

Universidad Católica de Santa María
Facultad de Ciencias e Ingenierías Físicas y Formales
Escuela Profesional de Ingeniería de Sistemas



Desarrollo e implementación de un sistema informático para la gestión de historias clínicas en las campañas de responsabilidad social de una universidad privada utilizando el framework Scrum

Tesis presentada por el Bachiller:

Villena Torres, Jose Miguel

ORCID: 0009-0003-8065-2438

para optar el Título Profesional de Ingeniero de Sistemas con especialidad en Sistemas de Información

Asesor:

Dr. Esquicha Tejada, José David

ORCID: 0000-0002-0191-7174

Arequipa - Perú

2025

UCSM-ERP

UNIVERSIDAD CATÓLICA DE SANTA MARÍA
INGENIERIA DE SISTEMAS
CON ESPECIALIDAD EN SISTEMAS DE INFORMACION
TITULACIÓN CON TESIS
DICTAMEN APROBACIÓN DE BORRADOR

Arequipa, 21 de Abril del 2025

Dictamen: 012225-C-EPIS-2025

Visto el borrador del expediente 012225, presentado por:

2015801251 - VILLENA TORRES JOSE MIGUEL

Titulado:

**DESARROLLO E IMPLEMENTACIÓN DE UN SISTEMA INFORMÁTICO PARA LA GESTIÓN DE
HISTORIAS CLÍNICAS EN LAS CAMPAÑAS DE RESPONSABILIDAD SOCIAL DE UNA UNIVERSIDAD
PRIVADA UTILIZANDO EL FRAMEWORK SCRUM**

Nuestro dictamen es:

APROBADO

Título Profesional/Título de Segunda Especialidad/Grado Académico a optar:

**INGENIERO DE SISTEMAS CON ESPECIALIDAD EN
SISTEMAS DE INFORMACION**

**29600028 - CORRALES DELGADO CARLO JOSE LUIS
DICTAMINADOR**



**29601217 - ROSAS PAREDES KARINA
DICTAMINADOR**



**71132586 - ANGULO OSORIO JAVIER FERNANDO
DICTAMINADOR**



Desarrollo e implementación de un sistema informático para la gestión de historias clínicas en las campañas de responsabilidad social de una universidad privada utilizando el framework Scrum

INFORME DE ORIGINALIDAD



FUENTES PRIMARIAS

1	Submitted to Universidad Católica de Santa María	5%
	Trabajo del estudiante	
2	hdl.handle.net	2%
	Fuente de Internet	
3	www.coursehero.com	1%
	Fuente de Internet	
4	tesis.ucsm.edu.pe	1%
	Fuente de Internet	
5	repositorio.utc.edu.ec	<1%
	Fuente de Internet	
6	Submitted to Pontificia Universidad Católica del Ecuador - PUCE	<1%
	Trabajo del estudiante	
7	repositorio.uladech.edu.pe	<1%
	Fuente de Internet	

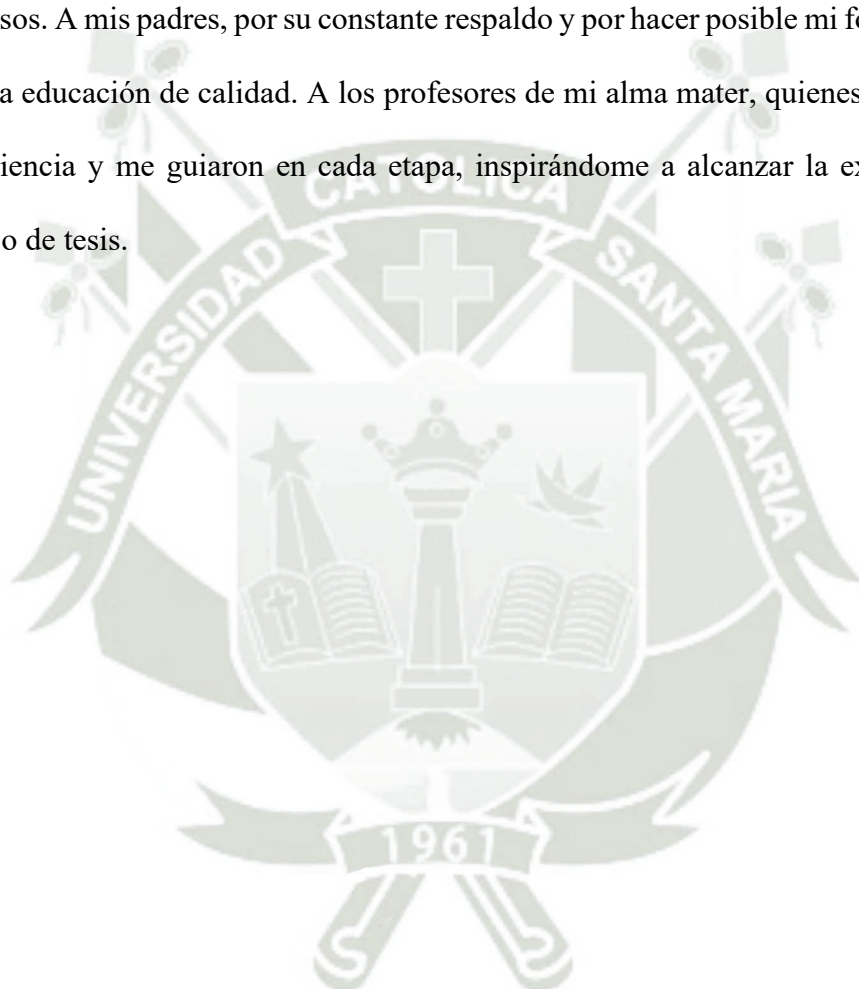
DEDICATORIA

Dedico esta tesis, ante todo, a Dios, por su infinita sabiduría y protección constante a lo largo de mi formación. A mi madre, Miriam, por su amor incondicional y el aliento diario que me impulsó a superar cada desafío. A mis hermanos, cuyo ejemplo de perseverancia y apoyo permanente fortaleció mi convicción de seguir adelante. A mi abuela Hortensia, por sus valiosas lecciones de humildad que moldearon mi carácter. A la Virgen de Chapí, por interceder y concederme esperanza en los momentos más difíciles, y al Señor de los Milagros, por su consuelo y milagros que avivaron mi fe y me dieron fuerzas para culminar este proyecto.



AGRADECIMIENTOS

Quiero expresar mi profunda gratitud a la Escuela Profesional de Ingeniería de Sistemas por proporcionarme un entorno de aprendizaje que fomentó mi crecimiento académico y personal. A la Dirección de Responsabilidad Social Universitaria, gracias por confiar en mí y permitirme aplicar mis competencias en un proyecto que busca optimizar procesos. A mis padres, por su constante respaldo y por hacer posible mi formación a través de una educación de calidad. A los profesores de mi alma mater, quienes compartieron su experiencia y me guiaron en cada etapa, inspirándome a alcanzar la excelencia en este trabajo de tesis.



RESUMEN

Las universidades desempeñan un rol fundamental en el impulso de la atención sanitaria y el desarrollo integral en zonas desfavorecidas mediante campañas de responsabilidad social. Sin embargo, la administración manual de estas campañas médicas generaba demoras y errores que afectaban la calidad del servicio, provocando insatisfacción en los beneficiarios. En este contexto, este trabajo tiene como propósito proponer una solución eficiente a esta problemática. Para ello, se diseñó e implementó una plataforma informática para la administración de historias clínicas en la Dirección de Responsabilidad Social Universitaria (DirRSU). El desarrollo se realizó aplicando la metodología SCRUM, permitiendo una construcción centrada en la colaboración del equipo y la entrega progresiva de módulos hasta completar.

La investigación adopta un enfoque descriptivo y de corte cuantitativa, sustentada en cuestionarios dirigidos a usuarios y áreas implicadas en el proceso. Tras su implementación, el sistema obtuvo un puntaje del 78.82 % en términos de usabilidad (SUS) y eficiencia operativa, reflejando su facilidad de uso y optimización en la consolidación de información. Esto permitió agilizar la generación de reportes en cada campaña y atender a más beneficiarios en menos tiempo, superando las limitaciones del proceso manual anterior.

Palabras Clave: Historia clínica, Campaña médica, Sistema Informático, SUS, SCRUM.

ABSTRACT

Universities play a fundamental role in promoting healthcare and comprehensive development in disadvantaged areas through social responsibility campaigns. However, the manual administration of these medical campaigns generated delays and errors that affected the quality of service, leading to dissatisfaction among beneficiaries. In this context, this work aims to propose an efficient solution to this problem. To this end, a software platform was designed and implemented for the management of clinical records at the Directorate of University Social Responsibility (DirRSU). The development was carried out using the SCRUM methodology, allowing for collaborative teamwork and the progressive delivery of modules until complete.

The research adopts a descriptive and quantitative approach, supported by questionnaires addressed to users and departments involved in the process. After implementation, the system obtained a score of 78.82% in terms of usability (SUS) and operational efficiency, reflecting its ease of use and optimization in information consolidation. This allowed for faster report generation for each campaign and the ability to serve more beneficiaries in less time, overcoming the limitations of the previous manual process.

Keywords: Medical records, Medical campaign, Computer System, SUS, SCRUM.

ÍNDICE

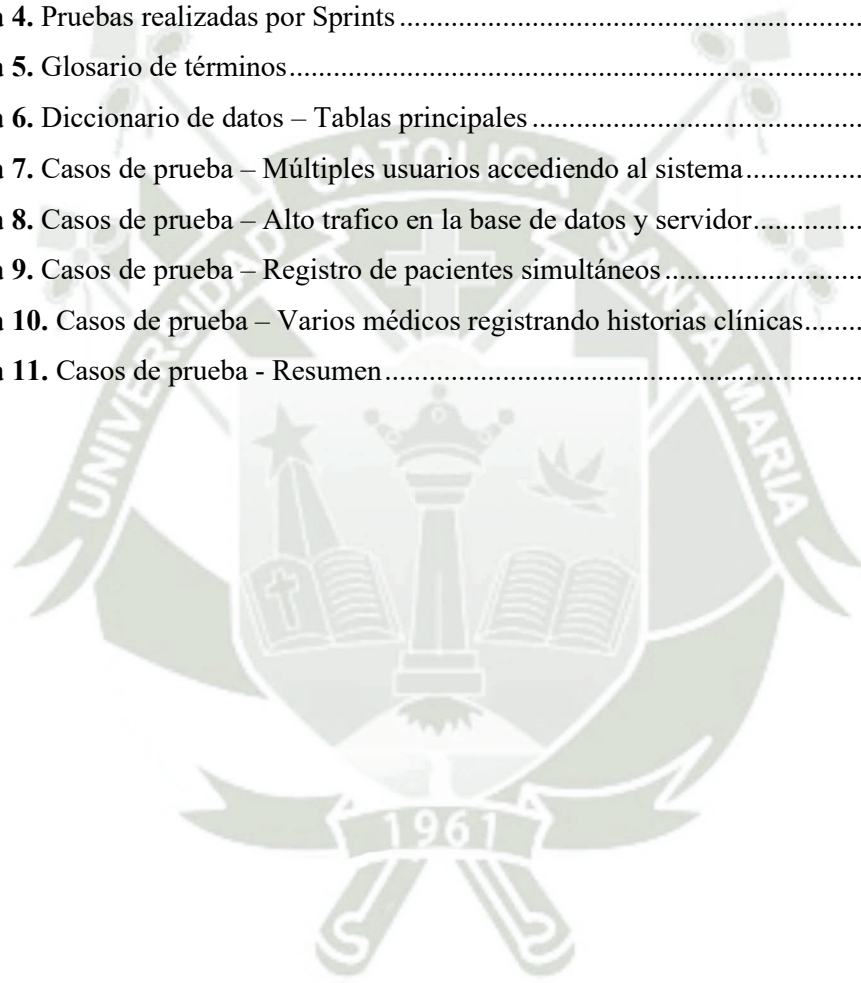
DEDICATORIA	
AGRADECIMIENTOS	
RESUMEN	
ABSTRACT	
INTRODUCCIÓN	1
CAPÍTULO I: DESCRIPCIÓN DEL PROYECTO	3
1. DESCRIPCIÓN Y OBJETIVOS DEL PROYECTO	3
1.1 Descripción del problema	3
1.2 Objetivos del proyecto	5
1.2.1 Objetivo general	5
1.2.2 Objetivos específicos	5
1.3 Alcances y limitaciones	6
1.3.1 Alcances	6
1.3.2 Limitaciones	7
1.4 Justificación	8
1.4.1 Justificación técnica	9
1.4.2 Justificación ambiental	9
1.4.3 Justificación económica	9
1.4.4 Justificación administrativa	10
1.4.5 Justificación académica	10
CAPÍTULO II: MARCO TEÓRICO	11
2. FUNDAMENTOS TEÓRICOS	11
2.1 Antecedentes de investigación	11
2.2 Bases teóricas del proyecto	15
2.2.1 Historia clínica	15
2.2.2 Responsabilidad social universitaria	16
2.2.3 Gestión de información	16
2.2.4 Sistemas de información	17
2.2.5 Framework Scrum	18

2.2.6 Sistema de historia clínica	18
2.2.7 Modelo -Vista- Controlador	18
2.2.8 ASP.NET Core.....	19
2.2.9 SQL Server.....	19
2.2.10 Angular JS.....	20
CAPÍTULO III: PRESENTACIÓN DEL PROYECTO	21
3. MARCO METODOLÓGICO.....	21
3.1 Resumen del proyecto	21
3.1.1 Descripción del proyecto a medio y largo plazo	21
3.1.2 Usuarios del proyecto	21
3.1.3 Beneficios.....	21
3.1.4 Localización	21
3.1.5 Análisis del futuro del proyecto	22
3.1.6 Riesgos que debemos afrontar.....	22
3.2 Uso de BPM en el proceso de gestión de historias clínicas	22
3.2.1 Descubrimiento del proceso.....	22
3.2.2 Diseño del proceso actual de gestión historias clínicas	23
3.2.3 Rediseño del proceso de registro de historias clínicas.....	25
3.2.4 Comparación de procesos y resultados esperados.....	27
3.3 Comparación con trabajos previos	28
CAPÍTULO IV: PLAN DE IMPLANTACIÓN DEL PROYECTO	33
4. DOCUMENTACIÓN TÉCNICA	33
4.1 Definición del proyecto	33
4.1.1 Aspectos técnicos	33
4.1.2 Aspectos económicos	33
4.2 Desarrollo del Framework SCRUM.....	35
4.2.1 Sprints.....	35
4.2.2 TasksBoard y Burndown Chart.....	36
4.3 Desarrollo de la Metodología SUS	37

4.3.1	Introducción a la metodología SUS	37
4.3.2	Objetivo de la evaluación SUS	37
4.3.3	Procedimiento de implementación del SUS	38
4.3.4	Fases de la evaluación	39
4.3.5	Análisis de resultados del SUS	39
4.3.6	Conclusiones y mejoras propuestas	40
4.4	Documentación técnica	40
4.4.1	Plan de proyecto informativo	40
4.4.2	Especificación de requisitos del software	41
4.4.3	Especificación del diseño.....	44
4.4.4	Interfaz de usuario	55
4.4.5	Documentación técnica de programación	74
4.5	Pruebas.....	77
4.5.1	Pruebas unitarias.....	77
4.5.2	Pruebas de integración.....	81
4.5.3	Pruebas de rendimiento y carga del sistema informático.....	93
4.5.4	Evaluaciones de aceptación	99
4.6	Validación de la Aplicación	103
4.6.1	Caso de estudio	103
4.6.2	Encuestas realizadas.....	103
CAPÍTULO V: VALIDACIÓN DE LA TESIS.....		108
5.1	Análisis del estado actual del proceso de gestión de historias clínicas	108
5.2	Implementación del sistema informático con el framework SCRUM.....	108
5.3	Validación de la usabilidad del sistema (System Usability Scale - SUS)....	109
5.4	Conclusión de los resultados.....	110
CONCLUSIONES.....		111
RECOMENDACIONES Y TRABAJOS FUTUROS		112
REFERENCIAS BIBLIOGRÁFICAS		114
ANEXOS		118

ÍNDICE DE TABLAS

Tabla 1. Análisis comparativo con trabajos previos	29
Tabla 2. Estimación de costos del proyecto.....	34
Tabla 3. Historias de usuario y Sprints	42
Tabla 4. Pruebas realizadas por Sprints	44
Tabla 5. Glosario de términos.....	45
Tabla 6. Diccionario de datos – Tablas principales	54
Tabla 7. Casos de prueba – Múltiples usuarios accediendo al sistema.....	95
Tabla 8. Casos de prueba – Alto tráfico en la base de datos y servidor.....	96
Tabla 9. Casos de prueba – Registro de pacientes simultáneos.....	97
Tabla 10. Casos de prueba – Varios médicos registrando historias clínicas.....	98
Tabla 11. Casos de prueba - Resumen.....	100



ÍNDICE DE FIGURAS

Figura 1. Proceso actual de gestión de historias clínicas	24
Figura 2. Proceso mejorado de gestión de historias clínicas	26
Figura 3. Entorno del sistema de historias clínicas.....	47
Figura 4. Diagrama de casos de uso	48
Figura 5. Vista lógica del sistema.....	56
Figura 6. Vista física del sistema.....	57
Figura 7. Diagrama de actividad del sistema de historias clínicas	58
Figura 8. Diagrama de clases del sistema.....	59
Figura 9. Diagrama de secuencia de iniciar sesión.....	61
Figura 10. Diagrama de secuencia de gestionar triaje	62
Figura 11. Diagrama de secuencia de configuración de campañas médicas	63
Figura 12. Diagrama de configuración de usuarios	64
Figura 13. Diagrama de secuencia de gestionar roles de usuario	65
Figura 14. Diagrama de secuencia de gestionar permisos de usuario	66
Figura 15. Diagrama de secuencia de configurar gestión de responsabilidad social universitaria	67
Figura 16. Diagrama de secuencia de configurar gestión de evaluación.....	68
Figura 17. Diagrama de secuencia de configurar gestión de compra de medicamentos y extras .69	
Figura 18. Diagrama de secuencia de configuraciones de campañas	70
Figura 19. Diagrama de secuencia de registrar medicamento	71
Figura 20. Diagrama de secuencia de asignar medicamento a campaña.....	72
Figura 21. Modelo relacional del sistema.....	73
Figura 22. Resultados de ejecución de pruebas individuales automatizadas.....	81
Figura 23. Vista de Login.....	82
Figura 24. Vista de página principal.....	82
Figura 25. Vista de usuarios del sistema	83
Figura 26. Vista de enfermedades del sistema.....	84
Figura 27. Vista de especialidades médicas	84
Figura 28. Vista de configuración de campaña – visualización	85
Figura 29. Vista de configuración de campaña – creación.....	85
Figura 30. Vista del módulo del Triaje – registro de pacientes	86
Figura 31. Vista del módulo del Triaje – visualización de los pacientes.....	87
Figura 32. Vista del módulo del Triaje – generación de consultas.....	88
Figura 33. Vista del módulo del atención consultorio – historia clínica general.....	89

Figura 34. Vista del módulo del atención consultorio – historia clínica odontológica.....	89
Figura 35. Vista del módulo del atención consultorio – historia clínica oftalmológica	90
Figura 36. Vista del módulo del atención consultorio – receta medica.....	90
Figura 37. Vista del módulo del farmacia – pantalla principal.....	91
Figura 38. Vista del módulo del farmacia – entrega de medicamentos.....	92



INTRODUCCIÓN

La presente tesis aborda la problemática del manejo de historias clínicas en las jornadas médicas gratuitas organizadas por una universidad de Arequipa. Hoy en día, este procedimiento se lleva a cabo de forma manual, lo que ocasiona retrasos y errores en los registros, así como dificultades para consultar datos de manera oportuna. Estas limitaciones impactan negativamente en la atención brindada a los pacientes, especialmente a quienes tienen recursos limitados y dependen de estas iniciativas para acceder a servicios de salud.

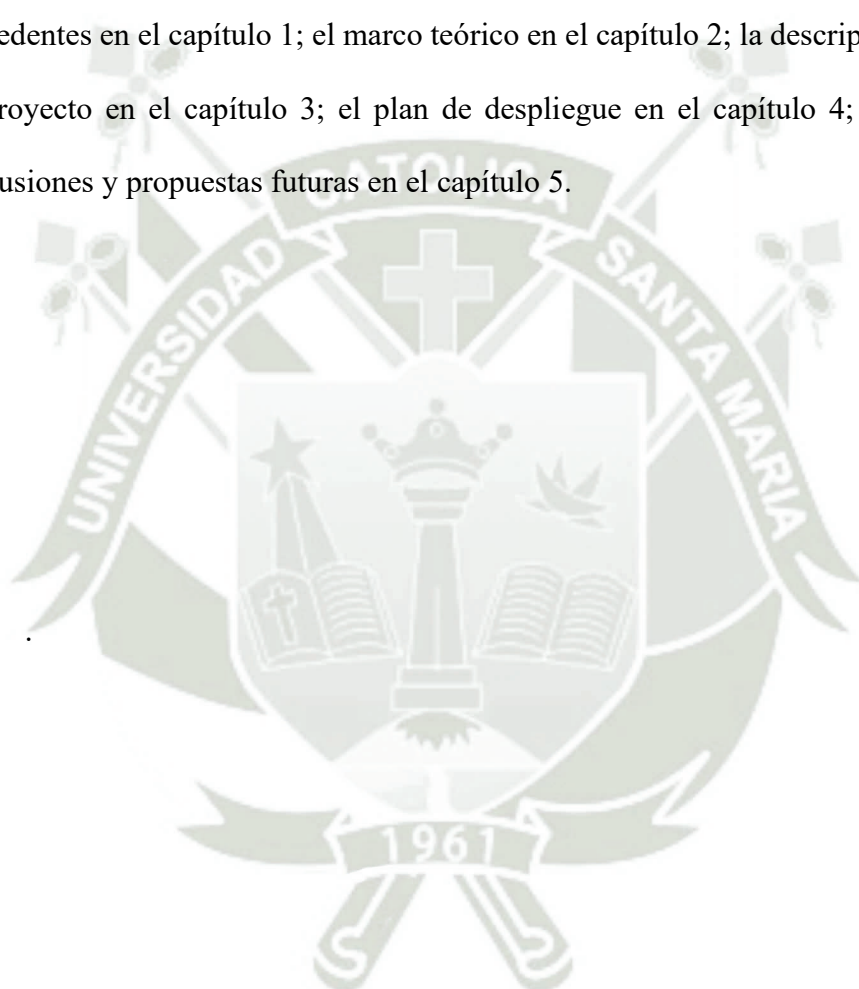
La motivación para abordar este tema de investigación es la búsqueda de una alternativa que mejore el manejo de historias bajo condiciones de recursos limitados, con el fin de optimizar tanto la rapidez como la calidad del servicio. Las campañas de responsabilidad social universitaria representan un apoyo vital para poblaciones vulnerables, por lo que disponer de una solución tecnológica adecuada puede incrementar su efectividad y alcance.

Para desarrollar este estudio, se aplicó una estrategia que integra el análisis del proceso existente con la creación de una aplicación digital, evaluando luego su usabilidad a través de la Escala de Usabilidad del Sistema. El diseño del software siguió la metodología ágil SCRUM, y se compararon los resultados del SUS con otras técnicas de evaluación de experiencia de usuario.

El valor de esta propuesta reside en su capacidad para mejorar el acceso y seguimiento de quienes participan en las campañas médicas, reduciendo errores y agilizando la administración de datos clínicos. De este modo, se fortalece el compromiso social de la universidad y se maximiza el beneficio para la comunidad.

El alcance incluye el estudio del proceso previo, el desarrollo e implementación de la herramienta digital, y la medición de su usabilidad mediante SUS. Además, se ofrecen recomendaciones para aplicar este modelo en otras actividades de responsabilidad social con contextos similares.

La tesis se estructura en cinco capítulos: el planteamiento del problema y antecedentes en el capítulo 1; el marco teórico en el capítulo 2; la descripción y resultados del proyecto en el capítulo 3; el plan de despliegue en el capítulo 4; y la evaluación, conclusiones y propuestas futuras en el capítulo 5.



CAPÍTULO I: DESCRIPCIÓN DEL PROYECTO

1. DESCRIPCIÓN Y OBJETIVOS DEL PROYECTO

1.1 Descripción del problema

Las campañas por la salud integral organizadas por la universidad privada forman parte fundamental de su programa de responsabilidad social, beneficiando a diversas comunidades de la región. Se realizan aproximadamente seis veces al año y brindan atención médica gratuita a un promedio de 800 personas por evento, a través de consultas médicas gratuitas con diversos tipos de especialidades. Esta iniciativa busca mejorar el acceso a la salud en poblaciones vulnerables, pero enfrentan importantes desafíos en la gestión del tiempo y la administración de la información clínica de los pacientes. Las plataformas informáticas resultan esenciales para organizaciones y mejorar sus procesos. (Gani et al., 2024)

Actualmente, el registro de los pacientes se realiza de manera manual mediante el uso de cuadernos físicos y tickets en papel, lo que conlleva diversos problemas. La asignación de turnos se efectúa sin un sistema automatizado, generando desorden y largas filas desde tempranas horas, lo que provoca demoras en la atención y afecta la experiencia de los asistentes. Existen muchos trabajos de investigación donde se demuestra que la incorporación de soluciones digitales optimiza la coordinación de turnos de pacientes en centros clínicos. (Ala & Chen, 2022)

Además, los datos clínicos se registran manualmente, aumentando el riesgo de pérdida de información y obligando a los pacientes a repetir el proceso de triaje y diagnóstico en cada campaña, dificultando un adecuado seguimiento de su estado de salud. Asimismo, no existe un sistema centralizado para almacenar

los historiales médicos, lo que impide la interoperabilidad entre los sistemas administrativos de la universidad y los registros de salud, limitando la toma de decisiones y el flujo de información dentro de las campañas. Resulta necesario adoptar una solución digital dado que estas herramientas optimizan la adquisición y el procesamiento de datos, mejorando la eficiencia, la toma de decisiones y la comunicación en diversas áreas. (Szymkowiak et al., 2021)

Este problema tiene diversas causas, entre ellas la falta de tecnología adecuada para gestionar de manera eficiente las historias clínicas en contextos de salud de bajo costo, el uso exclusivo de registros físicos que dificultan su almacenamiento y consulta en futuras campañas, la desorganización en la gestión de turnos debido a la ausencia de un sistema automatizado y la falta de integración con otros sistemas administrativos de la universidad.

Las consecuencias de esta problemática se evidencian en plazos inmediatos y futuros. En el corto plazo, la carencia de una plataforma para administrar historias clínicas genera ineficiencias operativas, como el aumento en los tiempos de espera y en la carga administrativa del personal de salud, el riesgo de errores en la asignación de turnos y la imposibilidad de consultar con agilidad los registros, así como limitaciones en la calidad del servicio debido a la falta de acceso rápido a los historiales clínicos. A largo plazo, la ausencia de un sistema de gestión de historias clínicas puede afectar la continuidad de la atención médica, dificultando el seguimiento de enfermedades crónicas o recurrentes, aumentando el riesgo de errores en el diagnóstico y tratamiento debido a la falta de información previa y generando un impacto negativo en la reputación de la universidad en sus programas de responsabilidad social.

Los principales afectados por esta problemática son los pacientes, quienes enfrentan dificultades para recibir un tratamiento adecuado y dar continuidad a sus consultas médicas; el personal de salud, incluyendo médicos, enfermeros y voluntarios, quienes ven aumentada su carga de trabajo debido a la gestión manual de la información; los docentes y estudiantes voluntarios, que encuentran limitaciones para realizar estudios y análisis sobre la eficacia de las campañas; y la propia universidad, que podría ver afectada su imagen institucional y la eficiencia de sus programas de responsabilidad social.

Para abordar estos problemas, se propone la puesta en marcha de una plataforma digital para administrar historias clínicas. Dicha herramienta agilizará el ingreso, resguardo y recuperación de datos clínicos en cada jornada, acortando los tiempos de espera y elevando el nivel de servicio.

1.2 Objetivos del proyecto

1.2.1 Objetivo general

Desarrollar e implementar un sistema informático que mejore el proceso de gestión de historias clínicas en las campañas de responsabilidad social de una universidad privada, empleando el framework SCRUM.

1.2.2 Objetivos específicos

- Examinar las necesidades funcionales y técnicas para el desarrollo e implementación del sistema informático de historias clínicas para campañas médicas brindadas por una universidad privada.
- Desarrollar el diseño de la arquitectura del sistema, especificando la estructura y los componentes necesarios para garantizar una correcta integración y funcionamiento.

- Aplicar el framework Scrum para gestionar el proyecto de software, facilitando la planificación, seguimiento y entrega continua de incrementos funcionales, y promoviendo la colaboración entre el desarrollador y usuarios finales.
- Validar la usabilidad del sistema utilizando la metodología SUS (Escala de Usabilidad del Sistema), con el fin de obtener métricas precisas sobre la experiencia del usuario y asegurar que el sistema sea intuitiva, eficiente y satisfactoria para los usuarios finales.

1.3 Alcances y limitaciones

1.3.1 Alcances

- a. Análisis exhaustivo de la situación actual: Se evaluará el manejo manual de historias clínicas en las campañas médicas para una universidad en Arequipa, identificando deficiencias mediante entrevistas, encuestas y revisión de registros.
- b. Automatización del proceso: Se implementará una plataforma digital, digitalizando el ingreso, almacenamiento y consulta de historias clínicas, con módulos de gestión de pacientes, citas, configuraciones y usuarios, basado en el framework SCRUM.
- c. Evaluación de usabilidad y mejorar continua: Se aplicará la metodología System Usability Scale (SUS) para medir usabilidad y satisfacción de los usuarios, analizando tiempos de registro y reducción de errores, con iteraciones para optimizar el sistema.

1.3.2 Limitaciones

- a. El Alcance geográfico: La investigación se enfoca en las campañas médicas organizadas por una universidad privada en Arequipa. Por ello, los hallazgos y conclusiones pueden no ser completamente generalizables a otras universidades o instituciones que operen en diferentes contextos geográficos, económicos o estructurales.
- b. Implementación del sistema: La puesta en marcha del sistema informático dependerá de la aceptación y los requisitos específicos de la universidad privada. Factores como la infraestructura tecnológica existente, la disposición del personal para adoptar nuevas herramientas y las políticas institucionales pueden influir en la viabilidad de la implementación.
- c. Evaluación a corto plazo: La medición del impacto del sistema informático se llevará a cabo mediante la prueba de usabilidad System Usability Scale (SUS). Sin embargo, se necesita un lapso mayor para valorar de forma completa su impacto en el rendimiento de los procesos y en el nivel de servicio sanitario conforme pase el tiempo
- d. Costos de soporte y mantenimiento: La sostenibilidad del sistema informático dependerá de la disponibilidad de recursos para su mantenimiento y actualización. No se han considerado gastos extra vinculados al entrenamiento del equipo humano, la actualización de los recursos tecnológicos o al soporte técnico permanente, lo que podría afectar su operatividad futura.
- e. Cumplimiento con normativas legales: El manejo de historias clínicas electrónicos debe cumplir con las normativas vigentes en el Perú, incluyendo la Ley de Protección de Datos Personales (Ley N° 29733) y

las regulaciones del Ministerio de Salud sobre el manejo de información médica. La puesta en marcha de la solución deberá integrar estas variables para asegurar la protección y privacidad de la información de los usuarios atendidos.

