percepción de una plataforma digital

¿Cree que una plataforma digital ayudaría a mejorar la gestión y transparencia de la fundación?

- Si, facilitaría las donaciones y la información
- Si, pero solo si es fácil de usar
- No lo se
- No creo que sea necesario

¿Qué funcionalidades considera más importantes en una plataforma digital?
(Seleccione las más relevantes)

- Donaciones en línea con diferentes métodos de pago
- Información sobre los niños beneficiados y su progreso
- Notificaciones sobre campañas y eventos
- Registro y seguimiento de beneficiarios
- Espacio para conectar con otros donantes y voluntarios

¿Si usted es donante, le motivaría más contribuir si pudiera ver el impacto de su donación a través de la plataforma?

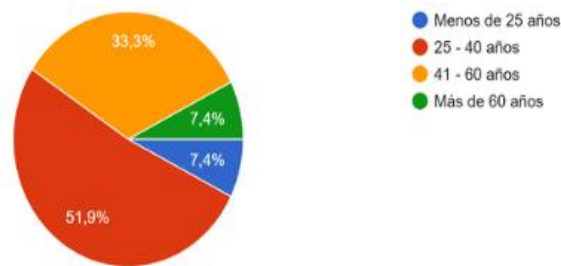
- Si, me daría más confianza
- Si, pero depende de la seguridad del sistema
- No cambiaria mi decisión de donar

A.2. Resultados gráficos de la encuesta

Sección 1: Perfil del encuestado

Figura 47. Pregunta 1: Indique su edad

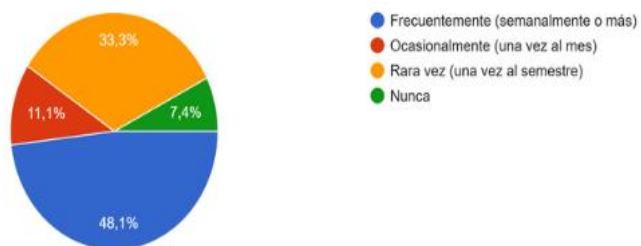
Indique su edad:



Fuente: Google forms - Encuesta aplicada Fundación OncoFeliz

Figura 48. Pregunta 2: ¿Con que frecuencia interactúa con la fundación OncoFeliz?

¿Con qué frecuencia interactúa con la fundación?



Fuente: Google forms - Encuesta aplicada Fundación OncoFeliz

Sección 2: Captación de fondos y donaciones

Figura 49. Pregunta 3: ¿Como conoció la Fundación OncoFeliz?

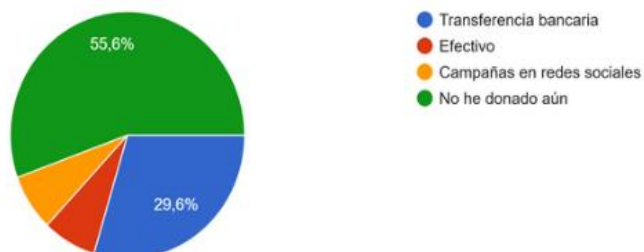
¿Cómo conoció la Fundación OncoFeliz?



Fuente: Google forms - Encuesta aplicada Fundación OncoFeliz

Figura 50. Pregunta 4: Si ha realizado donaciones, ¿Que método ha utilizado?

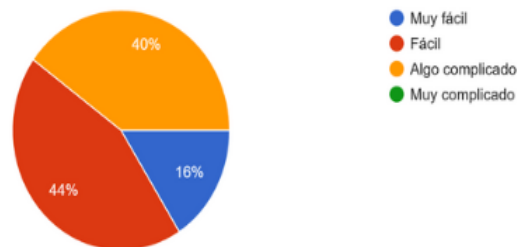
Si ha realizado donaciones, ¿Qué método ha utilizado?



Fuente: Google forms - Encuesta aplicada Fundación OncoFeliz

Figura 51. Pregunta 5: ¿Como calificaría la facilidad del proceso actual para donar a la Fundación?

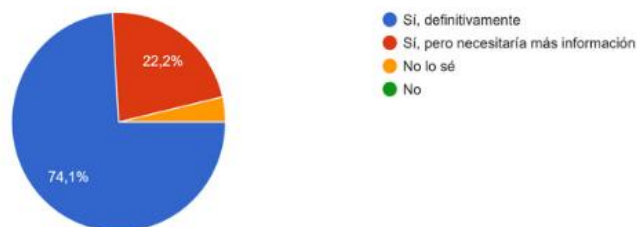
¿Cómo calificaría la facilidad del proceso actual para donar a la fundación?



Fuente: Google forms - Encuesta aplicada Fundación OncoFeliz

Figura 52. Pregunta 6: ¿Estaría dispuesto a través de una plataforma digital con opciones de pago seguras?

¿Estaría dispuesto a donar a través de una plataforma digital con opciones de pago seguras?