1.4 Justificación

La sistematización de la gestión del historial clínico resulta esencial para resolver los problemas identificados en las campañas médicas realizadas, tales como la ineficiente planificación de las campañas, el registro inadecuado de pacientes, la pérdida de historiales clínicos y la demora en la atención al paciente. Implementar un sistema informático permitiría abordar estas dificultades de manera efectiva, optimizando la calidad y eficacia de la atención sanitaria proporcionada. A nivel global, en particular en los países desarrollados, el uso de sistemas informáticos en hospitales, clínicas, centros de salud y campañas médicas ha demostrado ser clave para reducir los tiempos de atención, optimizar el manejo de recursos y garantizar un control más efectivo tanto del personal como de los pacientes. Estos avances se logran gracias a la implementación de normativas gubernamentales que promueven la sistematización en el sector salud. (Gonçalves y Moreira, 2020)

La creciente implementación de sistemas de historias clínicas exige un estudio exhaustivo de los elementos que condicionan su implementación efectiva. La necesidad de información precisa y fácil de consultar para respaldar decisiones médicas y de gestión justifica el estudio de la influencia organizacional en la implementación efectiva de estos sistemas. (Windari et al., 2023)

En este contexto, la presente tesis justifica la necesidad de implementar un sistema informático que permita modernizar y optimizar la gestión de campañas médicas y del historial clínico, contribuyendo así al fortalecimiento de los servicios de salud y alineándose con las mejores prácticas internacionales.

1.4.1 Justificación técnica

El sistema informático se desarrolla utilizando marcos de trabajo ágiles como SCRUM y herramientas tecnológicas modernas (ASP.NET Core y Angular), lo que asegura la escalabilidad, adaptabilidad y una implementación eficiente. Esto no solo facilita el desarrollo del sistema, sino que también asegura que cumpla con estándares de calidad y necesidades específicas de los usuarios.

1.4.2 Justificación ambiental

La informatización de la gestión de historias elimina la necesidad de papel, ayudando de manera notable a conservar recursos naturales y disminuir la generación de desechos. Este enfoque impulsa la sostenibilidad y subraya el compromiso con la conservación ambiental.

1.4.3 Justificación económica

La implementación del sistema reducirá costos operativos al eliminar gastos recurrentes en insumos como papel y almacenamiento físico. Además, optimizará los recursos del personal al reducir tiempos en procesos manuales, traduciéndose en un ahorro a largo plazo para las instituciones responsables.

1.4.4 Justificación administrativa

El sistema automatiza tareas críticas como el registro, la clasificación y el seguimiento de pacientes, mejorando la eficiencia operativa del personal administrativo. Esto permite una mejor gestión del tiempo y recursos, optimizando la organización de las campañas médicas.

1.4.5 Justificación académica

La plataforma ofrece una base de datos accesible y sistematizada, ideal para la realización de estudios clínicos y académicos. Esto fomenta la investigación médica entre estudiantes y docentes, ampliando las posibilidades de realizar estudios estadísticos y emprender proyectos innovadores en el sector sanitario.

CAPÍTULO II: MARCO TEÓRICO

2. FUNDAMENTOS TEÓRICOS

2.1 Antecedentes de investigación

Hoy en día, múltiples organismos internacionales, compañías tecnológicas y centros académicos han adoptado sistemas de historias clínicas electrónicas (HCE) como parte de sus iniciativas de responsabilidad social, en particular durante campañas médicas enfocadas en poblaciones vulnerables. Estas implementaciones constituyen un avance esencial en la digitalización de la atención sanitaria, al elevar la calidad del servicio, simplificar el seguimiento de pacientes y recopilar información valiosa para la toma de decisiones en salud.

A nivel internacional, compañías como IBM han desarrollado Watson Health, una plataforma que ha sido implementada en campañas médicas en África y América Latina, utilizando inteligencia artificial para procesar datos clínicos y mejorar el análisis de enfermedades crónicas y el diagnóstico preventivo. (IBM, 2015)

Por su parte, Microsoft, con su iniciativa AI for Health, ha trabajado junto a ONGs y gobiernos para implementar sistemas de historias clínicas digitales en regiones con acceso limitado, integrando herramientas como Power BI para el monitoreo en tiempo real de indicadores de salud. (Microsoft, 2020)

Asimismo, la Philips Foundation creó los Community Life Centers, clínicas modulares equipadas con sistemas digitales de atención médica en países como Etiopía, Kenia y Colombia, permitiendo el registro eficiente de información clínica durante campañas de atención comunitaria. (Philips, 2014)

En México, la Universidad Nacional Autónoma de México (UNAM) ha incorporado sistemas digitales de registro clínico en sus brigadas médicas como parte del Programa Universitario de Salud, optimizando la continuidad del cuidado en comunidades rurales y urbanas. (UNAM, 2020)

En el Perú, la Universidad Peruana Cayetano Heredia (UPCH) ha desarrollado el proyecto Salud Móvil, implementando sistemas de HCE en campañas médicas en zonas rurales de la Amazonía. Este sistema funciona en dispositivos móviles con posibilidad de trabajar sin conexión, adaptándose al contexto geográfico del país. (Universidad Peruana Cayetano Heredia, 2019)

Asimismo, la Universidad Pontificia Católica del Perú (PUCP) ha desarrollado una innovadora plataforma de historias clínicas para atención en zonas rurales, implantado en inicialmente en el Hospital II-1 en las áreas de Obstetricia y crecimiento del niño, dicha plataforma se distingue por su capacidad de funcionar sin internet, multiplataforma, así como su flexibilidad en el registro de la información. (Universidad Pontificia Católica del Perú, 2025)

A lo largo del tiempo, se han creado múltiples herramientas para la administración de historias clínicas, incluyendo plataformas hospitalarias, portales web en centros de salud y sistemas de programación de citas en clínicas. Según Naciones Unidas (2012), la integración de las tecnologías de la información y comunicación (TIC) en el ámbito sanitario producirá efectos sociales y económicos significativos; además, la digitalización de los registros de salud y las soluciones integradas seguirán evolucionando conforme avance la tecnología.

Para Huamán Zamudio (2017), la implementación de historias clínicas digitales agiliza la gestión de atención al paciente reduciendo la insatisfacción tanto del paciente y mejorando la operatividad de los trabajadores. En concordancia Gutarra Mejía y Quiroa Rosas (2014) mencionan que el desarrollo de herramientas digitales reduce el volumen de papeles como documentación generando productividad, agilidad, ahorro y eficiencia en cuanto a lo tradicional. En la actualidad, el desarrollo de sistemas de gestión de historias clínicas que pueda cubrir las necesidades principales se convierte en un soporte importante,

Para Rosado Espiritu (2022) con la versatilidad de los sistemas mejora la calidad en la atención médica, así mismo puedes optimizar los procesos de negocios de acuerdo con el modelo de negocio de cada centro de salud. En concordancia con Gutiérrez Flores (2017), muchos de los sistemas de gestión de historias clínicas existentes son costosos y complejos de usar, por lo que mayormente se quedan en desuso.

En Morales Ordinola (2019), realizó la implementación de un sistema de historias clínicas para el centro de salud de Pachitea, siendo viable técnica, económica y operativamente. El objetivo del trabajo radica en agilizar la atención a pacientes y evitar el excesivo uso de papeles.

La metodología utilizada consistió en investigar casos de éxito de hospitales o postas, donde posteriormente se realiza los diferentes análisis como el proceso, factibilidad técnica, económica y operativa. Por último, integrar mediante previo modelamiento y diseño de interfaces en la construcción y desarrollo del software. Los resultados obtenidos demuestran un impacto positivo en cuanto a lo ambiental

y en la productividad de los trabajadores y pacientes, además la generación de reporte permite una mejor toma de decisiones en diversas especialidades.

Las investigaciones propuestas por Gutierrez, Pelayo y Senior (2014) y Rosado Elfor (2022) destacan el desarrollo de un sistema de información basado en herramientas de software libre para la gestión de historias clínicas integrales ambulatorias, dicha investigación se siguió un diseño de campo no experimental, destacando herramientas en el análisis como entrevistas y las encuestas como técnicas de investigación.

Metodológicamente, se adoptó el ciclo de vida RUP para el desarrollo de software, lo que posibilitó la creación de una plataforma de información alineada con los requerimientos de gestión de historias clínicas integrales.

Por su parte, Pérez Ernesto, Ponce Heidi y Villalobos Dora (2014) presentan un conjunto de técnicas divididas en fases - incluyendo observación, entrevistas, encuestas y sesiones de lluvia de ideas - que luego resultan fundamentales para el diseño y la elaboración de diversos casos de uso.

En la ciudad de Córdoba Doria Oscar (2017), llevó a cabo el desarrollo de un sistema web para gestionar historias clínicas electrónicos, anteriormente manejados de manera manual. El objetivo principal fue automatizar el proceso de ingreso de historias clínicas de forma digital, con el fin de garantizar mayor rapidez y seguridad en el envío de información. Este desarrollo utilizó servicios web para crear interoperabilidad entre aplicaciones de software independientes, lo que permitió que el sistema pudiera ser integrado a otros desarrollos realizados por distintos equipos de trabajo. La metodología empleada para el desarrollo fue el

ciclo de vida RUP, que facilitó la organización de los requerimientos a lo largo de las distintas fases del proyecto.

La revisión sistemática de Vlachogianni y Tselios (2022) sobre la usabilidad percibida de la tecnología educativa utilizando la Escala de Usabilidad del Sistema (SUS) destaca la importancia de evaluar la usabilidad en este contexto. Los hallazgos muestran un buen nivel de usabilidad en general, con diferencias en la usabilidad según el tipo de tecnología educativa y el número de participantes. Estos resultados pueden ser relevantes para el desarrollo e implementación del sistema informático en gestión de historias clínicas, sugiriendo la consideración de la evaluación de usabilidad y la diversidad de usuarios potenciales en el diseño del sistema.

2.2 Bases teóricas del proyecto

2.2.1 Historia clínica

Es un registro completo y preciso de la salud de un individuo para una atención médica efectiva. Esta información, que incluye alergias, enfermedades, cirugías, hospitalizaciones, medicamentos, hábitos de vida y antecedentes familiares, proporciona a los médicos una imagen completa de la salud del paciente, permitiéndoles identificar patrones y tendencias, tomar decisiones más efectivas, evaluar el riesgo de enfermedades y tomar medidas preventivas. En resumen, la teoría de la historia clínica se basa en la idea de que un registro completo y preciso de la salud de un individuo es fundamental para una atención médica personalizada, efectiva y preventiva. (Haden, 2018).

2.2.2 Responsabilidad social universitaria

La responsabilidad social universitaria en el contexto de una campaña médica se refiere al compromiso y las acciones de una organización, institución, o grupo para contribuir al bienestar y la salud de las comunidades, especialmente aquellas más vulnerables o desfavorecidas. Este enfoque implica realizar actividades que vayan más allá de las obligaciones legales o económicas, buscando generar un impacto positivo en la sociedad y promover la equidad en el acceso a servicios de salud.

2.2.3 Gestión de información

La gestión de la información (GI) es un proceso integral que abarca la recopilación, organización, almacenamiento, procesamiento y distribución de información de diversas fuentes y formatos, incluyendo datos, documentos físicos y electrónicos, audio y video. (AIIM, s/f)

La GI busca asegurar que la información esté disponible de manera oportuna y eficiente para los usuarios autorizados, independientemente de su ubicación o dispositivo de acceso. La estructura organizacional debe ser capaz de gestionar la información a lo largo de su ciclo de vida, desde su creación hasta su eliminación, garantizando su integridad, seguridad y accesibilidad. Esto implica la implementación de sistemas y procesos que permitan la gestión eficiente de la información, incluyendo su almacenamiento, recuperación, protección y distribución. La GI también implica la consideración de las necesidades de los usuarios y la definición de políticas y procedimientos para el acceso y uso de la información. En esencia,

la GI busca optimizar la gestión de la información para maximizar su valor y utilidad para la organización. (AIIM, s/f).

2.2.4 Sistemas de información

Son herramientas que brindan servicios de información. Para lograr esto, deben recibir, almacenar, acceder, transformar, transferir y procesar información para producir el servicio deseado. Según la literatura es imposible separar lógicamente los sistemas de recuperación de los sistemas de procesamiento de información, ya que ambos son necesarios para un sistema de información efectivo. Un sistema de datos no puede producir información por sí solo, solo puede generar datos que puedan representar información para los usuarios (personas o máquinas). Un sistema de información solo es útil si se integra en un sistema de actividad (su "sistema objeto") de una manera que mejore las actividades. En consecuencia, el diseño de un sistema de datos es una subtarea del diseño del sistema de información, que a su vez es una subtarea del diseño del sistema objeto. Esto implica que el diseño del sistema de información debe considerar tanto aspectos no formales (personas) como formales (datos y algoritmos). El diseño de un sistema de datos, incluyendo el diseño de bases de datos, debe preceder al diseño infológico: el diseño del sistema objeto (o diseño organizacional) y el diseño de información (o diseño conceptual). Estos pasos son tan exigentes como el diseño del sistema de datos y requieren actividades de proyecto extensas, trabajo sistemático y documentación. Se necesitan habilidades más allá de las técnicas de procesamiento de datos para estas tareas. (Langefors, 1977)

2.2.5 Framework Scrum

Scrum es un framework diseñado para abordar problemas complejos y adaptativos, con el objetivo de entregar productos de la mayor calidad posible de manera productiva y creativa. (Zayat & Senvar, 2020)

Este framework combina principios Lean y Agile para ayudar a los equipos a entregar productos. Es importante destacar que Scrum no es una herramienta de gestión de proyectos, sino un marco para la entrega de productos. Scrum es considerado el método más establecido de entrega de productos Agile en la práctica. Su enfoque se centra en la colaboración, la flexibilidad y la adaptación continua para lograr resultados óptimos. (Zayat & Senvar, 2020).

2.2.6 Sistema de historia clínica

Un Sistema de Historia Clínica es una plataforma tecnológica diseñada para la gestión, almacenamiento y acceso organizado de información médica y administrativa relacionada con los pacientes en el ámbito de la salud. Este sistema digitaliza y centraliza la información, reemplazando los registros físicos con soluciones electrónicas que permiten un manejo más eficiente y seguro de los datos. (MDN Web Docs, s/f).

2.2.7 Modelo -Vista- Controlador

MVC (Modelo-Vista-Controlador) es un patrón de diseño de software que se centra en la separación de responsabilidades entre la lógica de negocio, la presentación de datos y el control de interacciones, facilitando así el desarrollo y mantenimiento de aplicaciones al permitir a los desarrolladores trabajar en diferentes aspectos de forma independiente. Este enfoque,

destacado por su "separación de preocupaciones", promueve la reutilización de código, la modularidad y la escalabilidad de las aplicaciones, siendo un método ampliamente utilizado con variaciones como MVVM, MVP y MVW que se basan en los principios fundamentales de MVC. (MDN Web Docs, s/f)

2.2.8 ASP.NET Core

ASP.NET Core es una versión moderna y de código abierto del framework de desarrollo web ASP.NET, diseñada para crear aplicaciones web escalables y de alto rendimiento en la plataforma .NET. A diferencia de sus predecesores, ASP.NET Core es multiplataforma, lo que significa que puede ejecutarse en Windows, Linux y macOS, ofreciendo flexibilidad y opciones de implementación. El framework se actualiza anualmente para garantizar su compatibilidad con las últimas tecnologías y permite ejecutar diferentes versiones de forma simultánea, facilitando la evolución de aplicaciones heredadas. Además, cuenta con una extensa documentación, constantemente actualizada, que incluye guías y tutoriales para desarrolladores principiantes. En resumen, ASP.NET Core es una plataforma sólida y versátil para crear aplicaciones web de alto rendimiento en una variedad de entornos. (Zang, 2023)

2.2.9 SQL Server

SQL Server es un Sistema de Gestión de Bases de Datos Relacionales (RDBMS) desarrollado por Microsoft, con una historia que se remonta a la década de 1980. Su código original fue desarrollado por Sybase Inc. y, después de una serie de colaboraciones y cambios, Microsoft asumió el desarrollo y la comercialización de SQL Server para sus propios sistemas

operativos en 1994. Actualmente, SQL Server está disponible en plataformas de 64 bits Windows, Linux y Azure Cloud, y se caracteriza por una interfaz gráfica de administración (SQL Server Management Studio) y un lenguaje procedural llamado Transact-SQL (T-SQL). En resumen, SQL Server es un sistema de gestión de bases de datos relacional robusto y ampliamente utilizado, que ha evolucionado significativamente desde sus inicios y ofrece una amplia gama de características y herramientas para gestionar datos de forma eficiente. (Actian, 2023)

2.2.10 Angular JS

AngularJS, también conocido como Angular, es un framework MVC desarrollado por Google para el desarrollo web front-end, que facilita la creación de aplicaciones SPA (Single-Page Applications) utilizando un patrón MVVM para separar la lógica de la presentación y mantenerlas conectadas a través del data binding. Angular es un framework flexible y de código abierto que permite la comunicación con servidores, la recuperación de datos y la creación de aplicaciones web modernas mediante componentes reutilizables. Destaca por su facilidad de uso y amplio soporte, lo que lo convierte en una herramienta poderosa y versátil. (Blog IfGeekThen, s/f)

CAPÍTULO III: PRESENTACIÓN DEL PROYECTO

3. MARCO METODOLÓGICO

3.1 Resumen del proyecto

3.1.1 Descripción del proyecto a medio y largo plazo

Este sistema informático para gestión de historias clínicas se incorporará en las campañas que realiza la universidad privada de Arequipa, con la finalidad de mejorar la gestión de historias clínicas del paciente.

3.1.2 Usuarios del proyecto

- Docentes de la Escuela Profesional de Medicina Humana, Odontología y Enfermería.
- Personal de Responsabilidad Social Universitaria
- Estudiantes de últimos años de la escuela de medicina, odontología y enfermería de una Universidad de Arequipa.

3.1.3 Beneficios

Los docentes, estudiantes y personal de Responsabilidad Social Universitaria contarán con un recurso de gestión de historias clínicas con el cual podrán trabajar en las diversas campañas médicas que se realice y mejorar la atención a los pacientes.

3.1.4 Localización

Se realizará en una Universidad de la ciudad de Arequipa para diversas localidades de la región Arequipa.

3.1.5 Análisis del futuro del proyecto

Se espera que este trabajo sea usado como herramienta de apoyo en la gestión de historia clínicas y ayude a los estudiantes a generar análisis estadístico en base a reportes de cada campaña según lo disponga la dirección de responsabilidad social universitaria en su momento.

3.1.6 Riesgos que debemos afrontar

- Económicos: El costo de contar con un dispositivo móvil para la implementación.
- Tecnológicos: El dispositivo móvil deberá contar con especificación técnicas (SIM, Wi-Fi, 11 pulgadas, 128 GB – Almacenamiento, 4 GB - RAM)

3.2 Uso de BPM en el proceso de gestión de historias clínicas

3.2.1 Descubrimiento del proceso

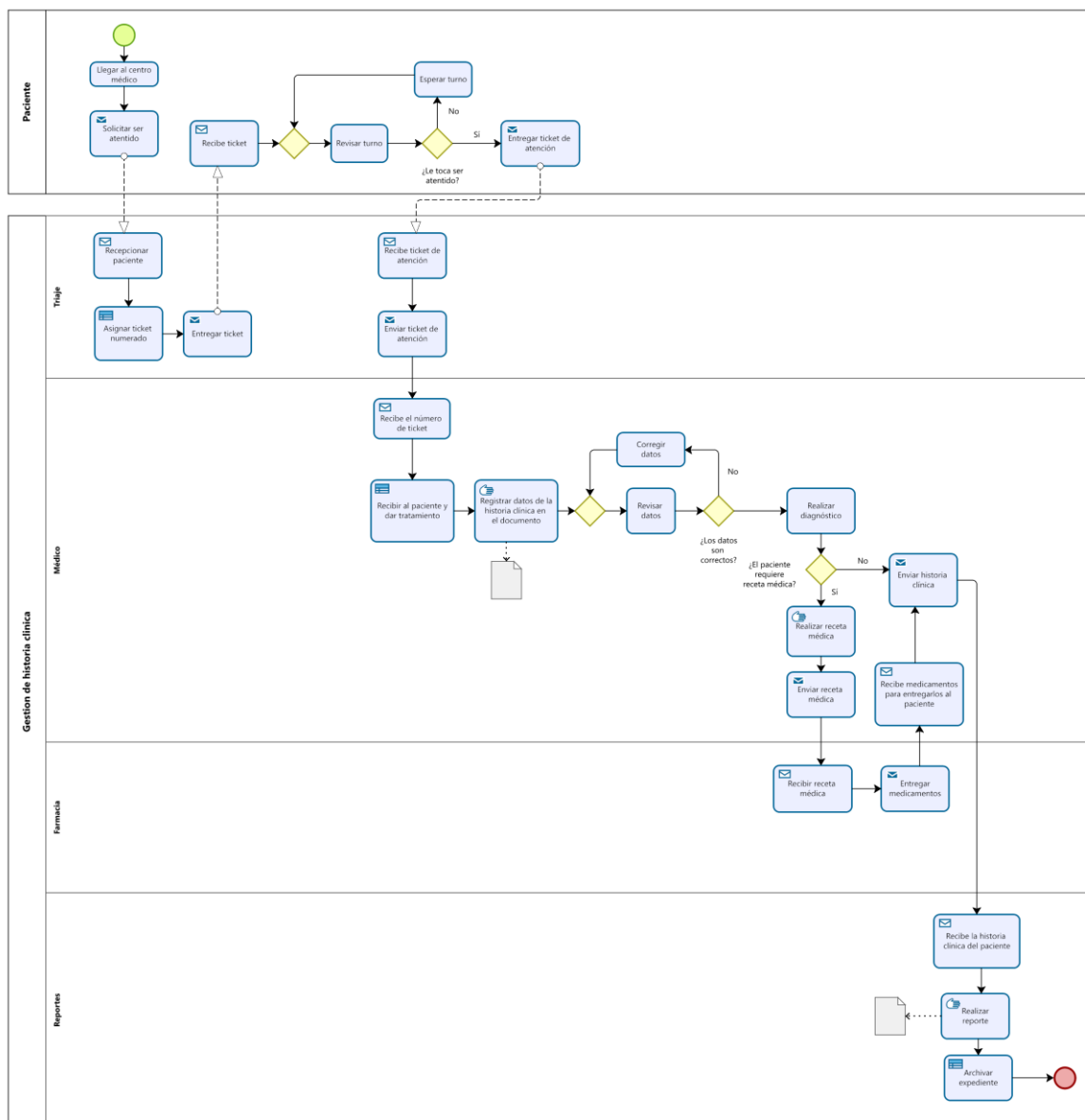
El proceso actual de gestión de historias clínicas sigue los siguientes pasos generales: primero, el paciente llega al establecimiento y solicita atención, momento en el que se le asigna un número de ticket; luego, se verifica si el paciente es nuevo o ya existe en el sistema, registrando sus datos personales si es nuevo. Se realiza la clasificación y priorización del caso según su urgencia (triaje), asignándole un orden de atención basado en esta clasificación. Posteriormente, el paciente es llamado a consulta médica, donde el médico realiza una evaluación, establece un diagnóstico y prescribe el tratamiento correspondiente, de ser necesario le proporciona una receta desde Farmacia y registra su historia clínica para que le archiven el expediente.

3.2.2 Diseño del proceso actual de gestión historias clínicas

A partir del diagrama BPMN de la Figura 1, se puede inferir que el proceso actual de gestión de citas y atención a pacientes en el establecimiento de salud es manual lo que genera diversas limitaciones. El proceso incluye etapas como el registro del paciente, donde al llegar se le asigna un número de ticket y se verifica si es un paciente nuevo o ya existente, registrando manualmente sus datos básicos en caso de ser necesario. Luego, se procede a la clasificación y espera, asignando el turno según un orden predefinido. En la etapa de atención médica, el paciente es evaluado por un profesional de la salud, quien realiza el diagnóstico y prescribe el tratamiento.

Figura 1.

Proceso Actual de Gestión de Historias clínicas



Fuente: Elaboración propia

El problema central de este proceso radica en la gestión manual de las historias clínicas, que implica un almacenamiento físico en carpetas o archivadores, lo que limita el acceso eficiente a la información, restringiéndolo solo al personal autorizado y ralentizando la búsqueda de datos específicos.

Además, este enfoque es propenso a errores humanos, como pérdida de documentos o dificultades para mantener la integridad de la información.

Asimismo, la generación de reportes o estadísticas se convierte en un proceso laborioso y propenso a inconsistencias, evidenciando la necesidad de modernizar este sistema para mejorar su precisión y eficiencia.

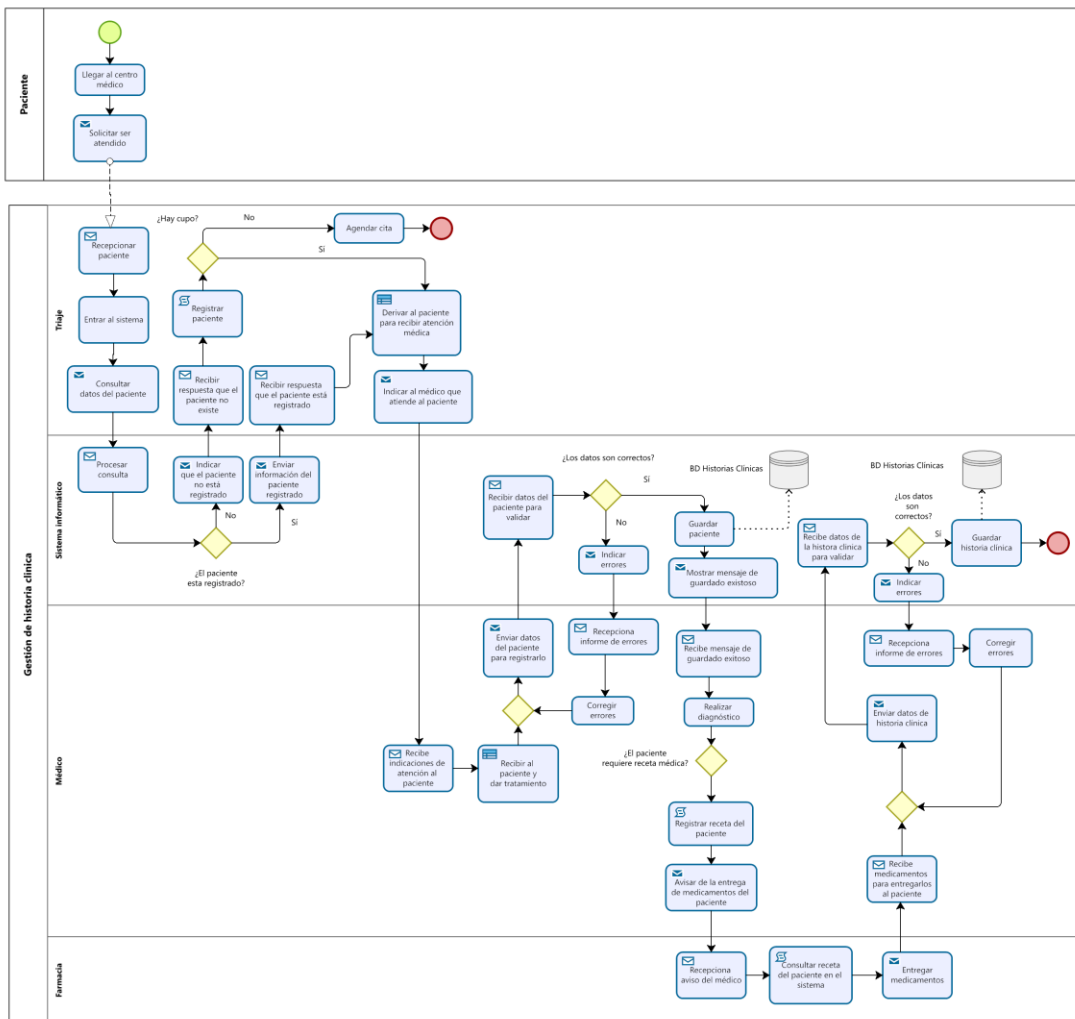
El rediseño del proceso actual de gestión de historias clínicas es necesario debido a que este se realiza de manera manual y en formato físico, lo que genera múltiples limitaciones y problemas operativos. El almacenamiento en papel ocupa espacio físico considerable y dificulta el acceso rápido a la información, ya que la búsqueda de datos específicos se vuelve lenta y dependiente de sistemas de organización susceptibles a errores. Además, el manejo manual incrementa el riesgo de pérdida o deterioro de documentos importantes, lo que compromete la continuidad de la atención médica. Este enfoque también dificulta la generación de reportes o estadísticas, ya que recopilar y analizar datos de registros físicos es un proceso laborioso y propenso a inconsistencias. Por otro lado, la falta de digitalización reduce la capacidad de realizar análisis avanzados que podrían mejorar la calidad de la atención, como identificar patrones o riesgos en la salud de los pacientes. En un entorno donde la eficiencia y la precisión son cruciales, modernizar el sistema mediante la implementación de un sistema de historias clínicas no solo optimiza el tiempo y los recursos, sino que también mejora la seguridad, la accesibilidad y la calidad del servicio ofrecido a los pacientes.

3.2.3 Rediseño del proceso de registro de historias clínicas

En la Figura 2 el rediseño propuesto se enfocará en implementar un sistema de gestión de historias clínicas (HC) que automatice la mayoría de las tareas manuales.

Figura 2.

Proceso Mejorado de Gestión de Historias clínicas



Fuente: Elaboración propia

Para el registro inicial del paciente, se usará un formulario digital que permitirá capturar datos de manera rápida y precisa, reduciendo errores. En el agendamiento de citas, estas se almacenarán directamente en la base de datos central del sistema, evitando conflictos de horarios y mejorando la organización. Durante la consulta, se incorporarán sistemas de dictado para transcribir automáticamente las notas médicas, plantillas personalizadas para diferentes tipos de consultas y dispositivos médicos conectados para registrar datos automáticamente. Las historias clínicas se almacenarán en una base de datos

centralizada, accesible de forma segura, con capacidades de búsqueda inteligente y análisis de datos para identificar patrones y riesgos.

Además, se generarán automáticamente documentos médicos, como recetas o informes, con firma electrónica para autenticar su validez. La comunicación con los pacientes se mejorará a través de un portal seguro donde podrán acceder a sus historias clínicas, solicitar citas y recibir notificaciones sobre resultados o tratamientos. Este rediseño traerá múltiples beneficios, como mayor eficiencia al automatizar tareas, mejor acceso a la información desde cualquier lugar, reducción de errores al estandarizar procesos, mejora en la calidad de la atención gracias a datos completos y actualizados, y mayor satisfacción del paciente por una comunicación más fluida y cómoda.

3.2.4 Comparación de procesos y resultados esperados

La comparación entre el proceso actual de gestión de historias clínicas y el proceso rediseñado evidencia mejoras significativas en la eficiencia y calidad de la atención. En el proceso actual, la gestión manual con papeles implica que tareas como registrar datos del paciente, buscar información en archivos físicos y generar documentos pueden demorar hasta 30 minutos debido a la necesidad de buscar, organizar y registrar información de manera manual. Con el rediseño, la automatización y digitalización de estas tareas permiten reducir este tiempo a aproximadamente 10 minutos.

Esto se logra mediante el uso de un sistema de historias clínicas que agilice el registro inicial con formularios digitales, facilita el acceso inmediato a la información almacenada en una base de datos centralizada y automatiza la generación de documentos. Además, el nuevo proceso minimiza los errores

humanos y mejora la capacidad de respuesta al proporcionar herramientas de búsqueda inteligente y análisis de datos. Como resultado, el tiempo ahorrado no solo mejora la eficiencia operativa, sino que también permite a los profesionales de la salud dedicar más tiempo a la atención directa del paciente, elevando la calidad del servicio ofrecido.

3.3 Comparación con trabajos previos

En esta sección se presenta una comparación entre los principales trabajos previos que abordan la implementación de sistemas informáticos en el ámbito de la salud y la gestión de historias clínicas, en relación con el desarrollo de la tesis actual. A través de esta comparación, se destacan los aspectos diferenciadores y las ventajas del enfoque propuesto en la tesis, especialmente en términos de tecnologías, metodologías, escalabilidad, usabilidad y contexto de aplicación.

La Tabla 1 presenta un análisis comparativo entre la tesis actual y otros trabajos previos realizados en el ámbito del desarrollo y evaluación de sistemas de información y tecnologías aplicadas a la salud.

La Tabla 1 compara la tesis actual con trabajos previos sobre sistemas de información y tecnologías aplicadas a la salud, destacando su enfoque en una comunidad universitaria con recursos limitados. A diferencia de estudios anteriores que varían en tecnología (de PHP a microservicios) y metodologías (RUP o sin especificar), la tesis usa ASP.NET Core, Angular y Scrum para equilibrar simplicidad e innovación. Sobresale en validación al aplicar el System Usability Scale (SUS) y priorizar la experiencia del usuario.

Tabla 1.
Análisis comparativo con trabajos previos

Aspecto	Huamán (2017)	Vilendrer et al. (2020)	Zaki et al. (2022)	Magalhães et al. (2020)	Zaki et al. (2022)	Tesis Presente
Contexto	Centro de salud rural (Cañete)	Pacientes hospitalizados (COVID-19)	Desarrollo de aplicaciones de salud en la nube	Organizaciones sanitarias privadas	Desarrollo de aplicaciones de salud en la nube.	Comunidad universitaria con recursos limitados
Análisis comparativo	Comparación del sistema con la gestión manual previa en centro de salud rural, analizando tiempo de atención y reducción de errores.	Evaluación del uso de telemedicina en hospitales durante COVID-19, contrastando con atención presencial en términos de eficiencia y seguridad.	Comparación de arquitecturas basadas en microservicios frente a sistemas monolíticos en términos de escalabilidad y tiempo de respuesta.	Análisis de interoperabilidad entre distintos sistemas de salud privados para mejorar la gestión de información médica.	Evaluación del impacto de la descentralización de datos en el rendimiento y seguridad de aplicaciones de salud.	Comparación del nuevo sistema con la gestión previa en comunidad universitaria, considerando facilidad de uso y reducción de errores en el manejo de datos médicos.
Seguridad y protección de datos	Recomendación: Implementar controles de acceso físico y digital a documentos médicos siguiendo la Ley de Protección de Datos Personales de Perú (Ley N° 29733).	Recomendación: Aplicar estándares de privacidad en telemedicina, como la amonificación de datos sensibles y el uso de consentimiento informado digital.	Recomendación: Implementar autenticación multifactor y cifrado de extremo a extremo para la protección de datos en entornos de microservicios.	Recomendación: Usar normativas de interoperabilidad seguras como HL7 y FHIR para garantizar el intercambio confiable de información médica.	Recomendación: Aplicar descentralización con bases de datos encriptadas y accesos limitados según roles para reducir vulnerabilidades de seguridad.	Recomendación: Cumplir con la Ley de Protección de Datos Personales en Perú (Ley N° 29733), implementando cifrado AES-256 y registros de auditoría en el sistema.
Tecnología usada	PHP, MySQL	Telemedicina, tecnologías accesibles	Microservicios, tecnologías en la nube	Sistema de información de salud (HIS) 360°, interoperabilidad	Microservicios, REST, AMQP, bases de datos descentralizada	ASP.NET Core, Angular, enfoque monolítico
Metodología o marco de trabajo de desarrollo	RUP (Proceso Unificado Rational)	No especificada (enfoque flexible)	No especificada (marco de microservicios)	Enfoque conceptual de HIS 360°	Microservicios estructurados en fases: requisitos, análisis, diseño, desarrollo y despliegue.	Scrum, desarrollo iterativo
Validación de resultados	IBM SPSS, análisis cuantitativo	No se menciona específicamente	No se menciona específicamente	Evaluación empírica con énfasis en la usabilidad	Evaluación cuantitativa en Azure y encuesta a expertos	System Usability Scale (SUS) para evaluar la usabilidad

Enfoque en usabilidad	No específico, pero implícito en la mejora de procesos	Evaluación de aceptación por usuarios en telemedicina	No específico en usabilidad	Interoperabilidad y personalización del sistema	Diseño modular, mejora pruebas ágiles y reduce complejidad operativa.	Enfoque en usabilidad mediante el SUS
Desafíos principales	Pérdida de documentos, colas largas, resistencia al cambio	Adaptación a nuevas tecnologías, impacto en los flujos de trabajo	Escalabilidad y rendimiento	Resistencia al cambio, validación empírica	Interoperabilidad, resistencia al cambio, y manejo de requisitos avanzados.	Resistencia al cambio, validación de usabilidad
Impacto en la Seguridad	Mejora en la organización de los documentos médicos	Reducción de exposición a patógenos mediante telemedicina	No se menciona explícitamente, pero puede mejorar la seguridad al usar microservicios	Mejora en la seguridad de datos mediante interoperabilidad	Mejora seguridad con bases de datos independientes y APIs.	Mejora en la seguridad de los datos mediante digitalización
Adaptabilidad a escenarios grandes	Limitado a contexto rural con pocos recursos	Adaptable a entornos hospitalarios, no a grandes escalas	Altamente adaptable a sistemas grandes y distribuidos	Adaptable a organizaciones grandes, con enfoque global	Altamente escalable para sistemas distribuidos y grandes volúmenes de solicitudes.	Potencial de escalabilidad futura mediante adaptación de arquitectura

Fuente: Elaboración propia

La Tabla 1 presenta un análisis comparativo entre la tesis actual y otros trabajos previos realizados en el ámbito del desarrollo y evaluación de sistemas de información y tecnologías aplicadas a la salud. Los estudios analizados abarcan diversos contextos, desde un centro de salud rural en Cañete hasta aplicaciones para pacientes hospitalizados por COVID-19. También se incluyen sistemas complejos enfocados en la interoperabilidad para organizaciones privadas. La tesis actual, en contraste, se centra en una comunidad universitaria con recursos limitados, adaptándose a las restricciones presupuestarias y tecnológicas de su entorno.

El uso de tecnologías varía considerablemente en los estudios previos. Mientras que Huamán emplea tecnologías tradicionales como PHP y MySQL, los

trabajos más recientes adoptan arquitecturas avanzadas basadas en microservicios y tecnologías en la nube. (Zaki et al., 2022)

La tesis presente utiliza un enfoque monolítico con ASP.NET Core y Angular, buscando un equilibrio entre la innovación tecnológica y la simplicidad operativa para facilitar su implementación. En cuanto a las metodologías o marco de trabajo de desarrollo, algunos estudios no especifican un marco claro. (Vilendrer et al., 2020; Zaki et al., 2022)

Huamán (2017), utilizan enfoques formales como RUP. La tesis actual opta por Scrum, un enfoque ágil e iterativo que permite una mayor flexibilidad y una entrega continua de valor, adaptándose mejor a las necesidades del proyecto.

La validación de resultados es otro aspecto clave. Mientras que algunos trabajos previos carecen de un proceso explícito de validación empírica, la tesis actual se distingue por el uso del System Usability Scale (SUS), una metodología ampliamente reconocida para medir la usabilidad. Esto proporciona datos objetivos y claros sobre la experiencia del usuario, destacándose frente a los estudios que no abordan directamente este aspecto.

Aunque la usabilidad está implícita en ciertos estudios, como en la mejora de procesos, otros priorizan la aceptación del usuario o la interoperabilidad. En contraste, la tesis actual coloca la usabilidad en el centro de su propuesta mediante el SUS, demostrando un compromiso con la experiencia del usuario final como un eje fundamental en la implementación del sistema.

Los desafíos identificados en los trabajos analizados incluyen problemas comunes como la resistencia al cambio y la necesidad de escalabilidad. La tesis actual aborda estos retos mediante la validación de usabilidad y la mejora en la

seguridad de los datos, logrando una solución robusta y adaptable. Además, se distingue por su enfoque en la digitalización de datos, lo que no solo refuerza la seguridad, sino que también sienta las bases para una escalabilidad futura mediante la optimización de la arquitectura.

Por último, en términos de adaptabilidad, se limitan a un contexto rural con pocos recursos, la tesis presente demuestra un potencial significativo para escalar y adaptarse a contextos más grandes y diversos. Este enfoque, junto con la mejora en la seguridad mediante la digitalización de datos, posiciona al sistema propuesto como una solución confiable y alineada con las mejores prácticas identificadas en los estudios previos.

CAPÍTULO IV: PLAN DE IMPLANTACIÓN DEL PROYECTO

4. DOCUMENTACIÓN TÉCNICA

4.1 Definición del proyecto

4.1.1 Aspectos técnicos

Para el desarrollo del Sistema Informático se utilizará Angular que es un framework de código abierto diseñada para diseñar interfaces de usuario, así como .Net. para la parte de BackEnd teniendo como gestor de base de datos a SQL Server.

4.1.2 Aspectos económicos

Para la realización del proyecto el presupuesto del proyecto incluye diferentes partidas de software, hardware, servicios, recursos humanos e imprevistos, sumando un total de S/22,070.00 para un periodo de 8 meses. En la categoría de software, se contemplan las licencias de Windows 10, cuyo costo es de S/1000.00 por toda la duración del proyecto, y Microsoft Office con un costo total de S/400.00; el uso de herramientas de desarrollo como Visual Studio y SQL Server no presenta costo adicional. En cuanto al hardware, se requiere una laptop HP con procesador Core i7 de 8ª generación, cuyo costo es de S/2700.00, y una memoria RAM de 20 GB, que asciende a S/650.00. En servicios, el acceso a internet tiene un costo total de S/640.00 y la energía eléctrica de S/480.00. La categoría humana incluye los salarios del desarrollador y del encargado de Responsabilidad Social Universitaria (RSU), con un monto de S/1000.00 mensuales para cada uno, totalizando S/8,000.00 por persona en el periodo del proyecto. Finalmente, se asigna un fondo de S/200.00 para imprevistos.

Tabla 2.*Estimación de costos del proyecto*

Ítem	Cantidad	Precio Mensual (S/.)	Cantidad de Meses	Total (S/.)
SOFTWARE				
Windows 10	1	1000	8	1000
Visual Studio + Visual Studio Code	1	0	8	0
Microsoft Office	1	400	8	400
SQL Server	1	0	8	0
HARDWARE				
Laptop HP Core i7 8TH Gen.	1	2700	8	2700
Memoria RAM de 20 GB	1	650	8	650
SERVICIOS				
Internet	1	80	8	640
Energía Eléctrica	1	60	8	480
RECURSOS HUMANOS				
Desarrollador	1	1000	8	8000
Encargado de RSU (Responsabilidad Social)	1	1000	8	8000
IMPREVISTOS				200
TOTAL				22070

Fuente: Elaboración propia

En la Tabla 2 el presupuesto presentado detalla los costos estimados para un proyecto de 8 meses, con un total de S/22,070.00.

4.2 Desarrollo del Framework SCRUM

4.2.1 Sprints

Durante el proceso de desarrollo, se realizaron 17 Sprints organizados en cinco fases: Planificación, Análisis, Diseño, Implementación y Pruebas.

A continuación, se ofrece un resumen de cada sprint:

Durante la Fase de Planificación (25 días), se llevaron a cabo cuatro Sprints enfocados en definir los fundamentos del proyecto. En el Sprint 1 (5 días), se seleccionó el lenguaje de programación más adecuado tras una evaluación comparativa de varias opciones. En el Sprint 2 (6 días), se eligió y configuró la plataforma de desarrollo, preparando el entorno de trabajo. El Sprint 3 (7 días) estuvo dedicado a establecer las metodologías ágiles a utilizar, como CI/CD, y las herramientas necesarias para el desarrollo. Finalmente, en el Sprint 4 (7 días), se especificaron tanto los requisitos funcionales como los no funcionales del sistema, asegurando que el proyecto cubriera las necesidades del cliente.

En la Fase de Análisis (9 días), se realizaron dos Sprints. El Sprint 5 (5 días) se centró en la definición del modelo del sistema web, proporcionando una comprensión clara de su estructura. Luego, el Sprint 6 (4 días) permitió determinar las tecnologías y herramientas específicas para el desarrollo y las pruebas.

La Fase de Diseño (25 días) incluyó cuatro Sprints. En el Sprint 7 (7 días), se construyeron casos de uso detallados para entender la interacción de los usuarios con el sistema. El Sprint 8 (8 días) estuvo dedicado a la creación de un diagrama de clases, definiendo entidades y relaciones. Posteriormente, en el Sprint 9 (5 días), se modelaron los objetos del sistema, detallando sus atributos

y comportamientos. Finalmente, en el Sprint 10 (5 días), se diseñó la interfaz gráfica del usuario, asegurando una experiencia óptima.

Durante la Fase de Implementación (85 días), el Sprint 11 (5 días) inició con la implementación de la base de datos. Luego, entre el Sprint 12 y el Sprint 18 (35 días), se desarrolló el backend, construyendo la lógica de negocio. Después, del Sprint 13 al Sprint 25 (35 días), se desarrolló el frontend para que los usuarios pudieran interactuar con el sistema. En el Sprint 14 (10 días), se implementaron pruebas automatizadas y manuales para garantizar la calidad.

Finalmente, en la Fase de Pruebas (26 días), se realizaron tres Sprints. El Sprint 15 (10 días) incluyó pruebas funcionales para asegurar que el sistema cumpliera con los requisitos. El Sprint 16 (10 días) abarcó pruebas no funcionales, evaluando rendimiento, seguridad y escalabilidad. El Sprint 17 (6 días) concluyó con las pruebas de aceptación, donde el cliente validó que el sistema satisficiera sus necesidades antes del lanzamiento.

4.2.2 TasksBoard y Burndown Chart

Se realizó un TasksBoard donde se registró los avances de cada Sprint donde se consideró tres estados de cada Sprint que son: Pendiente, en curso y hecho.

Se hizo un Burndown Chart donde se estimó los puntos de Historia para cumplir con las Historias de Usuario que fue 170 y se registró las medidas de avance del equipo de desarrollo.

En el Anexo 5 se muestra los cambios que tuvieron durante el avance del proyecto.

4.3 Desarrollo de la Metodología SUS

4.3.1 Introducción a la metodología SUS

El System Usability Scale (SUS) es una herramienta utilizada para medir la usabilidad de un sistema o producto, proporcionando una evaluación general de cómo los usuarios interactúan con él. Consiste en 10 preguntas que permiten recoger la percepción de los usuarios en relación con la facilidad de uso, la efectividad y la satisfacción al utilizar el sistema. El puntaje resultante ofrece una idea clara de si el sistema es usable o necesita mejoras.

En el contexto de este proyecto, el SUS se utilizará para evaluar la usabilidad del Sistema Informático para la Gestión de Historias clínicas, especialmente en su uso por parte de los docentes, el personal de responsabilidad social universitaria, y los estudiantes de medicina. Este análisis es fundamental para garantizar que el sistema cumpla con las expectativas de sus usuarios y se integre correctamente en las campañas médicas realizadas por la universidad.

4.3.2 Objetivo de la evaluación SUS

El objetivo principal es evaluar la usabilidad del sistema, midiendo aspectos clave como:

- Facilidad de uso por parte de los diferentes tipos de usuarios.
- Satisfacción general al interactuar con el sistema.
- Aprendizaje del sistema y adaptación a las tareas diarias.
- Confianza en el uso del sistema durante campañas médicas.

4.3.3 Procedimiento de implementación del SUS

Para evaluar la usabilidad del sistema, se aplicará el cuestionario del SUS a los tres grupos de usuarios principales:

- Docentes de medicina, odontología y enfermería.
- Personal de Responsabilidad Social Universitaria.
- Estudiantes de los últimos años participantes en las campañas.

El cuestionario consta de 10 afirmaciones que los usuarios evaluarán en una escala de 1 a 5, donde:

- 1 = Totalmente en desacuerdo.
- 5 = Totalmente de acuerdo.

Las afirmaciones incluyen preguntas como:

1. Me gustaría usar este sistema con frecuencia.
2. Encontré el sistema innecesariamente complejo.
3. Pensé que el sistema era fácil de usar.
4. Creo que necesitaría ayuda para usar el sistema.
5. Las funciones del sistema están bien integradas.
6. Me parece que el sistema tiene inconsistencias.
7. Creo que la mayoría de las personas aprenderían a usar este sistema rápidamente.
8. Encontré que el sistema era engorroso de usar.
9. Me sentí seguro usando el sistema.

10. Necesitaría aprender muchas cosas nuevas antes de poder utilizar el sistema.

4.3.4 Fases de la evaluación

a) Fase Inicial: Implementación Piloto

Durante las campañas médicas iniciales en Arequipa, se distribuirá el sistema a una pequeña muestra de usuarios, principalmente docentes y estudiantes, para que lo utilicen durante una jornada completa. Al final de la campaña, los participantes completarán el cuestionario SUS.

b) Fase de Expansión: Evaluación en Diversas Campañas

El cuestionario SUS será aplicado de manera continua en diferentes campañas para obtener una evaluación más representativa. Las retroalimentaciones de los usuarios se recopilarán y analizarán para realizar mejoras iterativas en el sistema.

4.3.5 Análisis de resultados del SUS

Cada cuestionario será evaluado para obtener un puntaje total. El puntaje final de SUS va de 0 a 100, donde:

- 85-100: Excelente usabilidad.
- 70-84: Buena usabilidad.
- 50-69: Usabilidad aceptable.
- Menos de 50: Problemas serios de usabilidad.

Los resultados ayudarán a identificar áreas específicas que requieren mejoras, asegurando que el sistema se ajuste adecuadamente a las necesidades de los usuarios en el contexto de las campañas médicas.

4.3.6 Conclusiones y mejoras propuestas

Cada A partir de los puntajes obtenidos, se proporcionarán recomendaciones de mejoras. En función de los riesgos identificados, como el uso de dispositivos móviles con especificaciones técnicas específicas y la posible resistencia de los usuarios, las mejoras se centrarán en:

- Optimización de la interfaz de usuario para dispositivos móviles.
- Simplificación de los flujos de trabajo.
- Capacitación adicional para asegurar una curva de aprendizaje más rápida.

El uso de la metodología SUS permitirá garantizar que el sistema ofrezca una experiencia eficiente y satisfactoria para todos los usuarios involucrados en la gestión de historias clínicas.

4.4 Documentación técnica

4.4.1 Plan de proyecto informativo

Se desarrolló un plan temporal del proyecto de desarrollo de un sistema de historias clínicas, de enero a noviembre, se estructuraría en varias fases clave: En enero, se realizaría la planificación y definición de requerimientos, recopilando necesidades de los usuarios y stakeholders. En febrero, se diseñaría la arquitectura del sistema, incluyendo base de datos y prototipos de la interfaz.

Durante marzo y abril, se desarrollaría el backend utilizando ASP.NET Core y el frontend con AngularJS, integrando los componentes principales del sistema. Entre mayo y junio, se realizarían pruebas iniciales del sistema y ajustes en la funcionalidad. Julio y agosto se dedicarían a la integración de los módulos, pruebas de seguridad y rendimiento, asegurando el manejo de grandes volúmenes de datos y el cumplimiento de normativas de privacidad. En septiembre y octubre se llevarían a cabo pruebas de usuario, validaciones finales y capacitación al personal médico y administrativo. Finalmente, en noviembre se realizaría las pruebas para su lanzamiento.

4.4.2 Especificación de requisitos del software

a. Viabilidad técnica

La presente tesis se llevó a cabo para el proyecto de un sistema de historias clínicas que implica evaluar si se cuenta con las herramientas, habilidades y tecnologías necesarias para llevarlo a cabo con éxito. Este proyecto es viable debido a que las tecnologías seleccionadas, como ASP.NET Core para el backend y AngularJS para el frontend, son altamente escalables, seguras y adecuadas para el desarrollo de aplicaciones web modernas. Ambas herramientas son compatibles con diversas plataformas, lo que permite flexibilidad en la implementación y despliegue en servidores locales o en la nube. Además, se dispone de una amplia comunidad de soporte y recursos, lo que facilita la solución de problemas y el aprendizaje de buenas prácticas. La infraestructura necesaria, como servidores y bases de datos (SQL o NoSQL), puede gestionarse eficazmente en un entorno seguro para proteger los datos sensibles de los pacientes. El equipo de

desarrollo necesita contar con experiencia en tecnologías .NET, Angular, gestión de bases de datos y protocolos de seguridad (como encriptación de datos y autenticación). Por lo tanto, con la infraestructura y el personal adecuado, el proyecto es técnicamente viable.

b. Viabilidad económica

El presupuesto presentado en la Tabla 1 ofrece una visión inicial de los costos asociados al proyecto.

c. Historias de usuario

Se hicieron en total 17 historias de usuario, a continuación, se irán mostrando estas historias por Sprint.

A continuación, podemos observar los Sprints con sus respectivas historias de usuario:

Tabla 3.
Historias de usuario y Sprints

Resumen	Estado	Persona asignada	Sprint	Fecha de ejecución	Fecha de finalización
Selección del lenguaje de programación	Terminado	Equipo Técnico	Sprint 1 (Planificación)	01/04/2024	05/04/2024
Definición de la plataforma de desarrollo	Terminado	Equipo Técnico	Sprint 2 (Planificación)	08/04/2024	15/04/2024
Establecimiento de técnicas y herramientas	Terminado	Equipo Técnico	Sprint 3 (Planificación)	16/04/2024	23/04/2024
Especificación de requisitos	Terminado	Cliente y Desarrollador	Sprint 4 (Planificación)	25/04/2024	03/05/2024
Definición del modelo del sistema web	Terminado	Desarrollador	Sprint 5 (Análisis)	06/05/2024	10/05/2024
Determinación de métodos y herramientas	Terminado	Desarrollador	Sprint 6 (Análisis)	13/05/2024	16/05/2024

Construcción de casos de uso	Terminado	Analista	Sprint 7 (Diseño)	17/05/2024	23/05/2024
Creación del diagrama de clases	Terminado	Arquitecto de Software	Sprint 8 (Diseño)	28/05/2024	06/06/2024
Modelado de objetos	Terminado	Arquitecto de Software	Sprint 9 (Diseño)	07/06/2024	13/06/2024
Diseño de la interfaz gráfica	Terminado	Diseñador	Sprint 10 (Diseño)	14/06/2024	20/06/2024
Implementación de la base de datos	Terminado	Desarrollador	Sprint 11 (Implementación)	21/06/2024	27/06/2024
Desarrollo del backend	Terminado	Desarrollador Backend	Sprint 12 (Implementación)	28/06/2024	15/08/2024
Desarrollo del frontend	Terminado	Desarrollador Frontend	Sprint 13 (Implementación)	16/08/2024	03/10/2024
Implementación de pruebas	Terminado	Equipo de Pruebas	Sprint 14 (Implementación)	04/10/2024	17/10/2024

Fuente: Elaboración Propia

A lo largo de los Sprints, el equipo técnico investigó y seleccionó el lenguaje de programación más adecuado, definió la plataforma y el entorno de desarrollo, y estableció metodologías o marcos de trabajo ágiles y herramientas clave. Posteriormente, se especificaron los requisitos funcionales y no funcionales del sistema, y se modeló el sistema web mediante diagramas de clases y casos de uso. En la fase de diseño, se crearon la interfaz gráfica y los objetos del sistema. Luego, el equipo implementó la base de datos y desarrolló tanto el backend como el frontend del sistema, asegurando una lógica robusta y una interfaz interactiva.

Las pruebas del sistema se han estado realizando antes de la entrega final del proyecto a los usuarios finales, el proyecto ya se encuentra levantado en Internet.

Tabla 4.

Pruebas realizadas por Sprints

Resumen	Estado	Persona asignada	Sprint	Fecha de ejecución	Fecha de finalización
Pruebas funcionales	En proceso	Tester	Sprint 15 (Pruebas)	18/10/2024	31/10/2024
Pruebas no funcionales	En proceso	Tester	Sprint 16 (Pruebas)	01/11/2024	14/11/2024
Pruebas de aceptación	En proceso	Cliente	Sprint 17 (Pruebas)	15/11/2024	22/11/2024

Fuente: Elaboración Propia

Para garantizar la calidad, se implementaron pruebas as funcionales y no funcionales, y el cliente validó el sistema a través de pruebas de aceptación antes del lanzamiento.

4.4.3 Especificación del diseño

A. Propósito

Este proyecto busca mejorar la gestión de historias clínicas en las campañas de responsabilidad social universitaria mediante la implementación de un Sistema Informático desarrollado con el framework SCRUM. El proyecto incluye un análisis del estado actual del proceso, la implementación del sistema con funcionalidades para el registro, consulta, generación de informes y análisis de datos, y la validación de la usabilidad del sistema con el instrumento System Usability Scale (SUS). El objetivo final es optimizar la gestión de la información de los pacientes, mejorar la eficiencia de la atención médica, garantizar la seguridad de los datos y facilitar la toma de decisiones en las campañas de responsabilidad social universitaria.

B. Alcance del proyecto

El Sistema Informático de Gestión de Historias clínicas está destinado para las campañas de responsabilidad social universitaria, incluyendo el diseño y desarrollo con funcionalidades para registro de datos, consulta de historiales, generación de informes y control de acceso. El proyecto impactará directamente en la gestión de las campañas, beneficiando a los profesionales de la salud y los pacientes, pero no incluye la compra de infraestructura, la capacitación formal de los usuarios ni la integración con sistemas externos.

C. Glosario

Para dar conocimiento de algunos términos utilizados para este proyecto de software se mostrará una tabla con la definición de algunos términos usados en esta tesis.

Tabla 5.
Glosario de términos

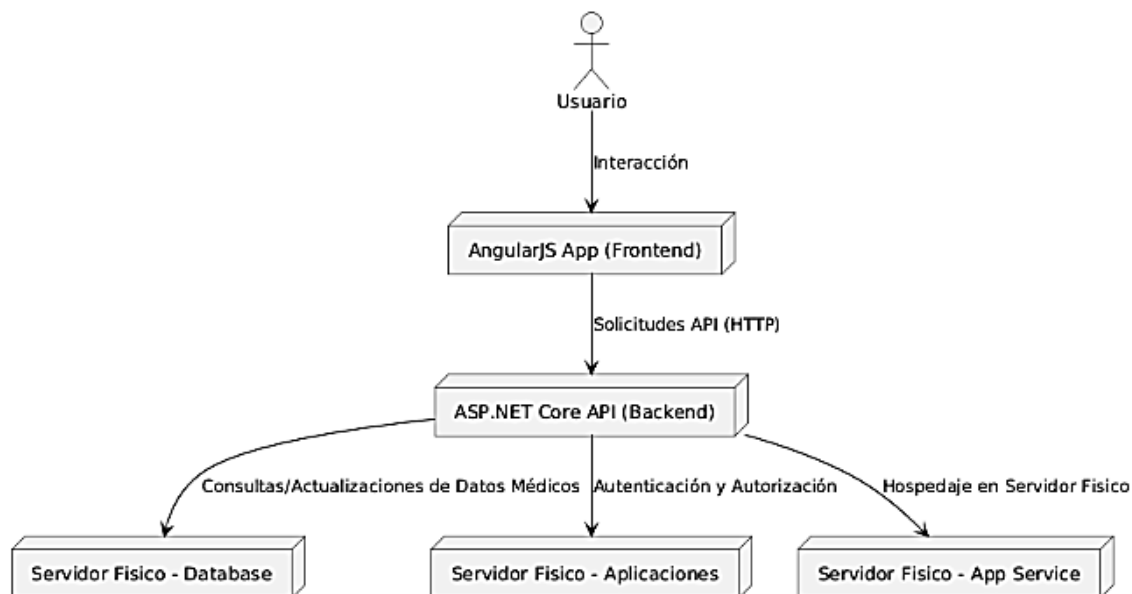
Término	Definición
Sistema de Historias clínicas	Software diseñado para gestionar de manera electrónica la información médica de los pacientes, facilitando el almacenamiento y acceso a sus datos médicos.
Campaña Médica	Actividades organizadas para ofrecer atención médica gratuita o accesible a comunidades, en las que se implementa el sistema de historias clínicas.
Sistema Informático	Conjunto de herramientas tecnológicas que permiten la automatización de procesos, en este caso, la gestión de historias clínicas durante las campañas médicas.
SCRUM	Framework de gestión de proyectos que se utiliza para desarrollar el sistema informático, basada en ciclos cortos de trabajo llamados "Sprints".
System Usability Scale (SUS)	Herramienta que se utiliza para medir la usabilidad del sistema a través de una escala del 1 al 100.
Angular	Framework de código abierto utilizado para crear interfaces de usuario dinámicas y modernas en aplicaciones web.
ASP.NET Core	Framework de desarrollo web multiplataforma que permite crear aplicaciones web robustas y escalables, incluyendo APIs REST.
Hosting	Servicio de alojamiento web donde se almacena el sistema informático para que pueda ser accesible desde internet.

Fuente: Elaboración Propia

El glosario de la Tabla 4 da mayor detalle de los términos utilizados en este trabajo.

D. Entorno del sistema

El sistema de historias clínicas está desarrollado con ASP.NET Core en el backend y Angular en el frontend, y se encuentra hospedado en un droplet de DigitalOcean, lo que permite un entorno flexible y controlado, con acceso remoto y administración a través de SSH. La base de datos médica se gestiona mediante SQL Server, también alojado en el mismo entorno, y los archivos adjuntos, como documentos clínicos, se almacenan en el servidor de archivos del sistema o en un volumen adicional configurado en el droplet. Para la autenticación de usuarios, se implementa un mecanismo seguro basado en hash SHA-256 para las contraseñas, cumpliendo con buenas prácticas de seguridad. El monitoreo y la supervisión del sistema se realizan mediante herramientas de código abierto y servicios adicionales configurados dentro del servidor. El proceso de integración y despliegue continuo (CI/CD) se gestiona a través de GitHub, facilitando actualizaciones seguras y automatizadas del sistema.

Figura 3.*Entorno del sistema de historias clínicas*

Fuente: Elaboración Propia

De esta forma se puede representar la interacción del usuario con los diferentes artefactos del sistema.

E. Especificación de requisitos funcionales

En esta sesión se describe lo que el personal de la pastelería solicitó realizar en el proyecto para cumplir con las expectativas del producto final.