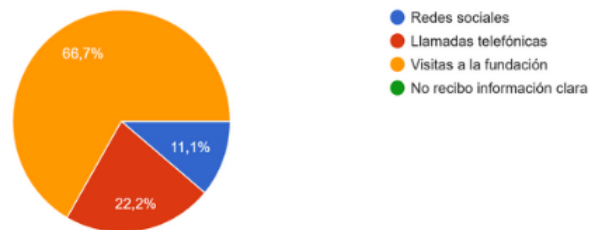


Fuente: Google forms - Encuesta aplicada Fundación OncoFeliz

Sección 3: Seguimiento y apoyo a las familias

Figura 53. Pregunta 7: ¿Cómo recibe información sobre la disponibilidad de apoyo y medicamentos?

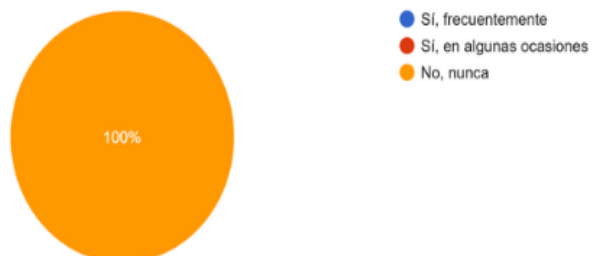
¿Cómo recibe información sobre la disponibilidad de apoyo y medicamentos?



Fuente: Google forms - Encuesta aplicada Fundación OncoFeliz

Figura 54. Pregunta 8: ¿Ha tenido dificultades para acceder a medicamentos o apoyo de la fundación?

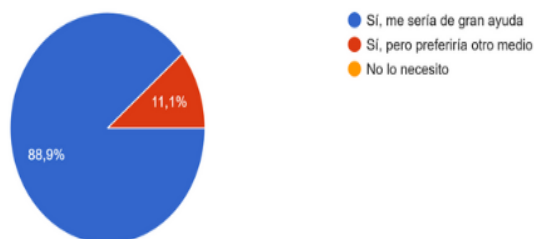
¿Ha tenido dificultades para acceder a medicamentos o apoyo de la fundación?



Fuente: Google forms - Encuesta aplicada Fundación OncoFeliz

Figura 55. Pregunta 9: ¿Le gustaría recibir notificaciones y recordatorios sobre el tratamiento de su hijo/a mediante una aplicación móvil?

¿Le gustaría recibir notificaciones y recordatorios sobre el tratamiento de su hijo/a mediante una aplicación móvil?

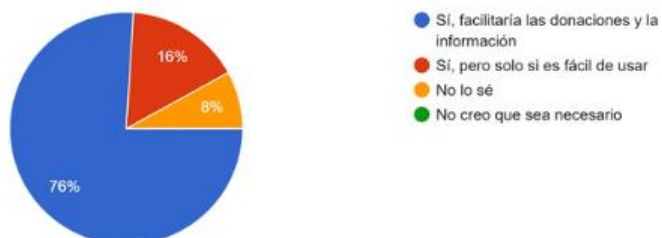


Fuente: Google forms - Encuesta aplicada Fundación OncoFeliz

Sección 4: Necesidad y percepción de una plataforma digital

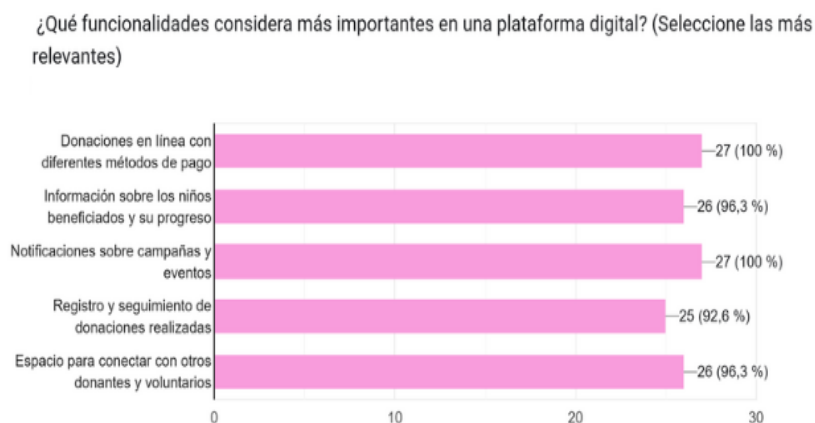
Figura 56. Pregunta 10: ¿Cree que una plataforma digital ayudaría a mejorar la gestión y transparencia de la fundación?

¿Cree que una plataforma digital ayudaría a mejorar la gestión y transparencia de la fundación?



Fuente: Google forms - Encuesta aplicada Fundación OncoFeliz

Figura 57. Pregunta 11: ¿Que funcionalidades considera más importantes en una plataforma digital?