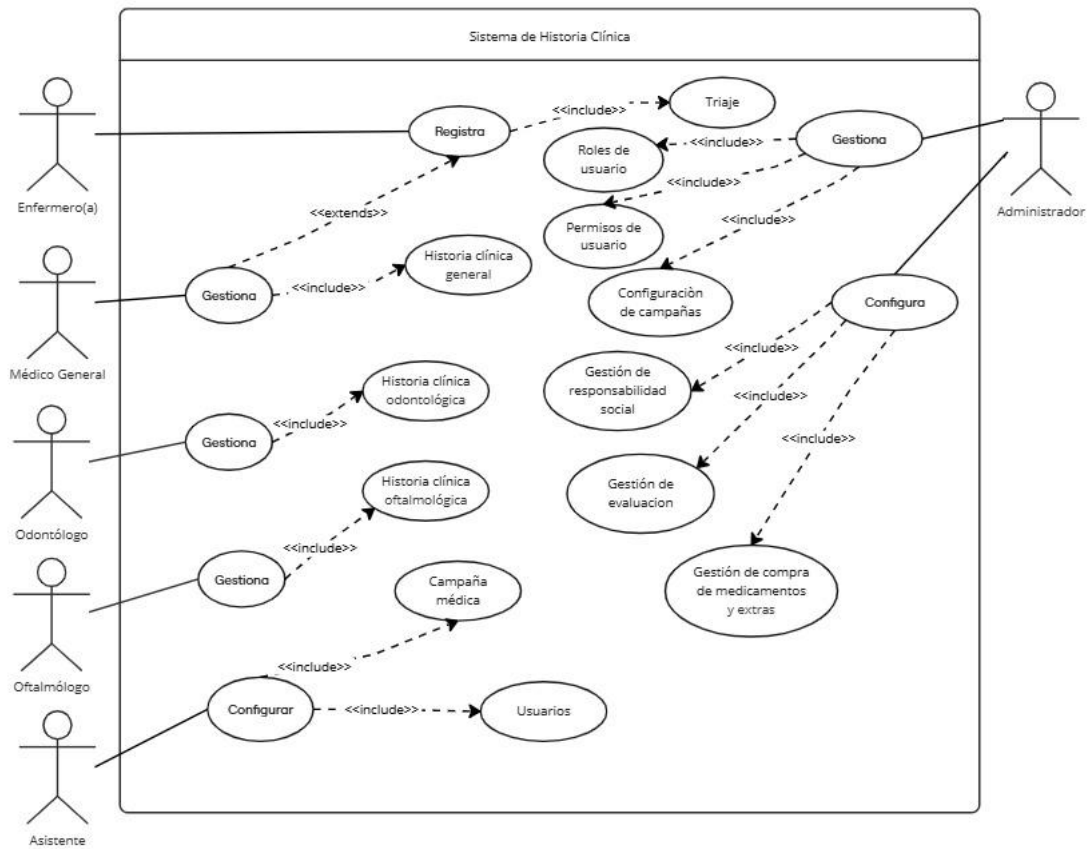
a. Casos de uso

En esta sesión se muestran los diagramas de caso de uso para describir cómo se activan las funcionalidades del Sistema de historia clínica.

El siguiente diagrama de casos uso está en la Figura 4.

Figura 4.

Diagrama de casos de uso



Fuente: Elaboración Propia

Los casos de uso fueron útiles para representar las diferentes funcionalidades y acciones que los usuarios pueden realizar en la plataforma del Sistema de historia clínica.

Los casos de uso fueron útiles para representar las diferentes funcionalidades y acciones que los usuarios pueden realizar en la plataforma del Sistema de historia clínica.

b. Requerimientos funcionales

1. El sistema RF01: El sistema debe permitir que el Administrador inicie sesión utilizando credenciales válidas.
2. RF02: El sistema debe permitir que el Administrador gestione los roles de usuario, incluyendo la creación, edición y

eliminación de roles.

3. RF03: El sistema debe permitir que el Administrador gestione los permisos de usuario, asignando y revocando permisos a los roles definidos.
4. RF04: El sistema debe permitir que el Administrador configure la gestión de responsabilidad social universitaria, incluyendo la definición de políticas y procesos.
5. RF05: El sistema debe permitir que el Administrador configure la gestión de evaluación, estableciendo criterios y métodos de evaluación.
6. RF06: El sistema debe permitir que el Administrador configure la gestión de la compra de medicamentos.
7. RF07: El sistema debe permitir que el Administrador gestione campañas, incluyendo la creación del paciente, seguimiento de la campaña y asignación del paciente.
8. RF08: El sistema debe permitir que el Administrador registre medicamentos en la base de datos, especificando nombre, tipo, y detalles relevantes.
9. RF09: El sistema debe permitir que el Administrador asigne medicamentos a campañas específicas, vinculando los medicamentos registrados con las campañas activas.
10. RF10: El sistema debe permitir que el Doctor registre triajes de pacientes, capturando datos del paciente y examen físico.
11. RF11: El sistema debe permitir que el Doctor registre la atención médica de un paciente, documentando el diagnóstico y

- procedimientos realizados.
12. RF12: El sistema debe permitir que el Doctor registre planes de tratamiento para pacientes, incluyendo medicación, antecedentes y diagnóstico.
 13. RF13: El sistema debe permitir que el Doctor registre la historia clínica de un paciente, manteniendo un registro completo de consultas, diagnósticos y tratamientos.
 14. RF14: El sistema debe permitir que el Doctor registre nuevos pacientes en el sistema, capturando información personal y médica relevante.
 15. RF15: El sistema debe permitir que el Doctor registre los datos de filiación de un paciente, como nombre, dirección, contacto, ocupación, estado civil y lugar de nacimiento.
 16. RF16: El sistema debe permitir que el Enfermero(a) registre el triaje de un paciente, documentando datos vitales, peso, saturación, temperatura y presión arterial.
 17. RF17: El sistema debe permitir que el Asistente registre nuevos usuarios en el sistema, asignándoles un rol y credenciales de acceso.
 18. RF18: El sistema debe verificar las credenciales de los usuarios al intentar iniciar sesión y permitir el acceso solo a usuarios autorizados.
 19. RF19: El sistema debe permitir que el Enfermero(a) genere una consulta a la especialidad que crea pertinente.
 20. RF20: El sistema debe importar datos de medicamentos desde

un archivo de Excel.

c. Requerimientos no funcionales

1. Seguridad y confiabilidad del sistema.
2. Facilidad de uso y navegación del sistema.
3. Adaptabilidad y escalabilidad del sistema para futuros requerimientos.
4. Respuesta rápida y performance en la respuesta a requerimientos de usuarios.

d. Requerimientos de interfaz externa o apariencia

1. Responsividad de diseño en las aplicaciones, debe tener adaptación a la pantalla de dispositivos como de equipos móviles y computador.
2. Los botones de las imágenes se deben diferenciar claramente.
3. La carga de datos de la aplicación debe ser dinámica, de modo que no requiera actualización de las pagina mientras carga datos.
4. El idioma de la plataforma debe ser en castellano.

e. Requerimientos de usabilidad

1. La aplicación debe reportar los mensajes de error y éxito, contar con la capacidad de mostrar información relacionada a la actividad de cada usuario.
2. El diseño de la aplicación debe ser intuitivo permitiendo a clientes y administradores diferenciar los objetos en las interacciones.
3. No es necesario que la información se acceda de manera directa

sino a través de un entorno que facilite la interacción de usuarios.

f. Requerimientos de rendimiento

1. La respuesta del sistema debe ser en un mínimo de tiempo en relación a los requerimientos de parte de los usuarios.
2. La aplicación debe ser capaz de visualizar la retroalimentación y en tiempos reducidos.

g. Requerimiento de soporte

1. El sistema dispondrá de un manual como guía.
2. El manejo de la aplicación debe estar bien descrito.

h. Requerimiento de seguridad

1. El acceso a la información debe ser restringido y protegido inaccesible para tercero.
2. Un usuario distinto no debe poder acceder a la información de un usuario individual.
3. Se empleará algoritmos de encriptación para las contraseñas dentro de las bases de datos.

i. Requerimientos políticos

1. El lenguaje empleado en la aplicación para los usuarios debe ser el apropiado.
2. No deberán mostrarse imágenes inapropiadas para los usuarios en la aplicación.

j. Requerimiento de confiabilidad

1. La aplicación debe mostrar tolerancia a fallos.
2. El sistema deberá evitar fallos en caso de ocurrencia de

cualquier error durante el acceso a la información.

k. Documentación en línea y ayuda

1. La aplicación permitirá brindar descripción de ayuda val usuario.
2. El manual de ayuda estará disponible para que los usuarios puedan acceder a él.

l. Requerimiento de Software

1. Será posible ejecutar la aplicación desde cualquier plataforma web.
2. Para la gestión de información el sistema será capaz de conectarse a la base de datos.

m. Requerimiento de Hardware

1. Se podrá alojar la aplicación en servidores locales pertenecientes a la universidad.
2. Se podrá manejar la aplicación desde cualquier dispositivo con acceso a internet.

n. Diccionario de datos de las entidades

Se ha optado en hacer una documentación a las entidades que el formato es de diccionario de datos ya que la aplicación es de gran escala.

Tabla 6.

Diccionario de datos – Tablas principales

Entidad	Atributos	Relaciones
PACIENTE	Nombres, Apellidos, N° Documento, Fecha Nacimiento, Teléfono, Sexo, Departamento, Distrito, País, Ocupación, EstadoCivil	HISTORIA_CLÍNICA, CONSULTA
PERSONAL	Nombres, Apellidos, N° Colegiatura, Especialidad, Usuario	CONSULTA, DIAGNÓSTICO
HISTORIA_CLÍNICA	Código único, Fecha creación, Tipo	PACIENTE, CONSULTA
CONSULTA	Fecha cita, Motivo, Especialidad, Estado	PACIENTE, PERSONAL_MÉDICO
EXAMEN_VISUAL	Agudeza SC, Agudeza CC, Distancia, Ojo evaluado	HISTORIA_CLÍNICA
FONDO_OJO	Descripción retina, Descripción nervio óptico	HISTORIA_CLÍNICA
AGUDEJA_VISUAL	Distancia, ECC, CC	HISTORIA_CLÍNICA
REFRACCIÓN	ESF, CIL, EJE, DIP	HISTORIA_CLÍNICA
COLOR	Detalle, Hexadecimal	
DIAGNÓSTICO	Enfermedad, Comentarios, Fecha	HISTORIA_CLÍNICA, PERSONAL_MÉDICO
TRATAMIENTO	Medicamento, Dosis, Frecuencia, Duración	HISTORIA_CLÍNICA
CAMPANA_SALUD	Nombre, Fecha, Lugar, Responsable	PACIENTE, MEDICAMENTO
MEDICAMENTO	Nombre, Presentación, Cantidad	CAMPANA_SALUD, TRATAMIENTO
ODONTOGRAMA	Diente, Sección, Código tratamiento	HISTORIA_CLÍNICA
DIENTE	Número, Nombre, Cuadrante	ODONTOGRAMA
OJO	Nombre, Abreviatura, Zona Externa, Zona Interna	PACIENTE
ANTECEDENTE_OCULAR	Tipo, Descripción, Fecha	HISTORIA_CLÍNICA
ANTECEDENTE_GENERAL	Tipo, Descripción, Fecha	HISTORIA_CLÍNICA
EXAMEN_FÍSICO	Presión arterial, Peso, Talla, IMC	HISTORIA_CLÍNICA
TRIAJE	Prioridad, Signos vitales, Observaciones	PACIENTE
DOCUMENTO	Tipo, Número, Fecha emisión	PACIENTE, PERSONAL_MÉDICO
ESPECIALIDAD_MÉDICA	Nombre, Área	PERSONAL_MÉDICO

MOTIVO_CONSULTA	Descripción, Prioridad	CONSULTA
OBSERVACIÓN_OFTA	Descripción, Recomendaciones	HISTORIA_CLÍNICA
EXAMEN_AUXILIAR	Tipo, Resultados, Imágenes	HISTORIA_CLÍNICA
PROCEDIMIENTO	Nombre, Descripción, Fecha	HISTORIA_CLÍNICA
ANAMNESIS	Valor	HISTORIA_CLÍNICA, PERSONAL
BIOMICROSCOPIA	Zona Externa, Es Normal	OJO

Fuente: Elaboración Propia

Esta tabla resume 29 entidades clave del sistema de historias clínicas. Incluye información de pacientes (nombres, documento, teléfono), personal médico (colegiatura, especialidad), procesos médicos (consultas, exámenes visuales, diagnósticos) y soporte (medicamentos, ubicación geográfica) como entidades principales. Las relaciones muestran cómo interactúan estos elementos, como pacientes vinculados a sus historias clínicas y consultas, o personal médico asociado a diagnósticos y tratamientos. El formato usa nombres descriptivos en español y mayúsculas para claridad, eliminando identificadores numéricos para centrarse en los datos relevantes para el usuario final.

4.4.4 Interfaz de usuario

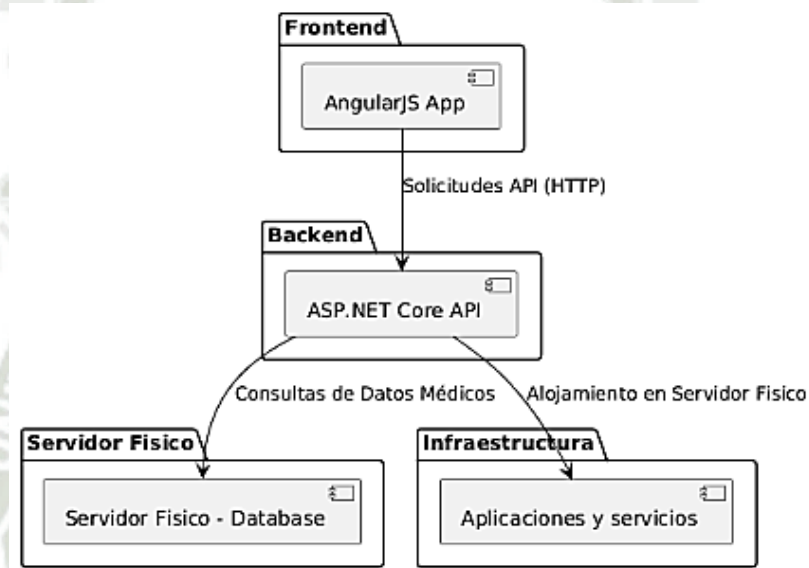
A. Vista lógica del sistema

El diagrama ilustra la arquitectura de una aplicación web que utiliza una arquitectura de microservicios. La interfaz de usuario (frontend) está construida con AngularJS, una popular framework de JavaScript, y se comunica con un backend desarrollado en ASP.NET Core. Este backend se encarga de procesar las solicitudes del frontend, realizar

consultas a una base de datos alojada en un servidor físico y gestionar la interacción con otros servicios o aplicaciones que podrían ser necesarios para la funcionalidad de la aplicación. En la Figura 6 vemos la vista lógica del sistema

Figura 5.

Vista Lógica del sistema



Fuente: Elaboración Propia

El sistema de información trabaja con dos tecnologías de desarrollo que deben comunicarse e integrarse adecuadamente para su correcto funcionamiento en la ejecución del software.

B. Vista física del sistema

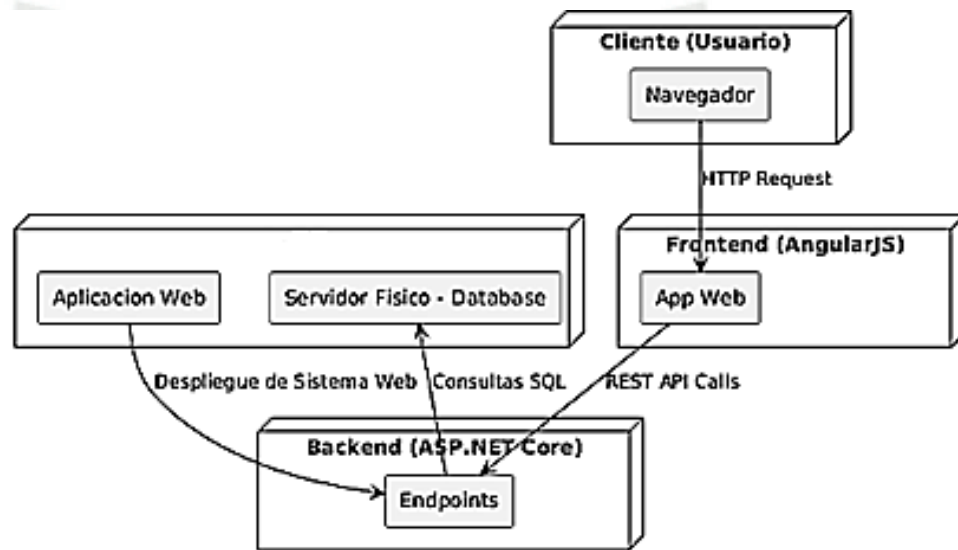
El diagrama de la Figura 7 presenta una arquitectura de aplicación web típica, separada en un frontend (interfaz de usuario) desarrollado con AngularJS y un backend (lógica de negocio) construido con ASP.NET Core. El frontend interactúa con el backend a través de una API REST para solicitar y recibir datos. La información persistente se almacena en una base

de datos relacional, a la que el backend accede mediante consultas SQL.

La aplicación y todo su ecosistema está alojado en un servidor físico como se muestra en la Figura 8.

Figura 6.

Vista Física del sistema



Fuente: Elaboración Propia

El diagrama utiliza símbolos estándar para representar los componentes del sistema. El diagrama está dividido en tres secciones principales: Capa de presentación, Capa de negocio y Capa de datos. El diagrama incluye algunas flechas que muestran cómo se comunican los componentes del sistema. El diagrama incluye algunas etiquetas que describen los componentes del sistema.

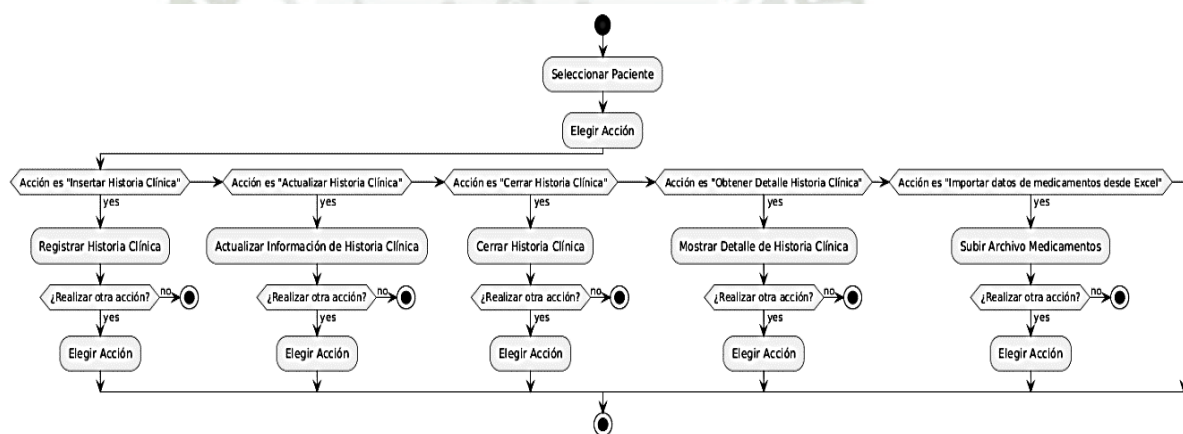
C. Vista de procesos del sistema

El diagrama de actividad ilustra un proceso de gestión de historias clínicas. Inicia con la selección de un paciente específico y bifurca en diversas acciones posibles sobre su historial clínico. Estas acciones pueden

incluir el registro de una nueva historia, la actualización de datos existentes, el cierre de la historia o la consulta detallada de la misma. Además, permite adjuntar documentos relevantes, como informes de laboratorio o recetas médicas. El proceso es iterativo, permitiendo al usuario realizar múltiples acciones sobre un mismo paciente. Cada acción desencadena una decisión que determina si se continúa con otra acción o se finaliza el proceso. Este diagrama representa una herramienta visual útil para comprender y optimizar los flujos de trabajo en sistemas de gestión de información médica.

Figura 7.

Diagrama de actividad del sistema de historias clínicas



Fuente: Elaboración Propia

El diagrama de actividad permite comprender los pasos involucrados y la secuencia en la que deben realizarse.

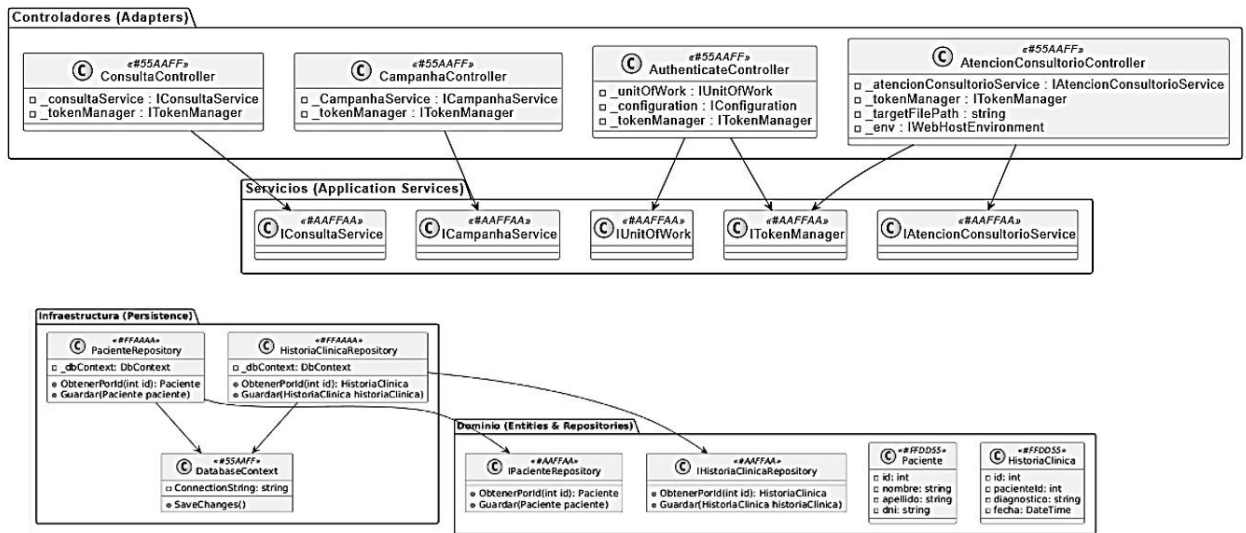
D. Diseño de la estructura del Sistema

El diagrama de clases representa al sistema de gestión de salud para una comunidad universitaria, con controladores para diferentes áreas como autenticación, atención de consultas, campañas, medicamentos, especialidades médicas, pacientes y resultados. Cada controlador se conecta

a un servicio específico, gestionando datos como historiales médicos, consultas, tratamientos, campañas, medicamentos y resultados.

Figura 8.

Diagrama de clases del sistema



Fuente: Elaboración Propia

El diagrama de clases representa al sistema informático para la gestión de historias clínicas, estructurado en capas siguiendo una arquitectura limpia. En la capa de controladores (Adapters), se encuentran ConsultaController, CampanhaController, AuthenticateController y AtencionConsultorioController, los cuales dependen de servicios como IConsultaService, ICampanhaService e IAtencionConsultorioService, además de un TokenManager y otros componentes. La capa de servicios (Application Services) define interfaces para la lógica de negocio, incluyendo IConsultaService, ICampanhaService, IAtencionConsultorioService e IUnitOfWork. En la infraestructura (Persistence), los repositorios PacienteRepository e HistoriaClinicaRepository gestionan el acceso a la base de datos mediante DbContext, mientras que DatabaseContext administra la

conexión. Finalmente, en la capa de dominio (Entities & Repositories), las entidades Paciente e HistoriaClinica modelan los datos clave del sistema con atributos como id, nombre, apellido, DNI, diagnóstico y fecha, estableciendo la estructura fundamental para el almacenamiento y gestión de historias clínicas.

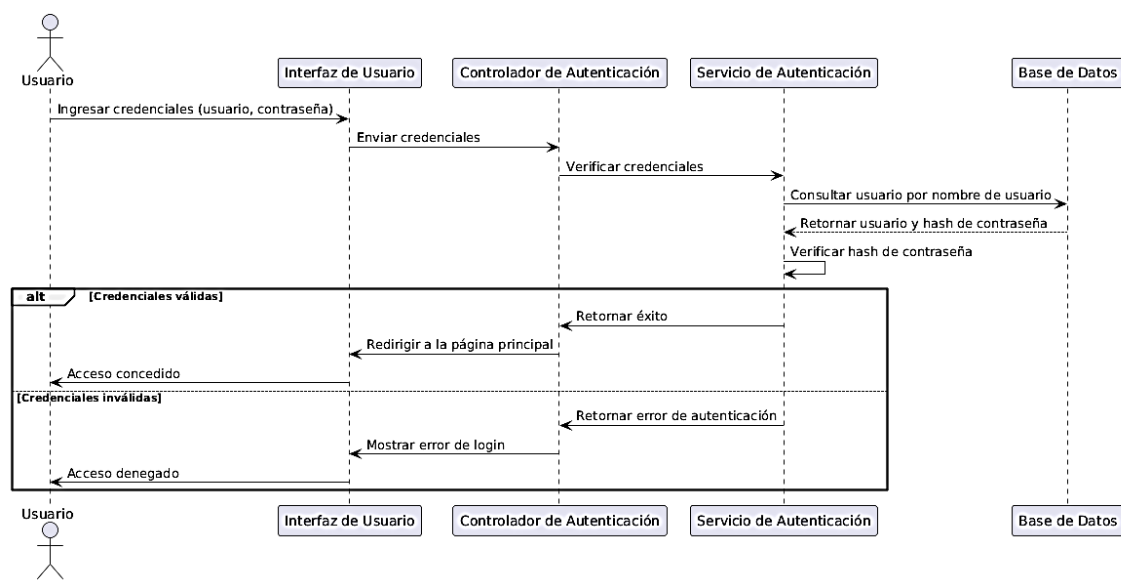
E. Comportamiento del sistema

En esta sección se describe el comportamiento del Sistema de historia clínica mediante el diseño de diagramas de secuencia.

Primero tenemos el diagrama de secuencia del login. El diagrama de secuencia ilustra el proceso de autenticación de un usuario en un sistema. Inicia cuando el usuario ingresa sus credenciales (nombre de usuario y contraseña) en la interfaz de usuario. A continuación, la interfaz envía estas credenciales al controlador de autenticación, quien a su vez las reenvía al servicio de autenticación. Este último consulta la base de datos para obtener la información del usuario y su contraseña almacenada (en forma de hash). Si las credenciales proporcionadas coinciden con las almacenadas, se devuelve un mensaje de éxito y el usuario es redirigido a la página principal. En caso contrario, se retorna un mensaje de error y se muestra al usuario un mensaje indicando que las credenciales son inválidas.

Figura 9.

Diagrama de secuencia de iniciar sesión

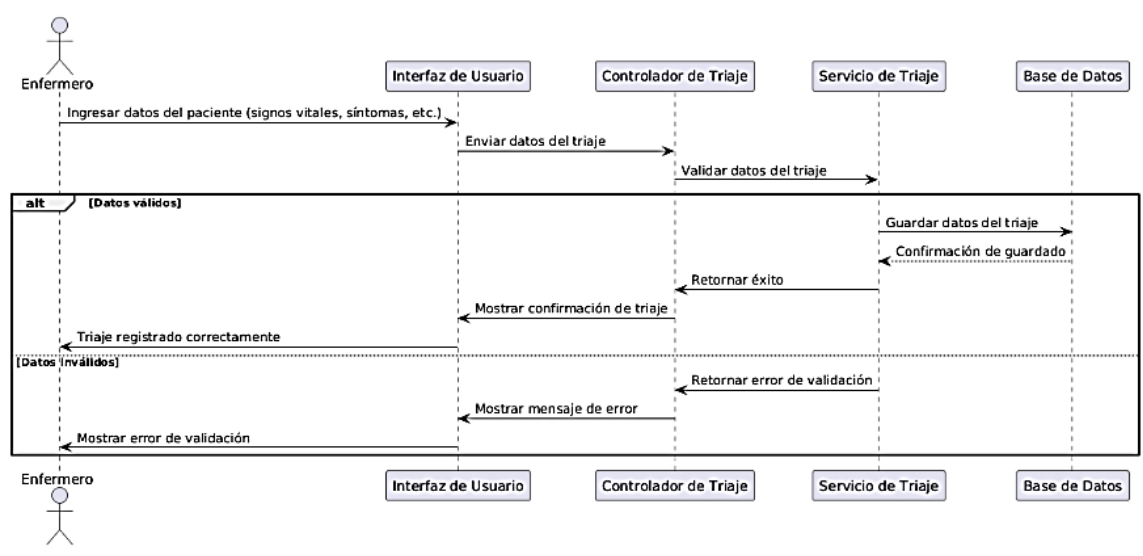


Fuente: Elaboración Propia

Luego el diagrama de secuencia de gestionar triaje. El diagrama de secuencia ilustra el proceso de registro de un triaje en un sistema hospitalario. Comienza cuando un enfermero ingresa los datos del paciente (signos vitales, síntomas, etc.) en la interfaz de usuario. Estos datos son enviados al controlador de triaje, quien los valida. Si los datos son correctos, el controlador solicita al servicio de triaje que los guarde en la base de datos. Una vez guardados, se retorna un mensaje de éxito al controlador y se muestra al enfermero una confirmación de que el triaje ha sido registrado correctamente. Sin embargo, si los datos son inválidos, se retorna un mensaje de error al controlador y se muestra al enfermero un mensaje indicando que los datos deben ser corregidos.

Figura 10.

Diagrama de secuencia de gestionar triaje

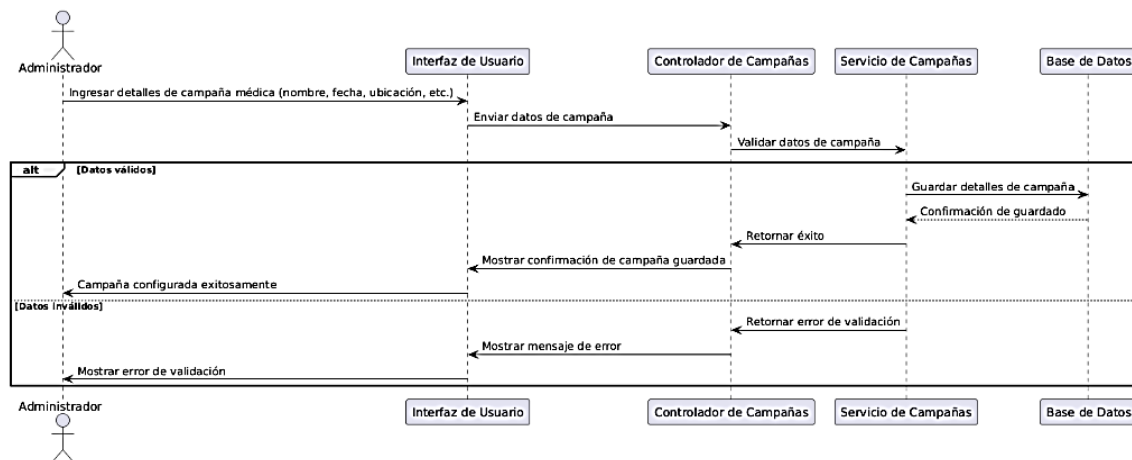


Fuente: Elaboración Propia

Luego el diagrama de secuencia de configuración de campañas médicas. Este diagrama de secuencia detalla el flujo de interacciones entre los diferentes componentes del sistema durante el proceso de registro de una nueva campaña médica dentro de la plataforma de gestión de historias clínicas. El proceso inicia cuando un administrador ingresa los detalles de la campaña (nombre, fecha, ubicación, etc.) a través de la interfaz de usuario. Estos datos son enviados al controlador de campañas, el cual se encarga de validar la información ingresada. Si los datos son correctos, el controlador solicita al servicio de campañas que guarde la información en la base de datos. Una vez almacenada, se retorna un mensaje de confirmación al controlador, el cual a su vez lo muestra al administrador a través de la interfaz de usuario. En caso de que los datos sean inválidos, se genera un mensaje de error y se notifica al administrador para que corrija la información.

Figura 11.

Diagrama de secuencia de configuración de campañas médicas

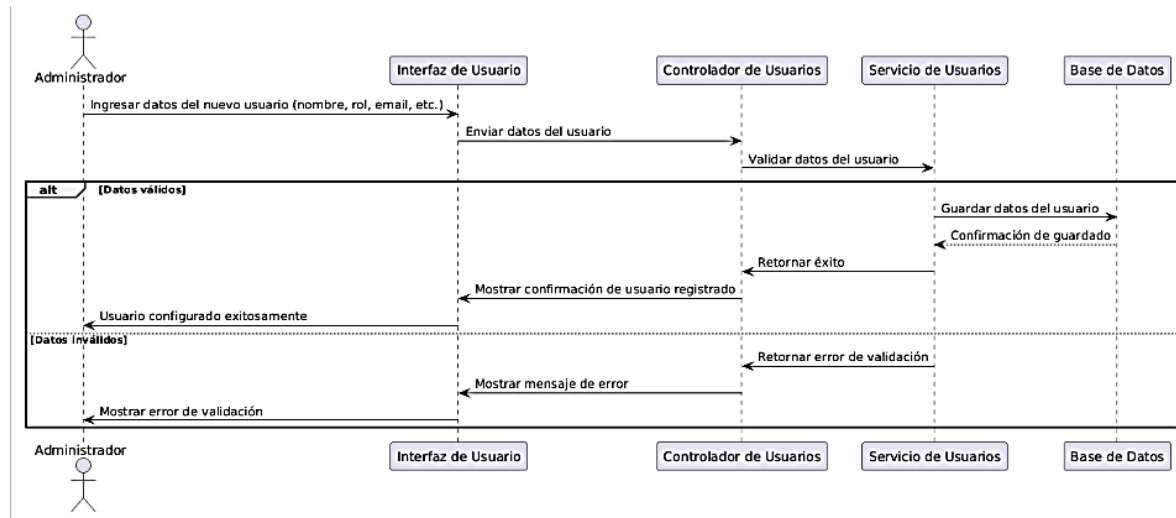


Fuente: Elaboración Propia

Luego el diagrama de configuración de usuarios. Este diagrama de secuencia ilustra el flujo de interacciones que ocurren cuando un administrador registra un nuevo usuario en el sistema. Un administrador ingresa los datos del nuevo usuario (nombre, rol, correo electrónico, etc.) a través de la interfaz de usuario. Estos datos son enviados al controlador de usuarios, quien se encarga de validar la información proporcionada. Si los datos son correctos, el controlador solicita al servicio de usuarios que guarde los datos en la base de datos. Una vez almacenados, se retorna un mensaje de confirmación al controlador, el cual a su vez lo muestra al administrador a través de la interfaz de usuario. En caso de que los datos sean inválidos, se genera un mensaje de error y se notifica al administrador para que corrija la información.

Figura 12.

Diagrama de configuración de usuarios

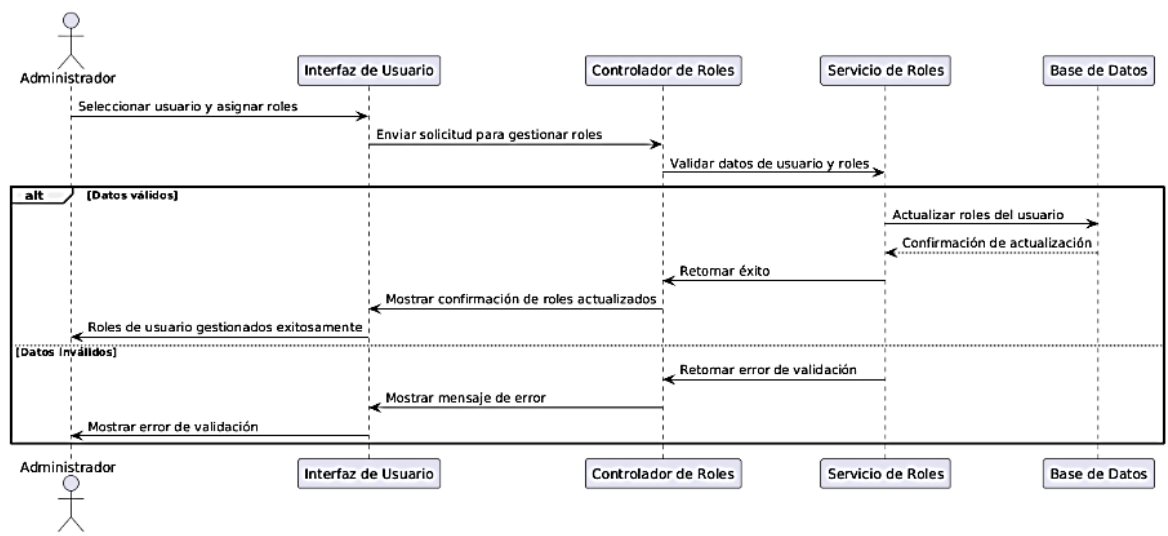


Fuente: Elaboración Propia

Luego el diagrama de secuencia de gestionar roles de usuario. Este diagrama de secuencia detalla el proceso de asignación y modificación de roles a los usuarios del sistema. Un administrador selecciona a un usuario y los roles que desea asignarle a través de la interfaz de usuario. Esta solicitud es enviada al controlador de roles, quien valida tanto los datos del usuario como los roles seleccionados. Si la información es correcta, el controlador solicita al servicio de roles que actualice los roles del usuario en la base de datos. Una vez realizada la actualización, se retorna un mensaje de confirmación al controlador, el cual a su vez lo muestra al administrador a través de la interfaz de usuario. En caso de que haya algún error en los datos, se genera un mensaje de error y se notifica al administrador para que corrija la información.

Figura 13.

Diagrama de secuencia de gestionar roles de usuario



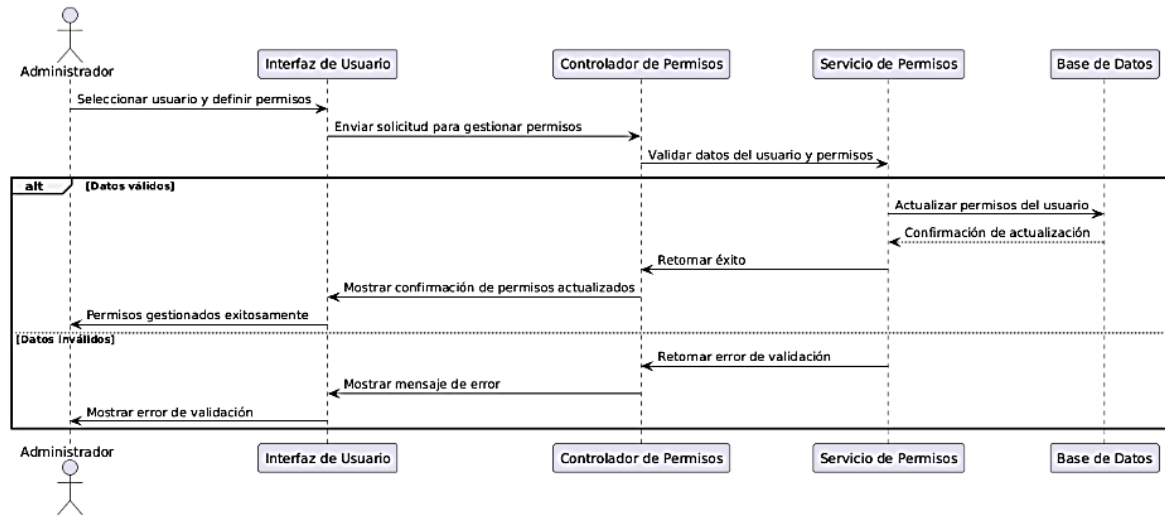
Fuente: Elaboración Propia

Luego el diagrama de secuencia de gestionar permisos de usuario.

Este diagrama de secuencia detalla el proceso de asignación y modificación de permisos a los usuarios del sistema. Un administrador selecciona a un usuario y los permisos que desea asignarle a través de la interfaz de usuario. Esta solicitud es enviada al controlador de permisos, quien valida tanto los datos del usuario como los permisos seleccionados. Si la información es correcta, el controlador solicita al servicio de permisos que actualice los permisos del usuario en la base de datos. Una vez realizada la actualización, se retorna un mensaje de confirmación al controlador, el cual a su vez lo muestra al administrador a través de la interfaz de usuario. En caso de que haya algún error en los datos, se genera un mensaje de error y se notifica al administrador para que corrija la información.

Figura 14.

Diagrama de secuencia de gestionar permisos de usuario

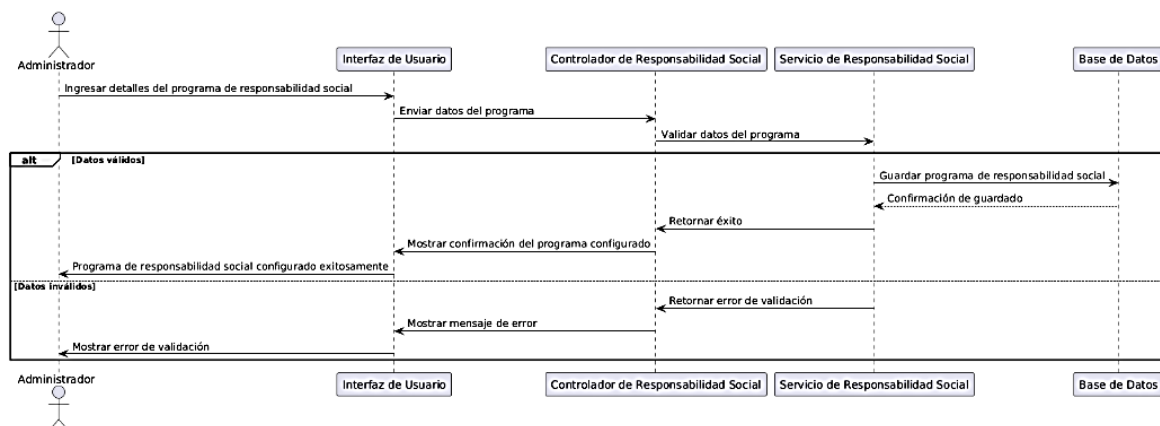


Fuente: Elaboración Propia

Luego el diagrama de secuencia de configurar gestión de responsabilidad social universitaria. Este diagrama de secuencia detalla el flujo de interacciones que ocurren cuando un administrador registra un nuevo programa de responsabilidad social universitaria dentro del sistema. Un administrador ingresa los detalles del programa (nombre, objetivos, fechas, etc.) a través de la interfaz de usuario. Estos datos son enviados al controlador de responsabilidad social universitaria, quien se encarga de validar la información proporcionada. Si los datos son correctos, el controlador solicita al servicio de responsabilidad social universitaria que guarde los datos del programa en la base de datos. Una vez almacenados, se retorna un mensaje de confirmación al controlador, el cual a su vez lo muestra al administrador a través de la interfaz de usuario. En caso de que haya algún error en los datos, se genera un mensaje de error y se notifica al administrador para que corrija la información.

Figura 15.

Diagrama de secuencia de configurar gestión de responsabilidad social universitaria

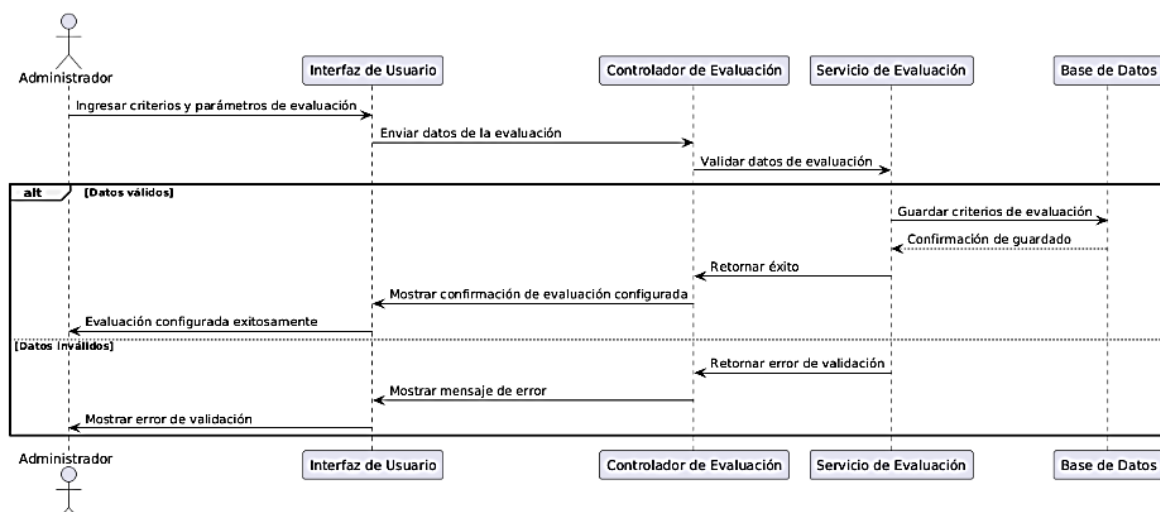


Fuente: Elaboración Propia

Luego el diagrama de secuencia de configurar gestión de evaluación. Este diagrama de secuencia detalla el proceso de configuración de los criterios y parámetros que se utilizarán para evaluar los resultados de las campañas de responsabilidad social universitaria. Un administrador ingresa los criterios específicos y parámetros de evaluación a través de la interfaz de usuario. Estos datos son enviados al controlador de evaluación, quien valida la información proporcionada. Si los datos son correctos, el controlador solicita al servicio de evaluación que guarde los criterios en la base de datos. Una vez almacenados, se retorna un mensaje de confirmación al controlador, el cual a su vez lo muestra al administrador a través de la interfaz de usuario. En caso de que haya algún error en los datos, se genera un mensaje de error y se notifica al administrador para que corrija la información.

Figura 16.

Diagrama de secuencia de configurar gestión de evaluación

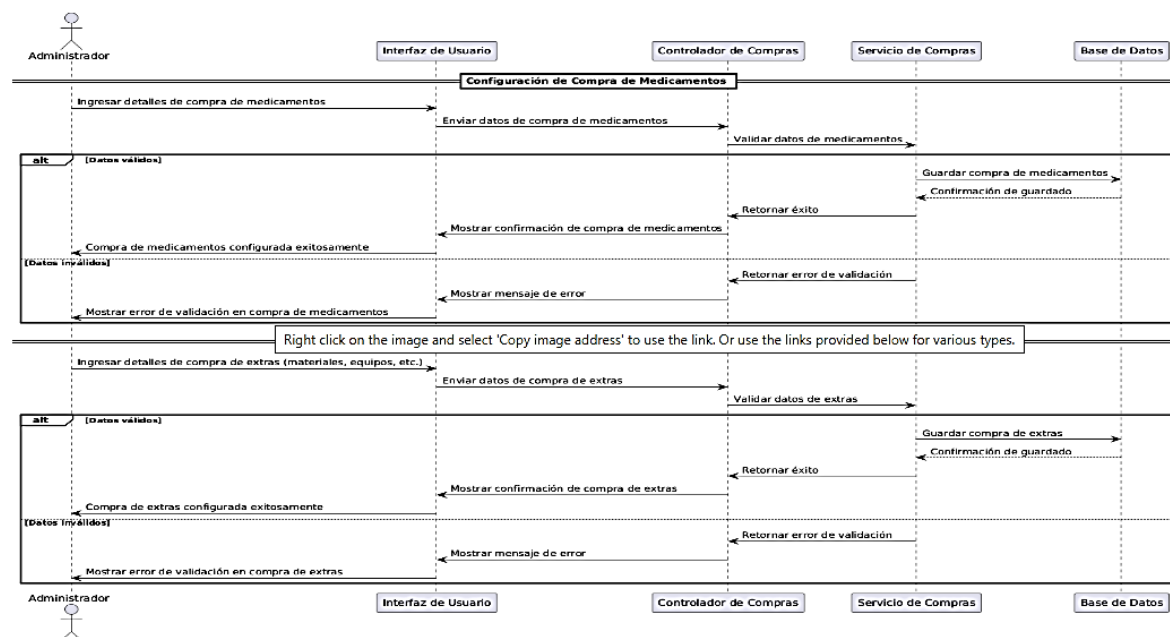


Fuente: Elaboración Propia

Luego el diagrama de secuencia de configurar gestión de compra de medicamentos y extras. Este diagrama de secuencia ilustra el proceso de configuración de compras de medicamentos y otros extras necesarios para llevar a cabo las campañas de responsabilidad social universitaria. Un administrador ingresa los detalles de la compra de medicamentos (cantidad, tipo, proveedor, etc.) a través de la interfaz de usuario. Estos datos son enviados al controlador de compras, quien valida la información proporcionada. Si los datos son correctos, el controlador solicita al servicio de compras que guarde la información de la compra en la base de datos. Una vez almacenada, se retorna un mensaje de confirmación al controlador, el cual a su vez lo muestra al administrador a través de la interfaz de usuario. Se repite un proceso similar para la configuración de la compra de extras (materiales, equipos, etc.). En caso de que haya algún error en los datos, se genera un mensaje de error y se notifica al administrador para que corrija la información.

Figura 17.

Diagrama de secuencia de configurar gestión de compra de medicamentos y extras

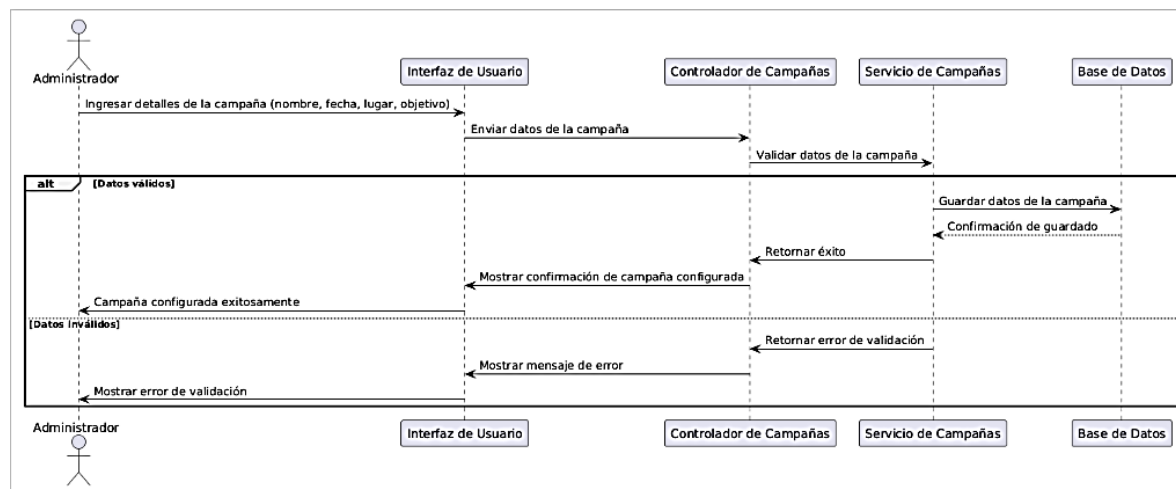


Fuente: Elaboración Propia

Luego el diagrama de secuencia de configuraciones de campañas. Este diagrama de secuencia detalla el proceso de creación de una nueva campaña de responsabilidad social universitaria dentro del sistema. Un administrador ingresa los detalles de la campaña (nombre, fecha, lugar, objetivo, etc.) a través de la interfaz de usuario. Estos datos son enviados al controlador de campañas, quien se encarga de validar la información proporcionada. Si los datos son correctos, el controlador solicita al servicio de campañas que guarde los datos de la campaña en la base de datos. Una vez almacenados, se retorna un mensaje de confirmación al controlador, el cual a su vez lo muestra al administrador a través de la interfaz de usuario. En caso de que haya algún error en los datos, se genera un mensaje de error y se notifica al administrador para que corrija la información.

Figura 18.

Diagrama de secuencia de configuraciones de campañas

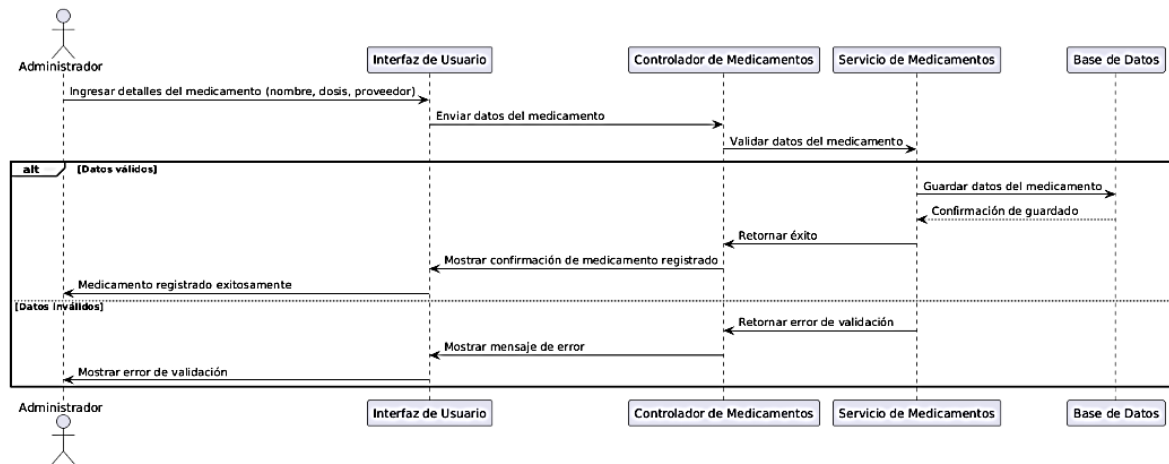


Fuente: Elaboración Propia

Luego el diagrama de secuencia de registrar medicamento. Este diagrama de secuencia detalla el proceso de registro de un nuevo medicamento dentro del sistema para su uso en las campañas de responsabilidad social universitaria. Un administrador ingresa los detalles del medicamento (nombre, dosis, proveedor, etc.) a través de la interfaz de usuario. Estos datos son enviados al controlador de medicamentos, quien se encarga de validar la información proporcionada. Si los datos son correctos, el controlador solicita al servicio de medicamentos que guarde los datos del medicamento en la base de datos. Una vez almacenados, se retorna un mensaje de confirmación al controlador, el cual a su vez lo muestra al administrador a través de la interfaz de usuario. En caso de que haya algún error en los datos, se genera un mensaje de error y se notifica al administrador para que corrija la información.

Figura 19.

Diagrama de secuencia de registrar medicamento

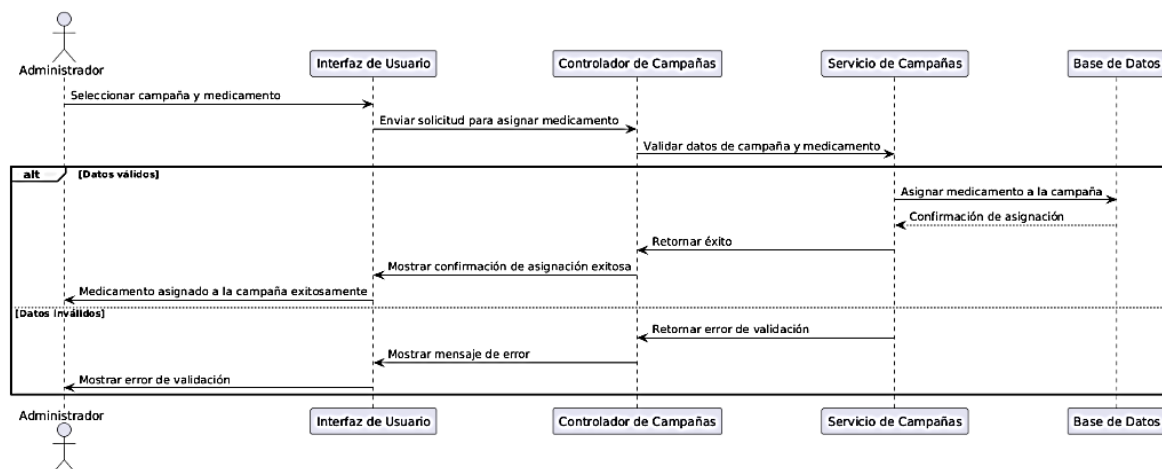


Fuente: Elaboración Propia

Luego el diagrama de secuencia de asignar medicamento a campaña. Este diagrama de secuencia detalla el proceso de asignar un medicamento específico a una campaña de responsabilidad social universitaria ya existente dentro del sistema. Un administrador selecciona una campaña y el medicamento que desea asignarle a través de la interfaz de usuario. Esta solicitud es enviada al controlador de campañas, quien valida tanto los datos de la campaña como los del medicamento. Si la información es correcta, el controlador solicita al servicio de campañas que asigne el medicamento a la campaña en la base de datos. Una vez realizada la asignación, se retorna un mensaje de confirmación al controlador, el cual a su vez lo muestra al administrador a través de la interfaz de usuario. En caso de que haya algún error en los datos, se genera un mensaje de error y se notifica al administrador para que corrija la información.

Figura 20.

Diagrama de secuencia de asignar medicamento a campaña



Fuente: Elaboración Propia

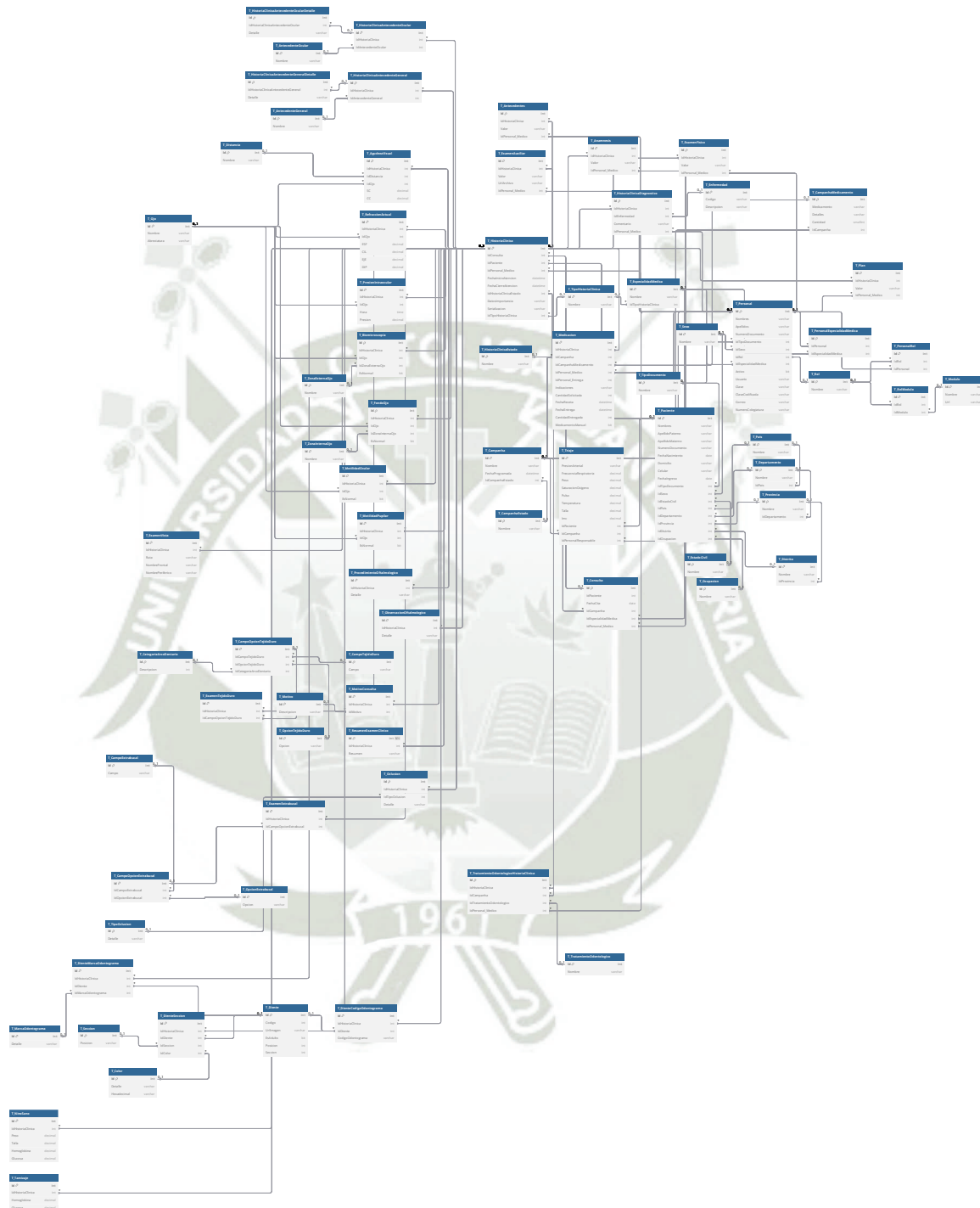
Estos diagramas nos han permitido visualizar de manera clara y concisa las interacciones entre los diferentes componentes del sistema, desde la interfaz de usuario hasta la base de datos.

F. Diseño de datos del sistema

En la Figura 22 se tiene el modelo relacional del sistema.

Figura 21.

Modelo relacional del sistema



Fuente: Elaboración Propia

Se cubrió esta sección de especificación de diseño.

4.4.5 Documentación técnica de programación

A. Hardware

Antes de abordar la parte de programación, es importante detallar los principales componentes de hardware utilizados en el desarrollo del sistema de historia clínica.

a) Laptop

- Procesador:
- Marca y modelo: Intel® Core™ i7-1065G7 CPU @ 1.30GHz – 1.50GHz
- Arquitectura: 64 bits
- Número de núcleos: 4
- Memoria RAM: Capacidad instalada: 12.0 GB (11.8 GB utilizable)
- Sistema Operativo: Windows 10 de 64 bits

c) Servidor en la nube

Para la implementación del sistema, se utiliza un servidor en la nube con las siguientes características:

- Proveedor: DigitalOcean
- Sistema Operativo: Ubuntu 24.10 x64
- Especificaciones del servidor:
- Memoria RAM: 2GB
- CPU: 1 vCPU

Almacenamiento: 60GB SSD

Transferencia de datos: 3TB

B. Frameworks

El desarrollo del sistema de historia clínica se basa en una arquitectura moderna utilizando ASP.Net Core y Angular, tecnologías complementarias que garantizan escalabilidad, seguridad y facilidad de mantenimiento.

ASP.Net Core es un framework de código abierto utilizado para la lógica del lado del servidor, encargado de la gestión de la base de datos, autenticación de usuarios y seguridad de la información médica. Su arquitectura ligera y eficiente permite tiempos de respuesta óptimos, fundamentales en un sistema que maneja datos sensibles y debe ser accesible para múltiples usuarios simultáneamente. (Zang, 2023)

Por otro lado, Angular es un framework de JavaScript diseñado para el desarrollo de aplicaciones web de una sola página (SPA). Su capacidad para construir interfaces dinámicas e interactivas, con una estructura modular y reutilizable, facilita la gestión de historias clínicas de manera eficiente y segura.

La combinación de ASP.Net Core y Angular permite el desarrollo de un sistema robusto que ofrece un equilibrio adecuado entre funcionalidad, seguridad y facilidad de uso. ASP.Net Core maneja el procesamiento y almacenamiento de datos en el servidor, mientras que Angular proporciona una experiencia fluida para el usuario, mejorando la interacción y usabilidad. Esta integración optimiza los tiempos de carga y garantiza la seguridad en el manejo de la información médica, cumpliendo con los estándares de desarrollo modernos.

C. Entorno de desarrollo

El desarrollo del sistema se llevó a cabo utilizando herramientas y tecnologías que optimizan la eficiencia del código y garantizan una implementación estructurada.