Fuente: Google forms - Encuesta aplicada Fundación OncoFeliz

BIBLIOGRAFÍA

- Abramov, D. (2021). *Redux documentation and best practices*. Recuperado de <https://redux.js.org/>
- Bocklisch, T., Faulkner, J., Pawlowski, N., & Nichol, A. (2017). *Rasa: Open source conversational AI*. arXiv preprint arXiv:1712.05181.
- Booch, G., Rumbaugh, J., & Jacobson, I. (2010). *The unified modeling language user guide* (2nd ed.). Addison-Wesley.
- Burns, B. (2020). *Mobile app development with cross-platform frameworks*. O'Reilly Media.
- Chen, P. P.-S. (1976). The entity-relationship model—toward a unified view of data. *ACM Transactions on Database Systems*, 1(1), 9–36.
- Dumas, J. S., & Redish, J. C. (1999). *A practical guide to usability testing*. Intellect Books.
- Elmasri, R., & Navathe, S. (2017). *Fundamentals of database systems* (7th ed.). Pearson.
- Elmasri, R., & Navathe, S. B. (2020). *Fundamentals of database systems* (7th ed.). Pearson.
- Fielding, R. T. (2000). Architectural styles and the design of network-based software architectures (Doctoral dissertation). University of California, Irvine.
- Flanagan, D. (2020). *JavaScript: The definitive guide* (7th ed.). O'Reilly Media.
- Freeman, E., & Robson, E. (2020). *Head first HTML and CSS* (2nd ed.). O'Reilly Media.
- Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT Press.
- Hevner, A., March, S., Park, J., & Ram, S. (2020). Design science in information systems research. *MIS Quarterly*, 28(1), 75–105.
- Holowaychuk, T. (2014). *Express in action: Writing, building, and testing Node.js applications*. Manning Publications.
- ISO. (2018). *ISO 9241-210:2018 Ergonomics of human-system interaction — Part 210: Human-centred design for interactive systems*. International Organization for Standardization.
- Jurafsky, D., & Martin, J. H. (2021). *Speech and language processing* (3rd ed.). Pearson.
- Kumar, A. (2023). Principles of intuitive user interface design: Enhancing usability and user retention. *UX Research Journal*, 15(2), 45–59.

- Kumar, S., & Gupta, R. (2019). Automated notification systems: Improving user engagement and operational efficiency. *International Journal of Information Management*, 44, 120–130.
- Laudon, K. C., & Laudon, J. P. (2021). *Management information systems: Managing the digital firm* (17th ed.). Pearson.
- Martínez, A., & López, C. (2021). *Evaluación de impacto de plataformas digitales: Métodos y métricas*. Editorial Académica.
- Meyer, E. (2019). *CSS: The definitive guide* (4th ed.). O'Reilly Media.
- Momjian, B. (2020). *PostgreSQL: Up and running* (3rd ed.). O'Reilly Media.
- Morrison, J. (2020). *Push notifications and digital communication strategies*. TechPress.
- Nader, D. (2021). *React Native in action* (2nd ed.). Manning Publications.
- Nielsen, J. (2020). *Usability engineering*. Morgan Kaufmann.
- OpenAI. (2023). *OpenAI API documentation*. Recuperado de <https://platform.openai.com/docs>
- Pressman, R. S., & Maxim, B. R. (2020). *Software engineering: A practitioner's approach* (9th ed.). McGraw-Hill Education.
- Prisma. (2023). *Prisma documentation*. Recuperado de <https://www.prisma.io/docs>
- Resig, J., & Bibeault, B. (2013). *Secrets of the JavaScript Ninja* (2nd ed.). Manning Publications.
- Rubin, J., & Chisnell, D. (2008). *Handbook of usability testing: How to plan, design, and conduct effective tests* (2nd ed.). Wiley.
- Rumpe, B. (2016). *Modeling with UML: Language, Concepts, Methods*. Springer.
- Russell, S., & Norvig, P. (2021). *Artificial intelligence: A modern approach* (4th ed.). Pearson.
- Salamon, L. M., & Anheier, H. K. (2020). *The nonprofit sector: A research handbook* (3rd ed.). Stanford University Press.
- Sommerville, I. (2021). *Software engineering* (11th ed.). Pearson.
- Tanenbaum, A. S., & van Steen, M. (2016). *Distributed systems: Principles and paradigms* (3rd ed.). Pearson.
- Tercer Sector Digital. (2024). *Transformación digital para ONG, fundaciones y entidades sin ánimo de lucro*.
- Tilkov, S., & Vinoski, S. (2010). Node.js: Using JavaScript to build high-performance network programs. *IEEE Internet Computing*, 14(6), 80–83.
- Waters, R. D. (2018). *Digital fundraising and social media: Strategies for nonprofit organizations*. Routledge.

- Wieggers, K., & Beatty, J. (2013). *Software requirements* (3rd ed.). Microsoft Press.
- World Wide Web Consortium (W3C). (2018). *Web Content Accessibility Guidelines (WCAG) 2.1*. Recuperado de <https://www.w3.org/TR/WCAG21/>