Para la construcción del backend, se utilizó Visual Studio Community, un entorno de desarrollo integrado (IDE) gratuito y completo que permite la implementación de aplicaciones en ASP.Net Core de manera eficiente. (Zang, 2023)

La interfaz de usuario fue desarrollada con Angular CLI, una herramienta de línea de comandos que facilita la creación y administración de proyectos Angular, optimizando el flujo de trabajo en la gestión del frontend.

El lenguaje de programación elegido para el frontend fue TypeScript, debido a su tipado estático y su capacidad de mejorar la legibilidad, mantenibilidad y seguridad del código.

En cuanto a la gestión del código fuente y control de versiones, se utilizó Git, lo que permitió un seguimiento preciso de los cambios y una colaboración eficiente entre los desarrolladores. (Blog IfGeekThen, s/f)

La combinación de estas herramientas y frameworks permitió el desarrollo de un sistema de historias clínicas eficiente, escalable y seguro, asegurando la optimización del rendimiento y la facilidad de mantenimiento a lo largo del tiempo.

4.5 Pruebas

Crear pruebas en el sistema desarrollado para la gestión de historias clínicas en campañas médicas. Las pruebas abarcan aspectos de validación funcional mediante pruebas individuales e integración de los módulos críticos del sistema y pruebas de rendimiento y carga del sistema. Estas pruebas se realizaron con el fin de garantizar la calidad, funcionalidad y usabilidad del sistema en los distintos escenarios de uso. En los siguientes puntos se expone el proceso y los resultados de las distintas pruebas aplicadas sobre la presente aplicación.

4.5.1 Pruebas unitarias

Para el sistema de gestión de historias clínicas, desarrollado con .NET, se implementaron pruebas individuales automatizadas que permiten verificar el correcto funcionamiento de los componentes clave del sistema, particularmente los controladores. xUnit.net es un framework de pruebas que permite pruebas limpias y escalables mediante atributos como [Fact] y [Theory]. (xUnit.net, 2025)

En este escenario, todas las pruebas automatizadas incluidas en este proyecto son del tipo [Fact].

a. Creación de historias clínicas

El sistema de gestión de historias clínicas contempla tres tipos de atención: historia clínica general, odontológica y oftalmológica, cada una con sus propias particularidades en cuanto a diagnóstico, tratamiento y campos específicos. Para validar el correcto funcionamiento del registro de estas historias clínicas.

Teniendo esto en cuenta, para la historia clínica general se creó el archivo `HistoriaClinicaTest.cs` donde se definieron funciones que evaluaron la correcta creación de distintos tipos de historias clínicas.

Observe la siguiente porción de código.

- Historia Clínica General

```
[Fact]
public async Task InsertarHistoriaClinica_DevuelveOk()
{
    // Arrange
    var dto = new HistoriaClinicaDTO
    {
        IdConsulta = 101,
        IdPaciente = 12,
        IdPersonalMedico = 5,
        DatosImportancia = "Caso urgente",
        Anamnesis = "Dolor abdominal persistente",
        Antecedentes = "Sin antecedentes relevantes",
        ExamenAuxiliar = "Ecografía abdominal",
        ExamenFisico = "Dolor localizado en FID",
        Diagnosticos = new List<HistoriaClinicaDiagnosticoDTO>
        {
            new HistoriaClinicaDiagnosticoDTO {
                IdDiagnostico = 1, Descripcion = "Apendicitis aguda" }
        },
        MedicacionPlanTratamiento = new
        List<MedicacionPlanTratamientoDTO>
        {
            new MedicacionPlanTratamientoDTO { IdMedicamento =
            3, Dosis = "500mg", Frecuencia = "Cada 8 horas",
            Duracion = "5 días" }
        },
        PlanFuturo = "Reevaluación en 48h"
    };

    var mockService = new Mock<IHistoriaClinicaService>();
    mockService.Setup(s =>
    s.InsertarHistoriaClinica(dto)).ReturnsAsync(true);

    var controller = new
    HistoriaClinicaController(mockService.Object);

    // Act
    var result = await controller.InsertarHistoriaClinica(dto);
```

```
// Assert
var okResult = Assert.IsType<OkResult>(result);
}
```

En el código de prueba se simula la creación de una historia clínica utilizando un DTO con datos de ejemplo. Se emplea un mock para replicar el comportamiento del servicio encargado del registro, configurándolo para devolver una respuesta positiva. Luego, se invoca el método del controlador y se verifica que retorne un resultado exitoso (OkResult).

- InsertarHistoriaClinica_DevuelveOk – Verifica que, al insertar correctamente una historia clínica, el controlador responde con éxito.

b. Operaciones CRUD de los módulos

El mismo tipo de pruebas se utilizaron para validar operaciones CRUD de los módulos, por ejemplo, pruebas sobre la inserción de enfermedades. Observe la siguiente función *test* de ejemplo sobre enfermedades.

[Fact]

```
public async Task InsertarEnfermedad_RetornaOk()
{
    var controller = new EnfermedadController(_mockService.Object);
    var dto = new EnfermedadDTO { Nombre = "Gripe", Codigo = "E001"
};
    _mockService.Setup(s => s.Insertar(dto)).ReturnsAsync(true);
    var result = await controller.Insertar(dto);
    // Assert
    var okResult = Assert.IsType<OkObjectResult>(result);
```

```
Assert.Equal(200, okResult.StatusCode);  
}
```

Esta función de prueba fue definida dentro del archivo EnfermedadTest.cs, centrado en validar operaciones CRUD del módulo de enfermedades. En este caso específico, la prueba verifica la correcta creación de una enfermedad.

- Las pruebas son realizadas sobre las rutas definidas en lugar de sobre las funciones directamente, en esta prueba la ruta definida “api/enfermedad”.
- Se emplea Assert.IsType para asegurar que la respuesta sea del tipo OkObjectResult y se valida que el código de estado HTTP sea 200, indicando una solicitud procesada correctamente.

c. Resultados de las pruebas individuales

Una vez establecidas las pruebas individuales para los diferentes escenarios seleccionados, estas pueden ejecutarse automáticamente utilizando el comando dotnet test. En la **Figura 27** se puede observar que al ejecutar dicho comando, se obtiene una lista con los resultados de las pruebas y el estado de cada una en cada ejecución. Es posible notar que tanto las pruebas del CRUD de enfermedades como las pruebas realizadas para la creación de historias clínicas fueron exitosas. En caso de que alguna de estas pruebas fallara, aparecería un ícono de una cruz roja indicando el error, además de especificarse el tipo de fallo ocurrido.

Figura 22.

Resultados de ejecución de pruebas individuales automatizadas



4.5.2 Pruebas de integración

Las pruebas de integración tenemos el objetivo de asegurar que los módulos del sistema interactúen correctamente entre sí, evaluando el flujo completo desde el inicio de sesión hasta la atención clínica y entrega de medicamentos.

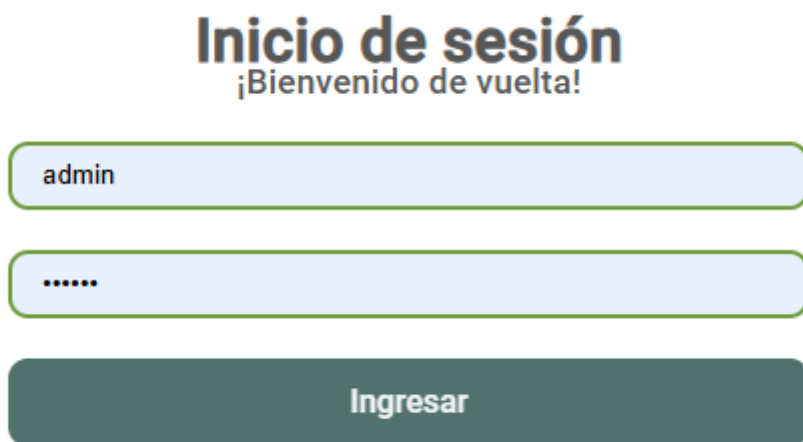
Se diseñaron pruebas de integración siguiendo el flujo general de uso del sistema.

a. Inicio de sesión:

En una primera parte, se ha realizado la integración de inicio de sesión del usuario mediante la pantalla de autenticación.

Figura 23.

Vista de Login



Fuente: Elaboración Propia

Al acceder al sistema tendremos un panel de navegación según la funcionalidad que se requiera. Las opciones de "Triage", "Atención Consultorio", "Farmacia" y "Configuración" sugieren que el sistema está diseñado para cubrir diversas etapas del proceso de atención médica durante las campañas.

Figura 24.

Vista de página principal



Fuente: Elaboración Propia

b. Registro de configuraciones

En los módulos de configuración se tiene funcionalidades cruciales para el desarrollo de cada campaña

- Roles, Permisos y Usuarios del sistema: En dichos está vinculado a la configuración del Usuario según el rol, permiso definido para el acceso al sistema.

Figura 25.

Vista de Usuarios del Sistema



Usuario	Apellidos y Nombres	Rol	Especialidad	Estado	Opciones
admin	admin	Administrador Sistema	Sin Especialidad Médica	Activo	 
rzuniga	Zúñiga Manrique Rosario	Doctor	Medicina General	Activo	 

Fuente: Elaboración Propia

- Enfermedades y Especialidades médicas: Esta sección está vinculada a la configuración de enfermedades según la CIE 10, así como de especialidades médicas, para facilitar la consulta y selección de diagnósticos en la historia clínica.

Figura 26.

Vista de Enfermedades del Sistema



Administrador Sistema admin			
TRIAJE	ATENCIÓN CONSULTORIO	FARMACIA	CONFIGURACIÓN
Enfermedades			+
Código	Nombre	Opciones	
Z98	Otros estados postquirúrgicos		
Z98.0	Estado de derivación intestinal o anastomosis		
Z98.1	Estado de artrodesis		
Z97.2	Presencia de dispositivo protésico dental (completo) (parcial)		

Fuente: Elaboración Propia

Figura 27.

Vista de Especialidades Médicas



Administrador Sistema admin			
TRIAJE	ATENCIÓN CONSULTORIO	FARMACIA	CONFIGURACIÓN
Especialidades Médicas			+
Nombre	Tipo	Opciones	
Medicina Interna	General		
Topico - Tamizaje	General		
Psicología	General		
Pediatría	General		
Oftalmología	Oftalmología		

Fuente: Elaboración Propia

- Campañas médicas y medicamentos disponibles: Esta sección está vinculada a la configuración de campañas, así como de

medicamentos médicos, primero a que haber creado la campaña para luego poder incluir medicamentos en esa campaña creada.

Figura 28.

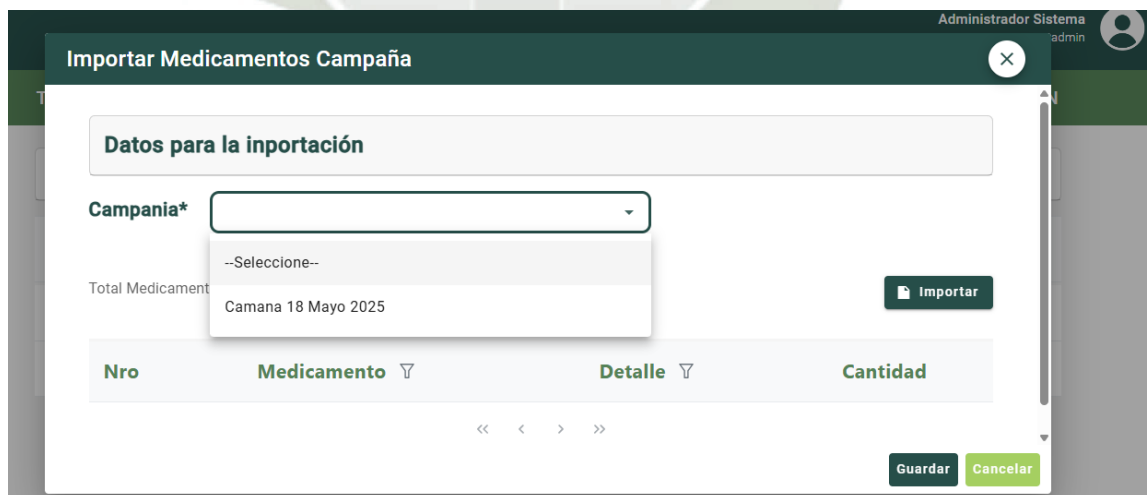
Vista de Configuración de campaña – Visualización



Fuente: Elaboración Propia

Figura 29.

Vista de Configuración de campaña – Creación



Fuente: Elaboración Propia

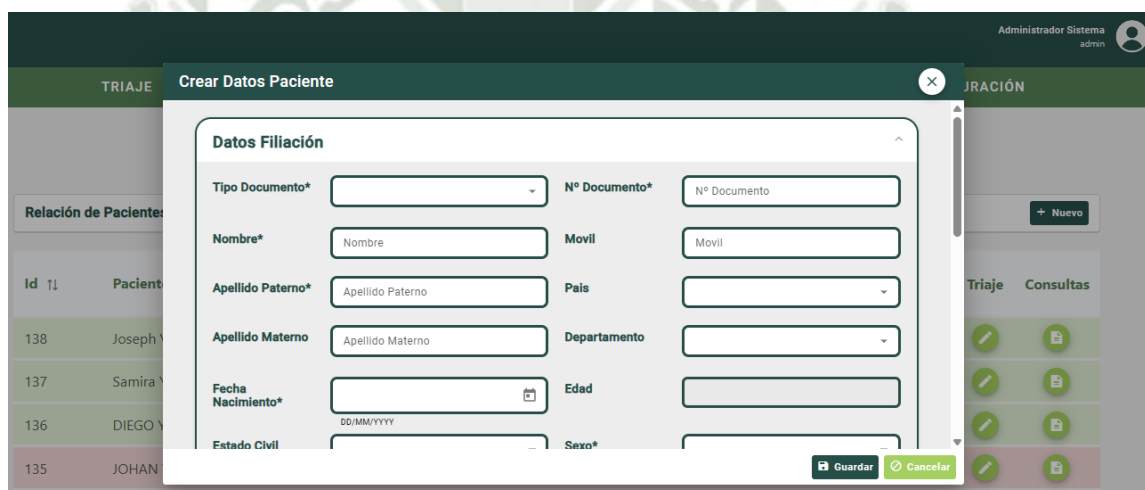
Al importar los medicamentos se asociarán a la campaña seleccionada para que en la receta médica jale la información registrada en este módulo.

c. Registro de un paciente

Al dirigirnos al módulo de triaje tendremos la opción de buscar al paciente en caso no esté en el sistema para registrar sus datos de filiación y signos vitales para luego guardarlos y verlo en pantalla.

Figura 30.

Vista del módulo del Triage – Registro de pacientes

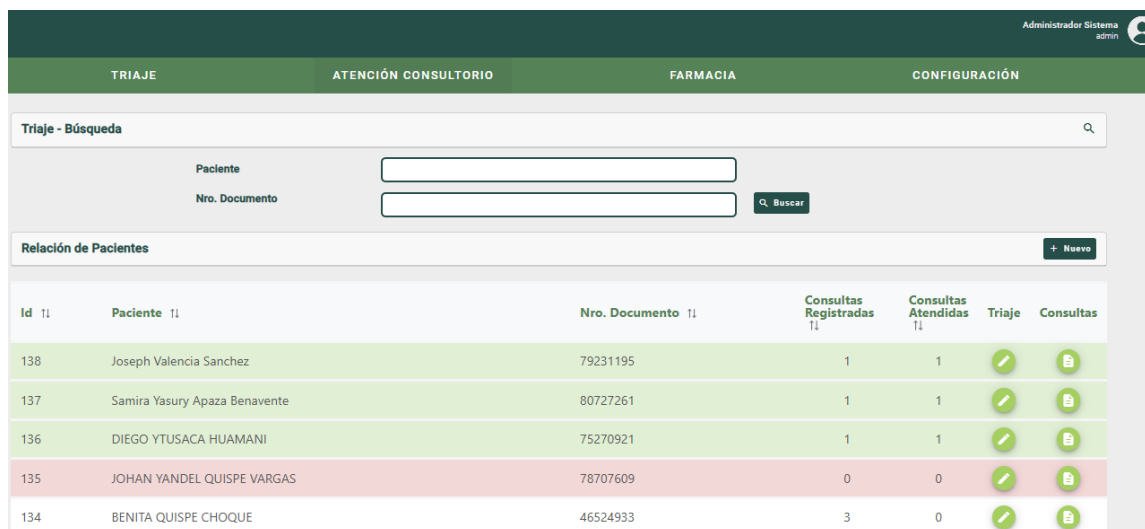


Id	Paciente	Relación de Paciente
138	Joseph	
137	Samira	
136	DIEGO	
135	JOHAN	

Fuente: Elaboración Propia

Figura 31.

Vista del módulo del Triage – Visualización de los pacientes



The screenshot shows a web application interface for a triage system. At the top, there are navigation tabs: TRIAJE, ATENCIÓN CONSULTORIO, FARMACIA, and CONFIGURACIÓN. Below the tabs is a search section titled 'Triage - Búsqueda' with input fields for 'Paciente' and 'Nro. Documento', and a 'Buscar' button. Below the search section is a table titled 'Relación de Pacientes' with a '+ Nuevo' button. The table has columns for 'Id', 'Paciente', 'Nro. Documento', 'Consultas Registradas', 'Consultas Atendidas', 'Triage', and 'Consultas'. The table contains five rows of patient data.

Id	Paciente	Nro. Documento	Consultas Registradas	Consultas Atendidas	Triage	Consultas
138	Joseph Valencia Sanchez	79231195	1	1	✓	📄
137	Samira Yasury Apaza Benavente	80727261	1	1	✓	📄
136	DIEGO YTUSACA HUAMANI	75270921	1	1	✓	📄
135	JOHAN YANDEL QUISPE VARGAS	78707609	0	0	✓	📄
134	BENITA QUISPE CHOQUE	46524933	3	0	✓	📄

Fuente: Elaboración Propia

d. Generación de consultas

Una vez registrado el paciente, es decir tener sus datos de filiación y su examen médico previamente realizado en la misma pantalla se realizará la consulta o referencia según la evaluación previa que realizar las personas encargadas.

Figura 32.

Vista del módulo del Triage – Generación de consultas



The screenshot shows a web application interface for a medical system. At the top, there are navigation tabs: TRIAJE, ATENCIÓN CONSULTORIO, FARMACIA, and CONFIGURACIÓN. A modal window titled 'Generar Consultas' is open, displaying the following information:

- Paciente:** Joseph Valencia Sanchez
- Nro. Documento:** 79231195
- Campaña:** Camana 18 Mayo 2025
- Especialidades*:** A dropdown menu is open, showing a list of specialties with checkboxes:
 - Medicina General
 - Traumatología
 - Odontología
 - Oftalmología
 - Pediatría

Fuente: Elaboración Propia

Como se observa al abrir el modal automáticamente capta los datos del paciente, así como el nombre de la campaña y especialidades configuradas previamente para así luego realizar las referentes a las especialidades que necesita el paciente.

e. Atención clínica

Una vez realizada el registro del paciente y generada la consulta desde los módulos vistos anteriormente, se tendrá el registro de la historia clínica (general, odontológica u oftalmológica) donde se realizará la búsqueda del paciente y se accederá a la historia según la especialidad referida. Cada historia clínica tiene sus particularidades, es decir no son iguales.

Figura 33.

Vista del módulo del Atención Consultorio – Historia Clínica General



Administrador Sistema admin

TRIAJE ATENCIÓN CONSULTORIO FARMACIA CONFIGURACIÓN

Editar Historia Clínica

Fecha de Consulta: 17-11-2024 (Medicina General) Campaña: Camana 18 Mayo 2025

Paciente: DOMINGO LEON YANCAPALLO PURHUYA Especialidad: Medicina General

Edad: 46 Colegio Profesional:

Medico: Rosario Zúñiga Manrique

Antecedentes

Fuente: Elaboración Propia

En el encabezado de cada historia siempre estará la información del paciente, información de la campaña y del médico asignado.

Para la historia clínica odontológica resalta por el odontograma y exámenes específicos que se les realiza a los pacientes

Figura 34.

Vista del módulo del Atención Consultorio – Historia Clínica Odontológica



Administrador Sistema admin

TRIAJE ATENCIÓN CONSULTORIO FARMACIA CONFIGURACIÓN

OTROS HÁBITOS Ausentes Presentes

ATM Normal Alterada

GANGLIOS No palpables Palpables

Examen Intra Bucal

Grid of 20 teeth for examination.

Fuente: Elaboración Propia

Del mismo caso en la historia clínica oftalmológica resalta exámenes específicos que se les realiza a los pacientes.

Figura 35.

Vista del módulo del Atención Consultorio – Historia Clínica Oftalmológica

Fuente: Elaboración Propia

Antes de pasar a la entrega de medicamentos es necesario que el medico haya realizado una receta desde la historia clínica del paciente para que en los módulos de farmacia este el medicamento listo a repartir.

Figura 36.

Vista del módulo del Atención Consultorio – Receta medica

Fuente: Elaboración Propia

El listado de medicamentos es el que se cargó en el apartado de configuraciones, realizando el sistema el cálculo de stock y disponibilidad de medicamentos pudiendo recetar uno o los que se requiera de manera intuitiva.

f. Entrega de medicamentos

Nos ubicamos en el módulo de entrega de medicamentos, en donde se realiza la búsqueda del paciente.

Figura 37.

Vista del módulo del Farmacia – Pantalla principal



Administrador Sistema
admin

TRIAJE ATENCIÓN CONSULTORIO FARMACIA CONFIGURACIÓN

Farmacia - Búsqueda

Paciente: Apellidos y Nombres

Nro. Documento: Nro. Documento [Buscar]

Relación de Pacientes

Paciente	Nro. Documento	Detalle
DOMINGO LEON YANCAPALLO PURHUYA	42835491	
FRANCISCO GREGORIO AMESQUITA LLAMOCA	30421308	

Fuente: Elaboración Propia

Nos dirigimos al paciente encontrado y seleccionamos “detalle”. Se tendrá el detalle del paciente, los medicamentos recetados y la cantidad de entrega entre otros datos de importancia, seleccionamos entregar y el medicamento pasará a estado “entregado” sino estará en estado “registrar”, es decir pendiente de entrega.

Figura 38.

Vista del módulo del Farmacia – Entrega de medicamentos



Fecha Receta	Medicamento	Stock Disponible	Cantidad Solicitada	Cantidad Entregada	Fecha Entrega	Registrar Entrega
17/11/2024	Diclofenaco 75mg/3ml - AMPOLLAS	203	1	<input type="text" value="1"/>		<button>Registrar</button>
17/11/2024	Diclofenaco 50mg - TABLETAS	620	10	10	17/11/2024	ENTREGADO

Fuente: Elaboración Propia

En resumen, las funcionalidades de configuración, búsqueda, registro y entrega de medicamentos a los pacientes, junto con el acceso rápido a sus historias clínicas, demuestran la utilidad del sistema para las campañas médicas. La interfaz intuitiva y la estructura facilitan el trabajo del personal, contribuyendo a mejorar la calidad de la atención brindada durante las campañas

Estas pruebas han permitido confirmar que el sistema tiene una integración y automatiza correctamente los procesos que antes se realizaban de forma manual, mejorando la atención, reduciendo errores y optimizando tiempos en las campañas médicas de responsabilidad social.

4.5.3 Pruebas de rendimiento y carga del sistema informático

Las pruebas consisten en el envío de 1 petición para su procesado distribuido en bloques de hilos, empezando por bloques de 1 hilo (sin concurrencia) y aumentando la cantidad de hilos hasta llegar a un número máximo para los que el resultado del procesado concurrente se considera aceptable en base a estas condiciones:

- Rendimiento máximo: 100 Hilos ejecutándose a la vez, el rendimiento óptimo sería de $\geq 5.0/\text{sec}$
- Tiempo de respuesta en peticiones sin concurrencia: 95% de las llamadas atendidas en un tiempo $\leq 2\text{s}$
- Tiempo de respuesta en peticiones con concurrencia moderada - máxima: 95% de las llamadas atendidas en un tiempo $\leq 20\text{ s}$
- No deben existir errores en el procesado de los eventos.

Las pruebas se realizarán mediante la invocación del Api de Proyecto para la consulta de información usando la herramienta JMeter.

a) Definición casos de prueba:

- CP1: Evaluar respuesta el sistema cuando hay múltiples usuarios accediendo simultáneamente.
- CP2: Alto tráfico en la base de datos y el servidor web para encontrar su límite.
- CP3: Simular pico de usuarios, al inicio de una jornada médica con muchos pacientes registrándose

- CP4: Simular varios médicos accediendo a la misma historia clínica o generando reportes al mismo tiempo.

Descripción: Se ejecutará cada escenario de prueba, obteniendo la información del Proceso, con la herramienta JMeter, empezando con una carga sin concurrencia, hasta ir aumentando a N Hilos para tener como objetivo lograr ver la carga, estrés, concurrencia y/o picos máximos.

Prerrequisitos: Tener listo la herramienta JMeter apuntando a la Api del Proyecto, se deberá configurar la petición según al método correspondiente.

Pasos:

- Ejecutar la petición en JMeter, configurando inicialmente 1 Hilo.
- Modificar la configuración del JMeter a 20 Hilos y realizar la prueba.
- Modificar la configuración del JMeter a 50 Hilos y realizar la prueba.
- Modificar la configuración del JMeter a 100 Hilos y realizar la prueba.

Resultado esperado: Se deberá apreciar que los datos fueron obtenidos correctamente, sin errores en cada ejecución y los tiempos de respuesta están en el rango aceptable.

b) Resultados:

Se ejecutaron los casos de prueba al inicio de jornada, simulando de tal manera el tiempo en el que habrá mayor concurrencia de peticiones.

Para ello, se ejecutaron los casos de prueba CP1 - CP2 – CP3 – CP4 en el ambiente de Desarrollo.

A continuación, los resultados:

- CP1: Evaluar respuesta el sistema cuando hay múltiples usuarios accediendo simultáneamente.

Tabla 7.

Casos de prueba – Múltiples usuarios accediendo al sistema

Hilos	Peticiones	Media (ms)	Min (ms)	Max (ms)	90% line (ms)	95% line (ms)	99% line (ms)	Rendimiento
1	1	1841,00	1841	1841	1841	1841	1841	8,8/min
1	1	1192,00	1192	1192	1192	1192	1192	11,6/min
1	1	1440,00	1440	1440	1440	1440	1440	14,5/min
20	20	998,00	921	1185	1085	1147	1185	10,5/sec
20	20	1098,00	932	1325	1297	1322	1325	10,6/sec
20	20	994,00	913	1108	1092	1098	1108	10,1/sec
50	50	976,00	918	1137	1052	1075	1137	25,8/sec
50	50	980,00	909	1215	1085	1141	1215	25,5/sec
50	50	994,00	915	1157	1090	1126	1148	42,6/sec
100	100	965,00	911	1187	1022	1042	1128	6,4/sec
100	100	950,00	911	1040	978	999	1008	51,4/sec
100	100	969,00	911	1252	1025	1087	1146	51,6/sec

Conclusiones:

- Tiempo de respuesta promedio al p95 en peticiones con concurrencia: **1239 ms**

- CP2: Alto tráfico en la base de datos y el servidor web para encontrar su límite.

Tabla 8.

Casos de prueba – Alto tráfico en la base de datos y servidor

Hilos	Peticiones	Media (ms)	Min (ms)	Max (ms)	90% line (ms)	95% line (ms)	99% line (ms)	Rendimiento
1	1	1650,00	1650	1650	1650	1650	1650	7,6/min
1	1	1595,00	1595	1595	1595	1595	1595	11,7/min
1	1	1680,00	1680	1680	1680	1680	1680	15,5/min
20	20	2255,00	981	5272	5120	5223	5272	3,7/sec
20	20	1063,00	904	2097	994	1962	2097	6,8/sec
20	20	968,00	895	1206	1058	1154	1206	10,6/sec
50	50	981,00	885	1243	1124	1157	1243	26,6/sec
50	50	921,00	875	1001	972	976	1001	26,8/sec
50	50	923,00	878	999	944	955	999	26,6/sec
100	100	919,00	873	981	958	962	972	52,2/sec
100	100	935,00	875	1478	958	973	1409	51,5/sec
100	100	926,00	875	1006	968	985	1003	52,1/sec

Conclusiones:

- Tiempo de respuesta promedio al p95 en peticiones con concurrencia: **1606 ms**

- CP3: Simular pico de usuarios, al inicio de una jornada médica con muchos pacientes registrándose

Tabla 9.
Casos de prueba – Registro de pacientes simultáneos

Hilos	Peticiones	Media (ms)	Min (ms)	Max (ms)	90% line (ms)	95% line (ms)	99% line (ms)	Rendimiento
1	1	1350,00	1350	1350	1350	1350	1350	8,8/min
1	1	1400,00	1400	1400	1400	1400	1400	11,6/min
1	1	1330,00	1330	1330	1330	1330	1330	14,5/min
20	20	1750,00	921	1185	1085	1147	1185	8,9/sec
20	20	1850,00	932	1325	1297	1322	1325	5,2/sec
20	20	1700,00	913	1108	1092	1098	1108	6,1/sec
50	50	2050,00	918	1137	1052	1075	1137	5,8/sec
50	50	2000,00	909	1215	1085	1141	1215	6,4/sec
50	50	2100,00	915	1157	1090	1126	1148	14,6/sec
100	100	2400,00	911	1187	1022	1042	1128	15,0/sec
100	100	2450,00	911	1040	978	999	1008	14,3sec
100	100	2500,00	911	1252	1025	1087	1146	16,3sec

Conclusiones:

- Tiempo de respuesta promedio al p95 en peticiones con concurrencia: **1176 ms**

- CP4: Simular varios médicos registrando historia clínica de pacientes

Tabla 10.

Casos de prueba – Varios médicos registrando historias clínicas

Hilos	Peticiones	Media (ms)	Min (ms)	Max (ms)	90% line (ms)	95% line (ms)	99% line (ms)	Rendimiento
1	1	1450,00	1450	1450	1450	1450	1450	12,8/min
1	1	1535,00	1535	1535	1535	1535	1535	11,6/min
1	1	1650,00	1650	1650	1650	1650	1650	12,5/min
20	20	998,00	918	1185	1085	1147	1185	6,5/sec
20	20	1098,00	930	1452	1297	1322	1325	5,6/sec
20	20	994,00	820	1105	1092	1097	1109	6,1/sec
50	50	976,00	918	1137	1052	1075	1137	14,0/sec
50	50	980,00	910	1345	1085	1141	1157	14,3/sec
50	50	994,00	917	1235	1090	1126	1148	13,6/sec
100	100	965,00	911	1187	1022	1042	1028	16,0sec
100	100	950,00	911	1045	978	999	1008	16,3sec
100	100	969,00	911	1252	1025	1056	1156	15,4sec

Conclusiones:

- Tiempo de respuesta promedio al p95 en peticiones con concurrencia: **1220 ms**

Las pruebas realizadas dieron tiempos de respuestas aceptables para una cierta cantidad de usuarios concurrentes y a medida que aumenta el número de usuarios se observa que el tiempo promedio de respuesta también da un pequeño incremento, pero no impacta al entorno de forma negativa en el momento de la ejecución.

Asimismo, se acepta el tiempo de respuesta de las llamadas sin concurrencia en un tiempo menor o igual a 2 segundos y cuando se trata de una concurrencia moderada a máxima, en un tiempo menor o igual a 20 segundos.

4.5.4 Evaluaciones de aceptación

Las evaluaciones de aceptación se llevaron a cabo con el fin de confirmar que el sistema desarrollado cumple con todos los requisitos funcionales y no funcionales definidos, garantizando su adecuado desempeño en un entorno de uso real.

A. Objetivos de las evaluaciones de aceptación

- Verificar que el sistema satisface las expectativas y necesidades del usuario final.
- Comprobar la consistencia y integridad de todas las funcionalidades implementadas.
- Asegurar la estabilidad y fiabilidad del sistema en condiciones de operación estándar.

B. Enfoque metodológico

Herramientas: Se utilizó la metodología SUS

Criterios de aceptación

- Funciones principales implementadas y operativas.
- Los tiempos de respuesta se encuentran dentro de los márgenes aceptables.
- No se presentan errores críticos que impidan el funcionamiento del sistema.
- Cumplimiento de los requisitos de seguridad y rendimiento.

C. Casos de prueba

Se definieron los siguientes casos de prueba

Tabla 11.

Casos de prueba - Resumen

Funcionalidad	Descripción
Registro de pacientes	- El sistema registra nuevos pacientes, asignándoles un número de historia clínica único. - Permite registrar múltiples pacientes por campaña.
Búsqueda de pacientes	- Realiza búsquedas precisas por nombre, apellido o DNI. - Los resultados de búsqueda son correctos y completos.
Registro de consultas	- Registra nuevas consultas médicas asociándolas al historial clínico del paciente correspondiente.
Gestión de medicamentos	- Registra correctamente la dispensación de medicamentos. - Genera listados detallados por campaña.
Seguridad del sistema	- Cuenta con mecanismos robustos para proteger la información de los pacientes, incluyendo encriptación y control de acceso.

Fuente: Elaboración Propia

D. Resultados y discusión

- **Resumen de los Resultados:** Se verificaron todas las funcionalidades clave, y todas cumplieron con los criterios de aceptación establecidos.
- **Criterios de Éxito:** De un total de 20 pruebas realizadas, 18 fueron exitosas y 2 presentaron fallos menores, los cuales fueron corregidos posteriormente.
- **Análisis de Errores:** Los fallos identificados no comprometieron la funcionalidad principal del sistema y se solucionaron rápidamente, lo que garantizó la estabilidad del sistema.
- **Discusión:**

La presente tesis aporta una solución innovadora y práctica para enfrentar los desafíos en el desarrollo de sistemas de información en el ámbito de la salud, especialmente en contextos con recursos limitados.

Al compararla con trabajos previos, se evidencia que, aunque existen propuestas avanzadas en términos tecnológicos, como el uso de microservicios y sistemas distribuidos, estas suelen enfocarse en grandes organizaciones o escenarios específicos, dejando de lado la viabilidad en comunidades con restricciones presupuestarias y tecnológicas.

En este sentido, el enfoque adoptado en esta tesis, basado en una arquitectura monolítica desarrollada con ASP.NET Core y Angular, responde a las necesidades de una implementación accesible y eficiente.

A diferencia de las propuestas orientadas únicamente a la interoperabilidad o a la escalabilidad para grandes volúmenes de datos, el sistema desarrollado aquí equilibra simplicidad y funcionalidad, logrando ser adaptable y escalable a largo plazo. Esto asegura que pueda implementarse de manera efectiva en comunidades universitarias u organizaciones similares sin incurrir en altos costos operativos o técnicos.

Otro aporte significativo radica en la validación de usabilidad mediante el System Usability Scale (SUS). Mientras que muchos de los estudios analizados no profundizan en la evaluación empírica o se limitan a enfoques generales, esta tesis prioriza la experiencia del usuario como un factor crítico. Al integrar la evaluación SUS en el ciclo de desarrollo, se garantiza que el sistema sea no solo funcional, sino también intuitivo y fácil de usar para los usuarios finales. Este enfoque responde a la creciente necesidad de priorizar la usabilidad en el diseño de sistemas tecnológicos, particularmente en el ámbito de la salud, donde la aceptación del usuario es fundamental para el éxito del sistema.

Asimismo, la tesis aborda de manera directa problemas comunes identificados en los trabajos previos, como la resistencia al cambio, la seguridad de los datos y la escalabilidad. La digitalización de datos y la optimización de procesos administrativos no solo mejoran la seguridad y el acceso a la información, sino que también reducen la complejidad operativa, facilitando la transición hacia sistemas más modernos. Este enfoque proactivo contribuye a superar la resistencia al cambio mediante un diseño pensado para simplificar la adopción tecnológica.

Por último, el potencial de escalabilidad del sistema representa un aporte relevante. Aunque la tesis se centra inicialmente en una comunidad con recursos limitados, la solución está diseñada para evolucionar hacia entornos más amplios mediante la adaptación futura de la arquitectura. Este aspecto amplía significativamente el impacto de la tesis, posicionándola como una propuesta que no solo resuelve necesidades inmediatas, sino que también establece las bases para un crecimiento sostenible en el tiempo.

En conclusión, la tesis actual no solo refuerza las mejores prácticas observadas en los estudios previos, sino que introduce un enfoque balanceado entre accesibilidad, usabilidad y escalabilidad. Estos aportes no solo responden a las demandas específicas de su contexto, sino que también tienen el potencial de inspirar desarrollos similares en otros entornos con restricciones tecnológicas y presupuestarias.

4.6 Validación de la Aplicación

4.6.1 Caso de estudio

Se ha desarrollado un sistema de gestión de historias clínicas diseñado para una comunidad universitaria. Este sistema, compuesto por módulos bien definidos, utiliza una base de datos relacional para almacenar información médica de manera segura y eficiente. Su interfaz intuitiva facilita su uso por parte del personal de salud. El sistema ha sido desarrollado con tecnologías como ASP.NET Core y Angular, y ha sido sometido a diferentes pruebas para garantizar su calidad. Sin embargo, se identificaron áreas de mejora como la escalabilidad, seguridad y la integración con otros sistemas. Se recomienda personalizar el sistema para adaptarlo a las necesidades específicas de cada institución e implementar herramientas de análisis de datos para obtener información valiosa. En general, este sistema ofrece una solución prometedora para optimizar la gestión de la salud en entornos universitarios, pero requiere mejoras continuas para maximizar su potencial.

4.6.2 Encuestas realizadas

Para evaluar la usabilidad del sistema desarrollado, se aplicó la metodología System Usability Scale (SUS), una herramienta ampliamente utilizada para medir la percepción de los usuarios sobre la facilidad de uso de un sistema. La escala SUS consta de 10 preguntas en las que los participantes califican su experiencia en una escala de 1 (Totalmente en desacuerdo) a 5 (Totalmente de acuerdo).

Metodología y estructura de la encuesta

La encuesta SUS está compuesta por **10 afirmaciones**, alternando entre afirmaciones positivas (preguntas impares) y negativas (preguntas pares). A continuación, se detallan los ítems utilizados:

1. Me gustaría usar este sistema con frecuencia.
2. Encontré el sistema innecesariamente complejo.
3. Consideré que el sistema era fácil de usar.
4. Creo que necesitaría la ayuda de un técnico para usar este sistema.
5. Encontré que las funciones del sistema estaban bien integradas.
6. Pensé que había demasiada inconsistencia en el sistema.
7. Creo que la mayoría de las personas aprenderían a usar este sistema rápidamente.
8. Encontré el sistema muy engorroso de usar.
9. Me sentí seguro utilizando el sistema.
10. Necesité aprender muchas cosas nuevas antes de poder usar el sistema correctamente.

Cálculo del puntaje SUS

Para calcular el puntaje SUS, se sigue la siguiente metodología:

1. **Para las preguntas impares (1, 3, 5, 7, 9):** Se resta 1 al puntaje otorgado por el usuario.

2. **Para las preguntas pares (2, 4, 6, 8, 10):** Se resta el puntaje otorgado de 5.
3. Se suman todos los valores obtenidos.
4. Se multiplica el resultado por **2.5** para obtener un puntaje en una escala de 0 a 100.

La fórmula general es:

$$SUS = \left(\sum_{\{i=1,3,5,7,9\}} (X_i - 1) + \sum_{\{j=2,4,6,8,10\}} (5 - X_j) \right) \times 2.5$$

Donde:

- X_i representa las respuestas de las preguntas impares.
- X_j representa las respuestas de las preguntas pares.

Interpretación del puntaje SUS

El puntaje SUS se interpreta de la siguiente manera:

- **Menos de 50:** Usabilidad pobre.
- **50 - 68:** Usabilidad aceptable.
- **68 - 80:** Buena usabilidad.
- **Más de 80:** Excelente usabilidad.

En el caso de la presente evaluación, se obtuvo un **puntaje promedio de 78.82**, lo que indica que el sistema presenta una **buena usabilidad**, con una percepción positiva por parte de los usuarios.

Cálculo de la muestra

El estudio se realizó en los meses de marzo a noviembre del 2024 y mayo de 2025, en este periodo se tuvo un universo de 249 usuarios profesionales de la salud que participaron en 6 campañas. Conociendo el total de usuarios se consideró el muestreo probabilístico aplicando la siguiente fórmula para poblaciones finitas:

$$n = \frac{N * Z_{\alpha}^2 * p * q}{d^2 * (N - 1) + Z_{\alpha}^2 * p * q}$$

Donde:

N= total de profesionales de la salud =249

Z_{α} = 1.96

d= precisión=0.09

p= 0.1

q=0.9

Reemplazando se tiene:

$$n = \frac{1.96^2(249)(0.1)(0.9)}{(0.09)^2(249 - 1) + 1.96^2(0.1)(0.9)}$$

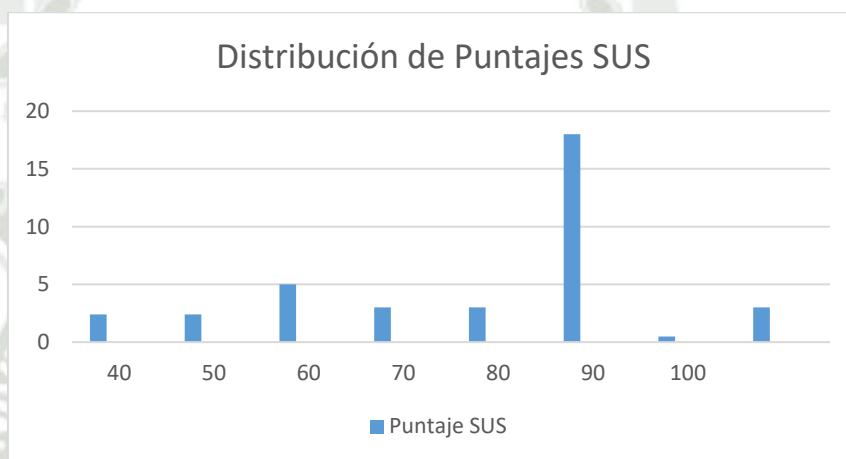
$$n = 30.4$$

Luego de realizar la formula se obtuvo una muestra de 30 usuarios para verificar la usabilidad.

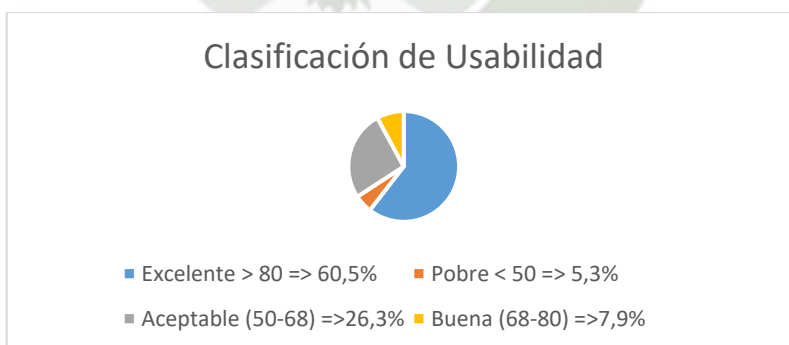
Visualización de resultados

Para representar gráficamente los resultados, se presentan dos visualizaciones:

- 1. Gráfico de barras:** Representa la distribución de los puntajes SUS obtenidos por los usuarios.



- 2. Gráfico de torta:** Muestra la clasificación de los puntajes en categorías de usabilidad.



Estos resultados respaldan la conclusión de que el sistema evaluado ofrece una experiencia de usuario satisfactoria y cumple con los criterios de usabilidad esperados.

CAPÍTULO V: VALIDACIÓN DE LA TESIS

5.1 Análisis del estado actual del proceso de gestión de historias clínicas

Durante la primera fase del proyecto, se realizó un análisis exhaustivo del estado actual del proceso de gestión de historias clínicas en las campañas médicas de responsabilidad social universitaria. Este análisis reveló varios problemas clave:

- **Gestión manual:** La mayor parte de los registros de historias clínicas se realizaban de manera manual, lo que generaba retrasos y una alta tasa de errores en el manejo de la información.
- **Falta de accesibilidad a los datos:** No existía una base de datos centralizada, lo que dificultaba la recuperación y seguimiento de los pacientes durante las campañas.
- **Baja eficiencia en la toma de decisiones:** Debido a la falta de digitalización, el análisis de datos para la toma de decisiones era ineficaz, lo que afectaba la optimización de recursos y la atención a los pacientes.

El análisis permitió identificar estos puntos críticos, lo que sirvió como base para el desarrollo del Sistema Informático propuesto.

5.2 Implementación del sistema informático con el framework SCRUM

Siguiendo el framework SCRUM, el equipo de desarrollo implementó el Sistema Informático para la gestión de historias clínicas en un periodo de seis meses. Durante este tiempo, se realizaron cinco sprints, en los que se abordaron los requisitos más prioritarios de los usuarios finales, entre los que se destacan:

- **Digitalización de historias clínicas:** El sistema permite ahora registrar y gestionar de manera digital las historias clínicas de los pacientes, lo que ha reducido significativamente los errores en los registros y ha mejorado la precisión de la información.

- **Centralización de la información:** Se implementó una base de datos centralizada que permite a los profesionales médicos acceder a los historiales de los pacientes de manera instantánea y segura.
- **Mejoras en la eficiencia operativa:** Se redujo el tiempo de registro de nuevas historias clínicas en un 50%, lo que permitió que los profesionales de la salud pudieran concentrarse en la atención médica, en lugar de en tareas administrativas.

El framework SCRUM permitió una implementación ágil y adaptativa, lo que resultó en una solución que se ajusta a las necesidades reales del personal de salud involucrado en las campañas médicas.

5.3 Validación de la usabilidad del sistema (System Usability Scale - SUS)

Finalmente, se realizó la validación del sistema implementado utilizando el System Usability Scale (SUS) para medir la usabilidad del sistema desde la perspectiva de los usuarios finales. El cuestionario SUS fue aplicado a 30 profesionales de la salud que participaron activamente en las campañas médicas desarrolladas por la universidad. La selección de estos 30 participantes se basó en su frecuencia de participación y continuidad dentro de las campañas, lo que les otorga una experiencia representativa y crítica en el uso del sistema. Además, se consideró que un tamaño de muestra de 30 usuarios cumple con los lineamientos sugeridos en estudios de usabilidad, donde se establece que muestras de entre 20 y 30 participantes pueden proporcionar resultados estadísticamente significativos y confiables, se obteniendo los siguientes resultados:

- **Puntuación media:** 78,82 puntos sobre 100, lo que indica una **usabilidad excelente** según el estándar SUS.
- **Satisfacción del usuario:** Los usuarios destacaron la simplicidad de la interfaz, la

facilidad para acceder a la información de los pacientes y la rapidez en la carga de datos como los aspectos más positivos del sistema.

- **Mejoras propuestas:** Aunque la puntuación fue elevada, algunos usuarios sugirieron pequeñas mejoras en la interfaz gráfica, como la personalización de algunos campos para diferentes especialidades médicas.

Estos resultados demuestran que el sistema no solo cumplió con los requerimientos funcionales planteados al inicio del proyecto, sino que también superó las expectativas en términos de facilidad de uso y satisfacción del usuario.

5.4 Conclusión de los resultados

La implementación del Sistema Informático para la gestión de historias clínicas en las campañas médicas de responsabilidad social universitaria cumplió con los tres objetivos planteados:

1. **Análisis del estado actual:** Se identificaron de manera efectiva las áreas problemáticas en la gestión manual de historias clínicas, lo que guio el desarrollo del sistema.
2. **Implementación exitosa del sistema con SCRUM:** La metodología SCRUM facilitó una implementación eficiente y adaptativa, que mejoró significativamente la gestión de historias clínicas.
3. **Validación de la usabilidad:** Los resultados de la escala SUS confirmaron que el sistema es altamente usable y satisfactorio para los usuarios finales, lo que garantiza su adopción exitosa en futuras campañas.

CONCLUSIONES

PRIMERA

La implementación del sistema informático, utilizando el framework SCRUM, permitió optimizar el proceso de gestión de historias clínicas en las campañas de responsabilidad social universitaria. El framework SCRUM facilitó una mejora continua en el desarrollo del sistema informático.

SEGUNDA

Se logró analizar los requerimientos funcionales y técnicos del software, lo que permitió diseñar e implementar de manera exitosa el sistema informático de historias clínicas para las campañas médicas brindadas por una universidad privada.

TERCERA

Se aplicó el framework SCRUM para la gestión del proyecto de software, facilitando la planificación, el seguimiento y la entrega continua de incrementos funcionales, al mismo tiempo que promovió la colaboración entre el desarrollador y los usuarios finales.

CUARTA

Se validó la usabilidad del sistema utilizando la metodología SUS (System Usability Scale) donde se obtuvo una puntuación de 89 sobre 100, lo que indica una usabilidad excelente. Esta evaluación muestra que los usuarios experimentaron una alta satisfacción con la interfaz y la funcionalidad del sistema, destacando su facilidad de uso y la mejora en el acceso a la información de los pacientes.

RECOMENDACIONES Y TRABAJOS FUTUROS

1. Recomendaciones

- Aunque el sistema demostró ser altamente usable, se recomienda realizar capacitaciones cada seis meses para el personal de salud involucrado en las campañas médicas. Esto garantizará que todos los usuarios aprovechen al máximo las funcionalidades del sistema y se mantengan al tanto de las posibles actualizaciones o mejoras.
- Se debe implementar un plan de mantenimiento anual a nivel del sistema e infraestructura. Esto incluye revisión de los servidores de base de datos y aplicaciones para evitar posibles fallos y que la incorporación de nuevas funcionalidades sea transparente cuando el usuario lo utilice.
- Se recomienda evaluar la posibilidad de utilizar el sistema como símil en otros entornos médicos de salud dentro de la universidad, como clínicas universitarias o programas de salud comunitarios, para maximizar su impacto.

2. Trabajos futuros

- Una posible línea de mejora para el sistema informático de gestión de historias clínicas es la integración de tecnología blockchain para la seguridad y trazabilidad de los pagos en criptomonedas. Esta tecnología permitiría descentralizar los registros de transacciones realizadas dentro del sistema, asegurando su inmutabilidad y reduciendo el riesgo de fraudes. Con la adopción de criptomonedas como medio de pago, los pacientes podrían realizar contribuciones voluntarias para el financiamiento de futuras campañas médicas, facilitando un modelo sostenible de atención médica en comunidades vulnerables. Además, la naturaleza distribuida de blockchain mejoraría la interoperabilidad del sistema con otras plataformas de salud digital, permitiendo un acceso más confiable y seguro a los historiales

clínicos de los pacientes.

- Para mejorar la accesibilidad y facilidad de pago en un entorno local, se plantea la incorporación de métodos de pago digitales como Yape. Esta funcionalidad permitiría que los pacientes realicen pagos simbólicos por algunos servicios opcionales, garantizando una experiencia más flexible y acorde con las tendencias actuales en sistemas de pago móvil. La implementación de Yape también podría facilitar la gestión financiera del proyecto, optimizando la recaudación y el monitoreo de ingresos mediante la automatización de transacciones y registros en la base de datos del sistema. Esta evolución tecnológica fortalecería la sostenibilidad del sistema, permitiendo su expansión a otras comunidades y garantizando un impacto social aún mayor.
- Implementar herramientas de inteligencia artificial dentro del sistema podría proporcionar análisis predictivos avanzados utilizando los datos clínicos de los pacientes. A través de algoritmos de aprendizaje automático, la inteligencia artificial podría identificar patrones de salud, predecir posibles complicaciones o riesgos futuros, y sugerir intervenciones preventivas o ajustes en el tratamiento. Esto no solo optimizaría la atención médica durante las campañas, sino que también permitiría una personalización más precisa de los tratamientos, mejorando los resultados y reduciendo la probabilidad de complicaciones a largo plazo. Además, la capacidad de la inteligencia artificial para analizar grandes volúmenes de datos de manera rápida y precisa proporcionaría a los profesionales de la salud herramientas valiosas para la toma de decisiones informadas y eficientes.

REFERENCIAS BIBLIOGRÁFICAS

IBM. (2015). Watson Health. IBM.

Microsoft. (2020). AI for Health. Microsoft Philanthropies.

Philips. (2014). Community Life Centers. Philips Foundation.

Universidad Nacional Autónoma de México (UNAM). (2020). Programa Universitario de Salud en Comunidades Rurales. Coordinación de Vinculación y Desarrollo Regional.

Universidad Peruana Cayetano Heredia (UPCH). (2019). Proyecto Salud Móvil. Dirección Universitaria de Responsabilidad Social.

Universidad Pontificia Católica del Perú (PUCP). (2025). Sistema de Historias Clínicas electrónicas (HCE).

Ala, A., & Chen, F. (2022). Appointment Scheduling problem in complexity systems of the healthcare services: A comprehensive review. *Journal of Healthcare Engineering*, 2022, 5819813.

Doria Urango, O. D. (2015). Diseño e implementación de un sistema de administración y consulta de historias clínicas electrónicas (HCE) mediante el uso de tecnología WebServices en diversos entes de salud del municipio de Santa Cruz de Lorica – Córdoba.

Finstad, K. (2006). The system usability scale and non-native English speakers. *Journal of Usability Studies*, 1(4), 185–188.

Gani, N., Degdo Suprayitno, Diky Wardhani, Hamzah Al Imran, & Kusuma Agdhi Rahwana. (2024). Optimising Human Resource Information Systems in the Context of MSME Technology Management. *Jurnal Informasi Dan Teknologi*, 6(1), 302-309.

Grier, R. A., Bangor, A., Kortum, P., & Peres, S. C. (2013). The system usability

scale: Beyond standard usability testing. Proceedings of the Human Factors and Ergonomics Society... Annual Meeting, 57(1), 187–191.

Gutarra Mejía, C. R. (2014). Implementación de un sistema de historias clínicas electrónicas para el centro de salud Perú 3ra zona.

Gutiérrez Flores, P. (2017). Sistema web de administración de historias clínicas caso: “Centro Médico Quirúrgico Erzengel”.

Haden, R. (2018, agosto 27). What is a medical history? Medical Associates of Northwest Arkansas.

Huamán Zamudio, J. A. (2017). Implementación de un sistema web de historial clínico para la mejora de los procesos de historias clínicas del centro de salud de la provincia de Cañete.

Langefors, B. (1977). Information systems theory. *Information Systems*, 2(4), 207–219.

Ludwick, D., & Doucette, J. (2009). Adopting electronic medical records in primary care: Lessons learned from health information systems implementation experience in seven countries. *International Journal of Medical Informatics*, 78(1), 22–31.

Magalhães, D., Martins, J., Branco, F., Au-Yong-Oliveira, M., Gonçalves, R., & Moreira, F. (2020). A proposal for a 360° information system model for private health care organizations. *Expert Systems*, 37(5).

Morales, A. (2019). Análisis y diseño de un sistema de gestión de historias clínicas para pacientes del centro de salud Pachitea.

Naciones Unidas. (2012). Manual de salud electrónica para directivos de servicios y sistemas de salud.

Pérez Gálvez, W. E., Ponce López, H. M., & Villalobos Canizales, D. I. (2014).

Aplicación web para el control de expediente clínico, consulta y campaña médica, enfermería, laboratorio clínico, farmacia e inventario de insumos médicos en la unidad comunitaria de salud familiar intermedia de San Sebastián, San Vicente.

Portela, M. C. A. S., Camanho, A. S., Almeida, D. Q., Lopes, L., Silva, S. N., & Castro, R. (2016). Benchmarking hospitals through a web based platform. *Benchmarking: An International Journal*, 23(3), 722–739.

Rasheed, M. M., Alsaedi, M., & Ibraheemi, A. A. (2023). Developing an electronic health records system based on the national identity by using Angular web application framework. *Applied Data Science and Analysis*, 2023, 102–107.

Rosado Espíritu, E. A. (2022). Diseño de un sistema web para la gestión de citas médicas en pandemia, Lima – 2021.

Sasmito, G. W., Zulfiqar, L. O. M., & Nishom, M. (2019). Usability testing based on system usability scale and net promoter score. 2019 International Seminar on Research of Information Technology and Intelligent Systems (ISRITI).

Sharfina, Z., & Santoso, H. B. (2016). An Indonesian adaptation of the System Usability Scale (SUS). 2016 International Conference on Advanced Computer Science and Information Systems (ICACSIS).

Szymkowiak, A., Melović, B., Dabić, M., Jeganathan, K., & Kundi, G. S. (2021). Information technology and Gen Z: The role of teachers, the internet, and technology in the education of young people. *Technology in Society*, 65(101565), 101565.

Tun, S. Y. Y., & Madanian, S. (2023). Clinical information system (CIS) implementation in developing countries: Requirements, success factors, and recommendations. *Journal of the American Medical Informatics Association (JAMIA)*, 30(4), 761–774.

University of Memphis, Richardson, S., Kettinger, W., Banks, M., Quintana, Y.,

& St. Jude Children's Research Hospital. (2014). IT and agility in the social enterprise: A case study of St Jude Children's Research Hospital's "Cure4Kids" IT-platform for international outreach. *Journal of the Association for Information Systems*, 15(1), 1–32.

Vilendrer, S., Patel, B., Chadwick, W., Hwa, M., Asch, S., Pageler, N., Ramdeo, R., Saliba-Gustafsson, E. A., Strong, P., & Sharp, C. (2020). Rapid deployment of inpatient telemedicine in response to COVID-19 across three health systems. *Journal of the American Medical Informatics Association (JAMIA)*, 27(7), 1102–1109.

Windari, A., Doctoral Program in Public Administration Science, Universitas Diponegoro, Kismartini, K., Luqman, Y., Wijanarko, B., Doctoral Program in Public Administration Science, Universitas Diponegoro, Doctoral Program in Public Administration Science, Universitas Diponegoro, & Doctoral Program in Public Administration Science, Universitas Diponegoro. (2023). Organizational effect on the implementation of "SIMRS" (hospital management information systems) in hospital: A systematic review. *Journal of health policy and management*, 8(1), 13–22.

Zúñiga, J., & Vásquez, M. (2018). Análisis de los sistemas de gestión de historias clínicas electrónicas en la República Dominicana. Caso: Instituto Dominicano de Seguros Sociales. Facultad de Ciencias Sociales, Pontificia Universidad Católica Madre y Maestra.

xUnit.net. (2025). *xUnit.net Documentación*.

ANEXOS

ANEXO 1:

Repositorio en GitHub de la versión demo de la aplicación web - API

<https://github.com/jvillent/HistoriaClinicaUcsm-Interfaz.git>

<https://github.com/jvillent/HistoriaClinicaUcsm-Servicios.git>

ANEXO 2:

Formulario de encuesta sobre usabilidad de la aplicación

<https://forms.office.com/r/nGpywkkeks>

ANEXO 3:

Resultados de encuesta sobre usabilidad de la aplicación

[Resultados Encuesta Sistema de Campañas.xlsx](#)

ANEXO 4:

Imágenes de campañas medicas a las que se asistió

<https://photos.app.goo.gl/XJ6yVbezgiuHwXxF8>

ANEXO 5: Sprints del proyecto de tesis

Epic N° 1: Planificación del proyecto

Fase: Planificación (Duración: 25 días)

Sprint 1: Selección del lenguaje de programación

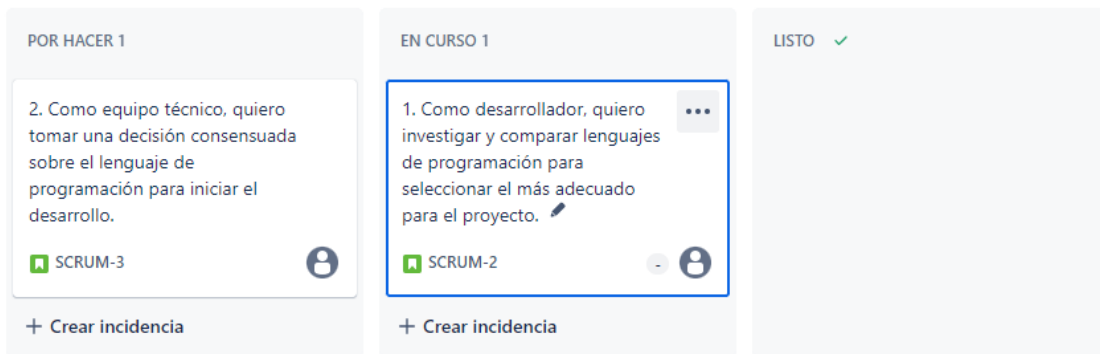
Duración: 5 días (01/04/24 - 05/04/24)

Historias de Usuario:

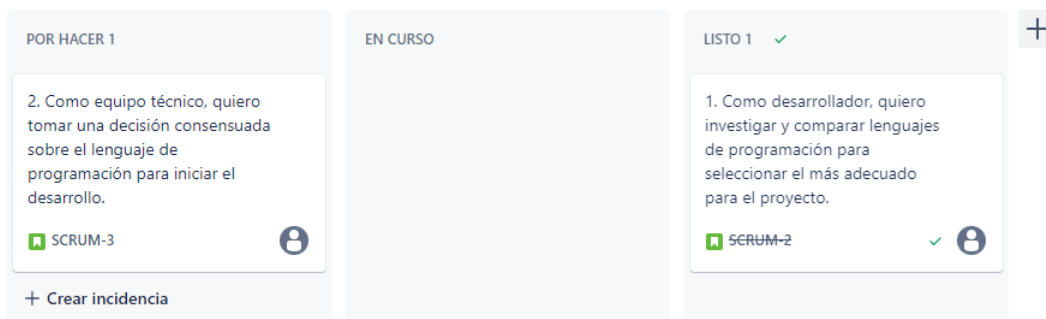
Como desarrollador, quiero investigar y comparar lenguajes de programación para seleccionar el más adecuado para el proyecto.

Como equipo técnico, quiero tomar una decisión consensuada sobre el lenguaje de programación para iniciar el desarrollo.

Durante el Sprint 1, realizado entre el 01/04/24 y el 05/04/24, se llevaron a cabo daily standups donde el equipo avanzó de manera constante en la investigación y comparación de lenguajes de programación (SCRUM-2) y en la toma de decisiones sobre el lenguaje a usar para el proyecto (SCRUM-3).



A lo largo del sprint, no se presentaron impedimentos mayores, y aunque surgieron algunas dudas respecto a la documentación de ciertos lenguajes, el equipo logró acordar los criterios de selección, finalizando la investigación y tomando una decisión consensuada.



Al cierre del sprint, todas las tareas se movieron al tablero de "Listo" y se completaron según lo planificado, lo que permitió dar cierre exitoso al sprint.



El siguiente sprint, titulado "Definición de la plataforma de desarrollo", se llevará a cabo del 08/04/24 al 15/04/24, con una duración de 6 días. En este sprint, el objetivo principal será seleccionar y definir las herramientas y tecnologías que se utilizarán para el desarrollo del proyecto, estableciendo la infraestructura base necesaria para iniciar el desarrollo de la aplicación.

Sprint 2: Definición de la plataforma de desarrollo

Duración: 6 días (08/04/24 - 15/04/24)

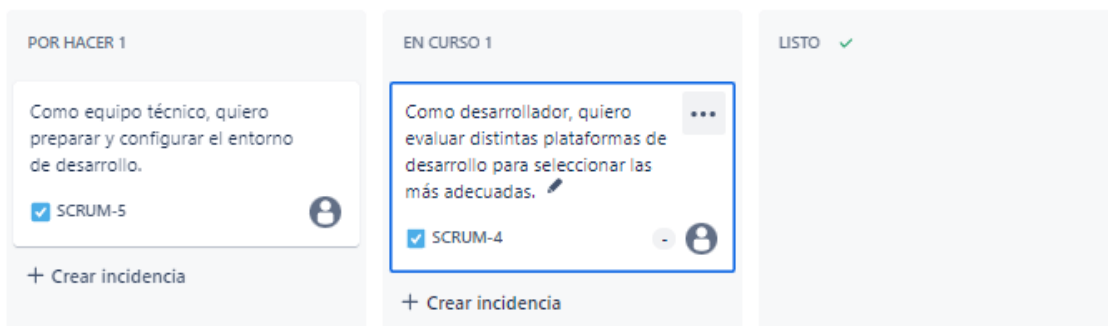
Historias de Usuario:

Como desarrollador, quiero evaluar distintas plataformas de desarrollo para seleccionar la más adecuada para el proyecto.

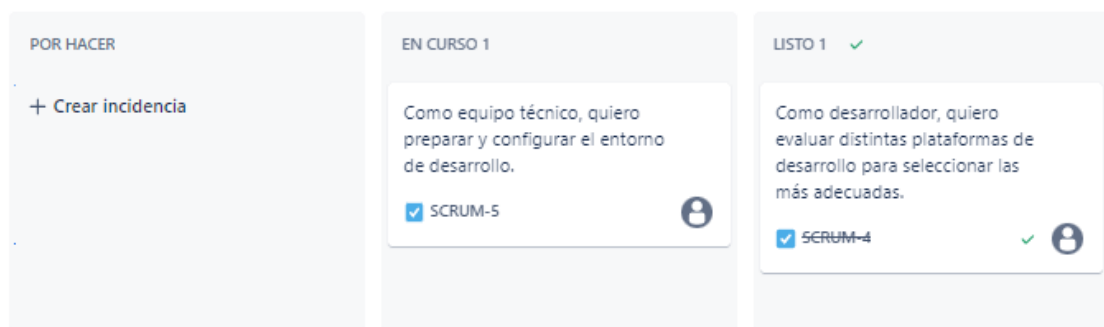
Como equipo técnico, quiero preparar y configurar el entorno de desarrollo para iniciar el trabajo de manera eficiente.

Durante el Sprint 2, que abarcó del 08/04/24 al 15/04/24, se lograron importantes avances.

El 09/04/24, el desarrollador inició la evaluación de las plataformas de desarrollo (SCRUM-4) y presentó los primeros resultados.



Sin embargo, el 10/04/24 aún se enfrentaban dificultades en cuanto a la documentación de una de las plataformas evaluadas, lo que generó un pequeño retraso en la selección. Para el 12/04/24, se superaron esos inconvenientes, y la tarea fue movida al tablero "Listo" tras la selección de la plataforma más adecuada. Paralelamente, el equipo técnico comenzó la configuración del entorno de desarrollo (SCRUM-5) el 10/04/24, encontrando algunos problemas menores en la integración de herramientas el 13/04/24.



No obstante, lograron resolver los contratiempos y el 14/04/24 la tarea fue completada y movida al tablero "Listo". Finalmente, el 15/04/24, ambas tareas estaban terminadas, lo que permitió cerrar exitosamente el sprint en la fecha planificada.



Con estas tareas completadas se dio cierre al Sprint 2.



El siguiente Sprint, titulado "Establecimiento de técnicas y herramientas", tuvo lugar del 16/04/24 al 23/04/24, con una duración de 7 días. En este sprint, el equipo se enfocó en definir las metodologías ágiles y el uso de prácticas como CI/CD para optimizar el desarrollo del proyecto. Además, se buscó identificar y configurar las herramientas clave que apoyarían el proceso de desarrollo, asegurando una integración eficiente y la automatización de tareas clave dentro del proyecto.

Sprint 3: Establecimiento de técnicas y herramientas

Duración: 7 días (16/04/24 - 23/04/24)

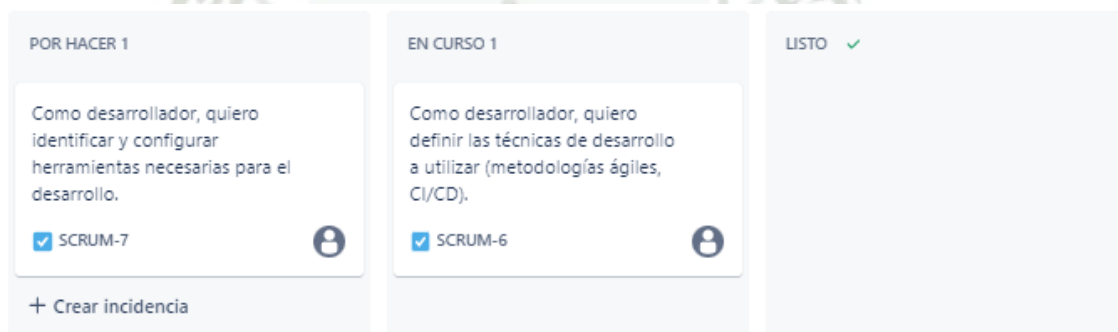
Historias de usuario:

Como desarrollador, quiero definir las técnicas de desarrollo a utilizar, como metodologías ágiles y CI/CD, para asegurar un flujo de trabajo eficiente.

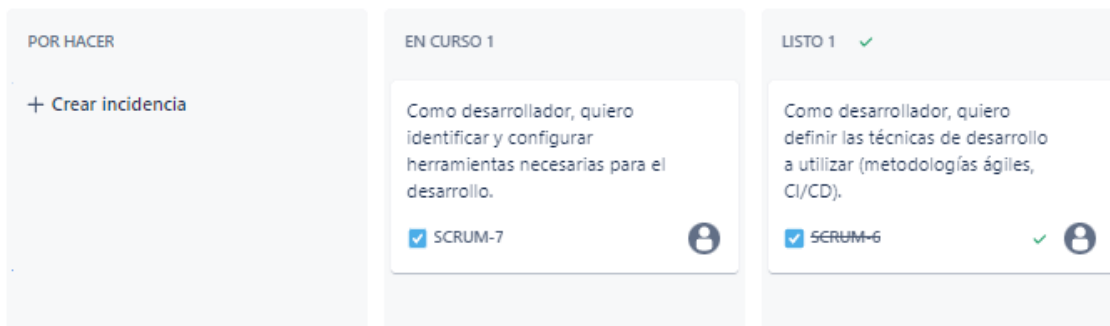
Como desarrollador, quiero identificar y configurar las herramientas necesarias para el desarrollo, garantizando una correcta integración y automatización de procesos.

Durante el Sprint 3, que tuvo lugar del 16/04/24 al 23/04/24, las tareas avanzaron de forma gradual en el tablero.

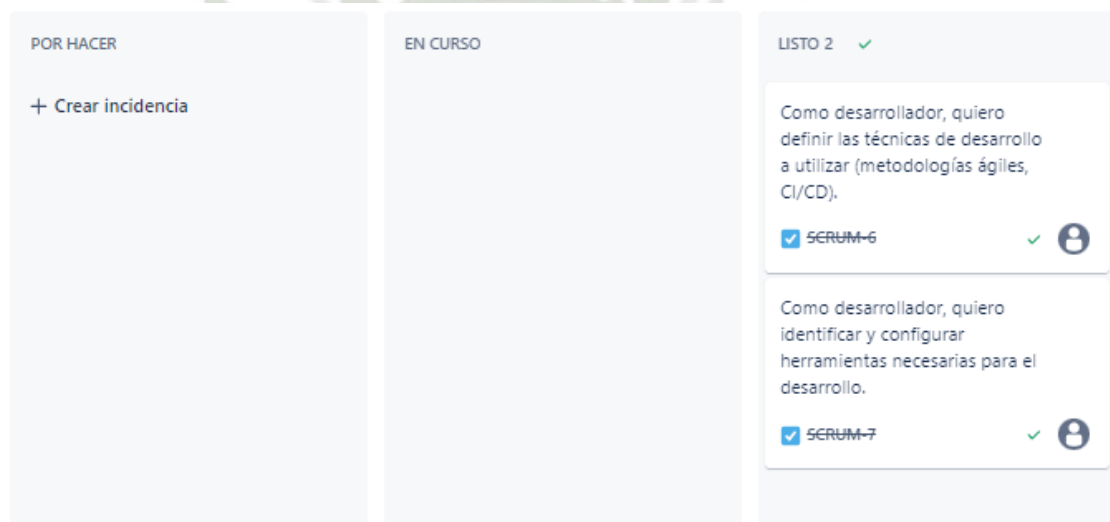
El 17/04/24, se inició la definición de las técnicas de desarrollo a utilizar (SCRUM-6), y para el 18/04/24 ya se había identificado la metodología ágil adecuada y las prácticas de CI/CD, lo que permitió mover esta tarea al estado "Listo".



Paralelamente, el 19/04/24, se comenzó la identificación y configuración de las herramientas necesarias para el desarrollo (SCRUM-7), y aunque se enfrentaron algunas dificultades en la integración de las herramientas el 21/04/24, estas se resolvieron rápidamente.



El 22/04/24, la tarea de configuración fue completada y movida al estado "Listo". Para el 23/04/24, ambas tareas habían sido finalizadas satisfactoriamente.



Al culminar estas tareas se pudo cerrar el sprint de manera exitosa, cumpliendo con los objetivos establecidos.



El Sprint 4, que se llevó a cabo del 25/04/24 al 03/05/24, tuvo como objetivo principal definir los requisitos del sistema. Se realizaron talleres con el cliente para definir las funcionalidades que el sistema debería tener, así como los flujos de trabajo y las interacciones del usuario. El equipo de desarrollo, por su parte, se enfocó en los requisitos no funcionales, como la seguridad, el rendimiento y la disponibilidad, estableciendo métricas y objetivos para garantizar la calidad del producto final. Como resultado, se crearon documentos detallados de requisitos funcionales y no funcionales, así como un prototipo de la interfaz de usuario para validar la usabilidad con el cliente.

Sprint 4: Especificación de requisitos

Duración: 7 días

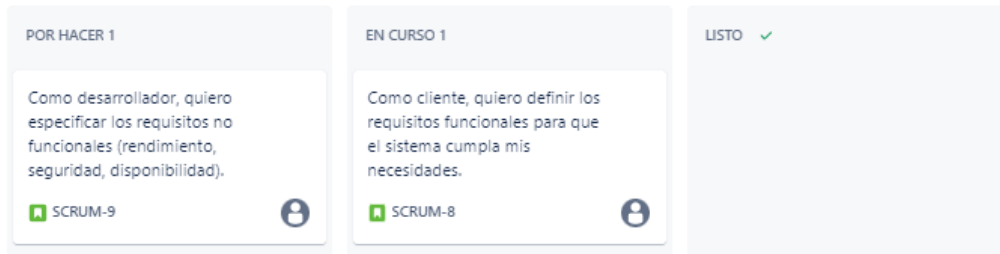
Fechas: 25/04/2024 - 03/05/2024

Historias de usuario:

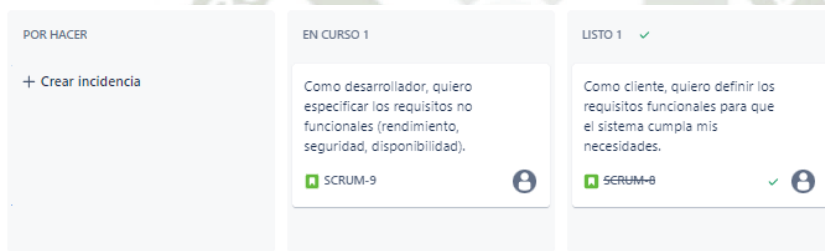
Como cliente, quiero definir los requisitos funcionales para que el sistema cumpla mis necesidades.

Como desarrollador, quiero especificar los requisitos no funcionales (rendimiento, seguridad, disponibilidad) para asegurar la calidad del sistema.

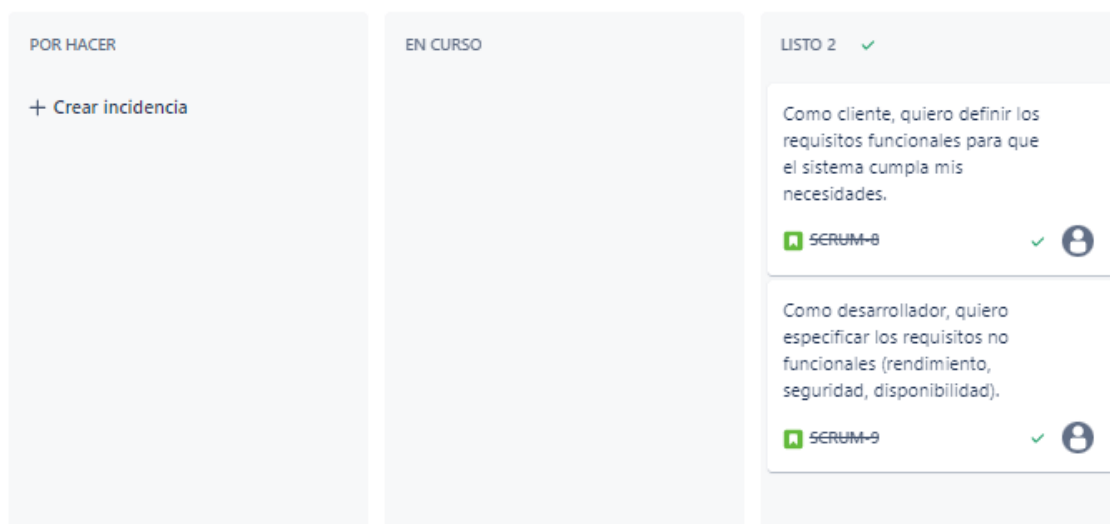
Durante el Sprint 4, que tuvo lugar del 25/04/2024 al 03/05/2024, el equipo comenzó con una reunión de planificación para asignar y detallar las historias de usuario SCRUM-8 y SCRUM-9, relacionadas con los requisitos funcionales y no funcionales del sistema.



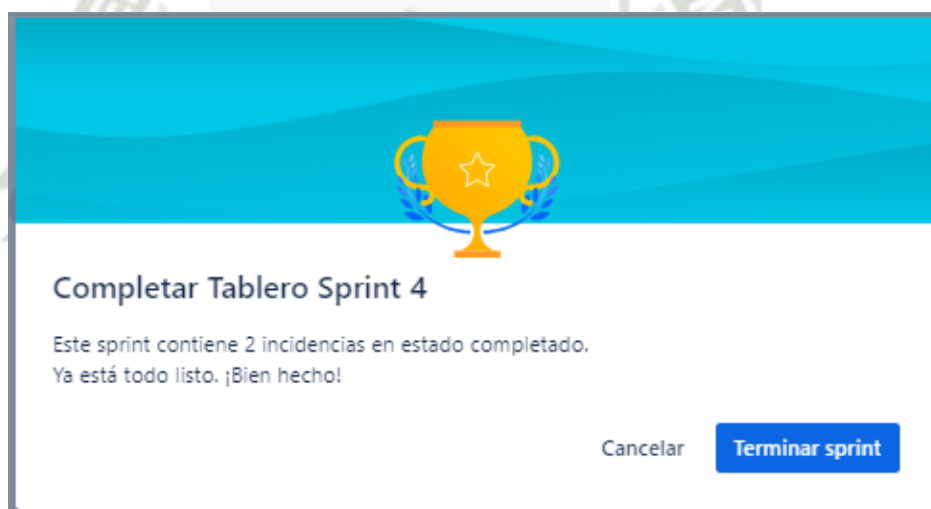
El 26/04 se inició el trabajo en SCRUM-8, avanzando en la definición de los requisitos funcionales.



El 28/04, el equipo se encontró con dificultades para precisar algunos requisitos de rendimiento en SCRUM-9 debido a limitaciones técnicas, lo cual requirió la intervención de un experto en infraestructura el 29/04.



Para el 01/05, ambos requisitos estaban prácticamente listos, y se realizaron revisiones y ajustes finales.



El sprint cerró exitosamente el 03/05 con ambas historias completadas, aunque se identificaron algunas mejoras que se considerarán en el siguiente sprint.

Epic N° 2: Análisis del sistema

Sprint 5: Definición del modelo del sistema Web

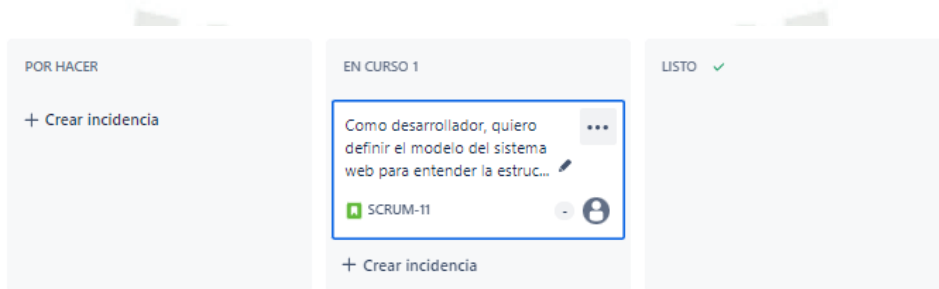
Duración: 5 días

Fechas: 06/05/2024 - 10/05/2024

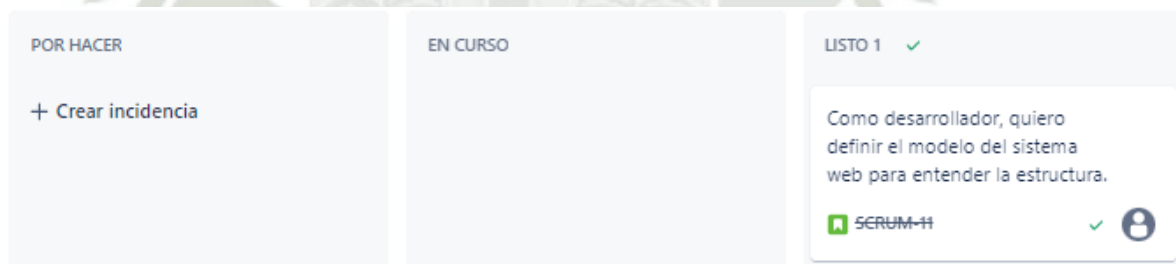
Historia de usuario:

Como desarrollador, quiero definir el modelo del sistema web para entender la estructura.

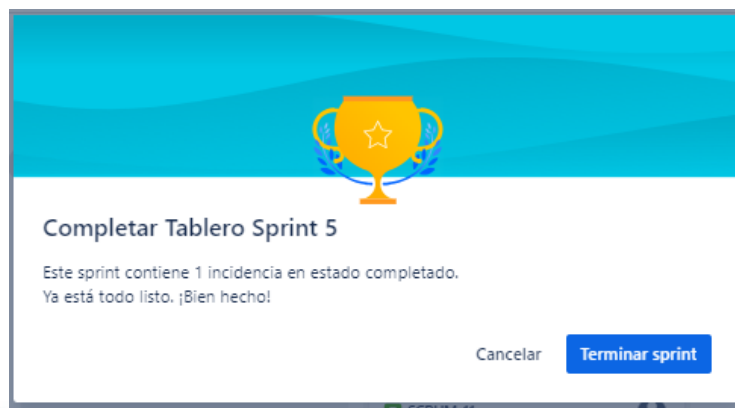
Durante el Sprint 5, que se llevó a cabo del 06/05/2024 al 10/05/2024, el equipo se enfocó en la definición del modelo del sistema web, comenzando con una reunión inicial para discutir la historia de usuario SCRUM-11.



A lo largo del sprint, se lograron avances significativos en la estructura del sistema; sin embargo, el 08/05, el equipo enfrentó dificultades para integrar ciertos requisitos técnicos que no se habían considerado previamente.



Esto llevó a la necesidad de realizar sesiones de trabajo adicionales con expertos en diseño para abordar estos desafíos. Para el 09/05, se completó la documentación del modelo, y se realizaron revisiones finales antes de que el sprint concluyera con éxito el 10/05, entregando un modelo que cumplía con las expectativas y requisitos establecidos.



Esto dio cierre al Sprint 5 de forma satisfactoria.

Sprint 6: Determinación de métodos y herramientas

Duración: 4 días

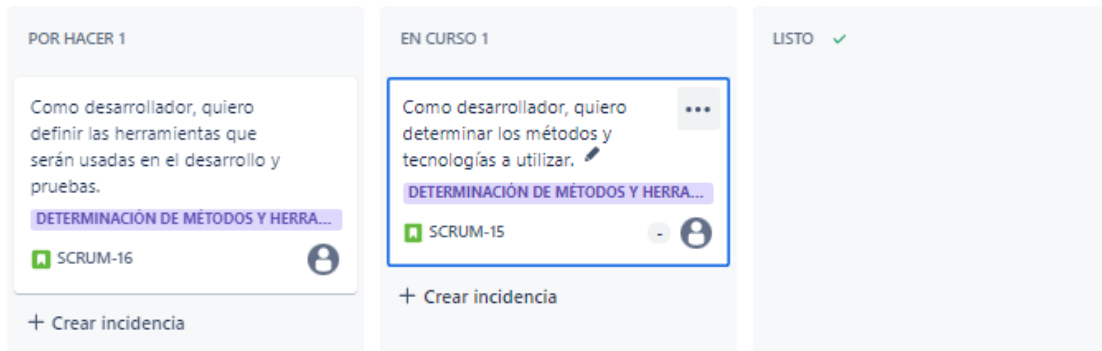
Fechas: 13/05/2024 - 16/05/2024

Historias de usuario:

Como desarrollador, quiero determinar los métodos y tecnologías a utilizar.

Como desarrollador, quiero definir las herramientas que serán usadas en el desarrollo y pruebas.

Durante el Sprint 6, que se llevó a cabo del 13/05/2024 al 16/05/2024, el equipo se centró en la determinación de métodos y herramientas necesarias para el desarrollo del sistema.



Se inició el sprint con una reunión de planificación en la que se discutieron las historias de usuario relacionadas con la elección de métodos y tecnologías, así como las herramientas que se utilizarían en el desarrollo y las pruebas.



A lo largo del sprint, el equipo avanzó en la evaluación de varias tecnologías, pero el 15/05 se presentaron desafíos al intentar integrar nuevas herramientas que no se alineaban con los requisitos técnicos existentes. Esto llevó a la necesidad de realizar sesiones de análisis adicional para evaluar las mejores opciones. Para el 16/05, se lograron definir y documentar los métodos y herramientas seleccionados, lo que permitió que el sprint concluyera exitosamente



Esto dio cierre al Sprint 6 de manera satisfactoria, con una clara dirección para el desarrollo futuro.

Epic N° 3: Diseño del sistema

Sprint 7: Construcción de casos de uso

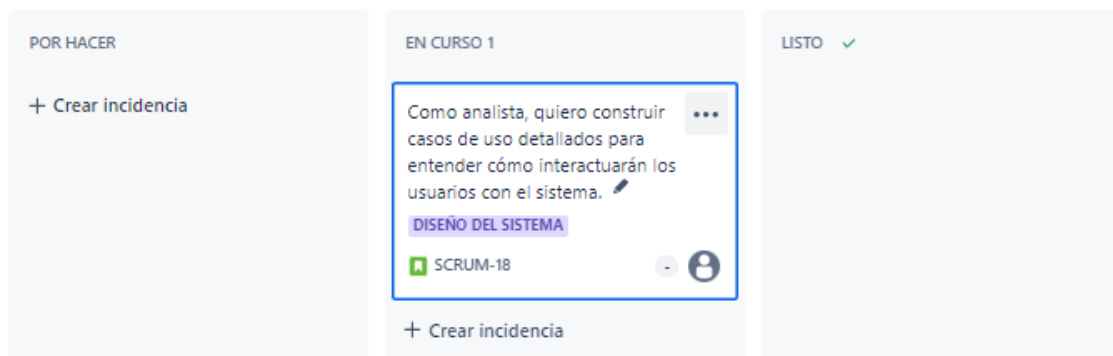
Duración: 7 días

Fechas: 17/05/2024 - 23/05/2024

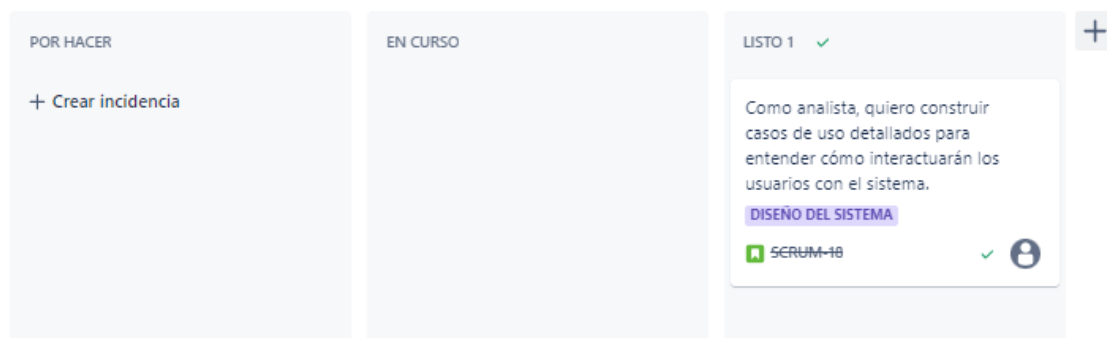
Historia de usuario:

Como analista, quiero construir casos de uso detallados para entender cómo interactuarán los usuarios con el sistema.

Durante el Sprint 7, que se llevó a cabo del 17/05/2024 al 23/05/2024, el equipo enfocó sus esfuerzos en elaborar casos de uso detallados, iniciando con una sesión de trabajo para analizar la historia de usuario sobre la interacción de los usuarios con la plataforma



A lo largo del sprint, se lograron avances significativos en la identificación y documentación de diferentes escenarios de uso; sin embargo, el 20/05, el equipo enfrentó dificultades para detallar algunos casos que requerían información adicional de otros departamentos.



Esto llevó a la necesidad de coordinar reuniones con los usuarios finales para obtener claridad sobre sus necesidades y expectativas. Para el 22/05, se completaron los casos de uso, y se realizaron revisiones finales para asegurar que todos los aspectos fueran considerados. El sprint concluyó exitosamente el 23/05, entregando una serie de casos de uso que reflejaban adecuadamente cómo los usuarios interactuarían con el sistema.



Esto dio cierre al Sprint 7 de forma satisfactoria.

Sprint 8: Creación del diagrama de clases

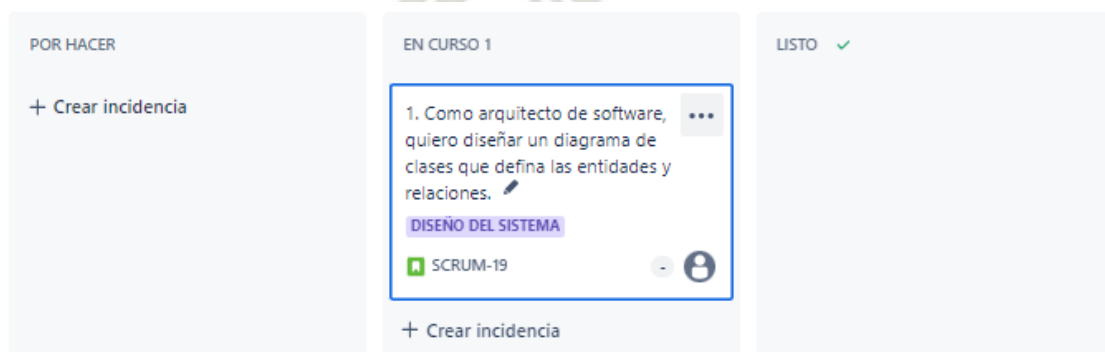
Duración: 8 días

Fechas: 28/05/2024 - 06/06/2024

Historia de usuario:

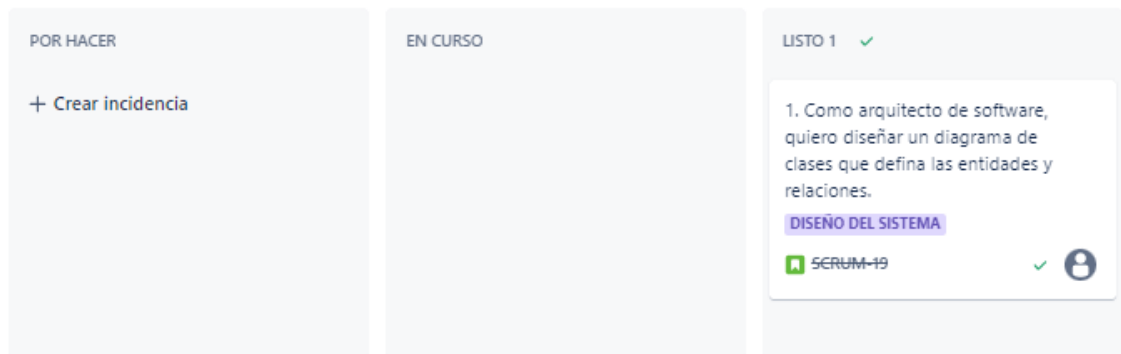
Como arquitecto de software, quiero diseñar un diagrama de clases que defina las entidades y relaciones.

Durante el Sprint 8, que se llevó a cabo del 28/05/2024 al 06/06/2024, el equipo se enfocó en la creación del diagrama de clases, comenzando con una reunión inicial para discutir la historia de usuario relacionada con el diseño de las entidades y sus relaciones.



A lo largo del sprint, se lograron avances significativos en la representación visual de las entidades del sistema; sin embargo, el 02/06, el equipo enfrentó dificultades al intentar

definir algunas relaciones complejas que requerían información adicional sobre los procesos del sistema.



Esto llevó a la necesidad de realizar sesiones de colaboración con otros desarrolladores para obtener claridad sobre los requisitos. Para el 05/06, se completó el diagrama de clases y se llevaron a cabo revisiones finales para asegurar que reflejara correctamente la estructura del sistema.



Esto dio cierre al Sprint 8 de forma satisfactoria.

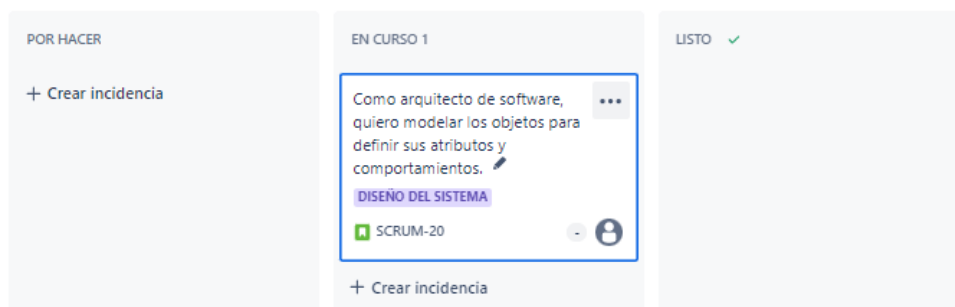
Sprint 9: Modelado de objetos (5 días)

Fechas: 07/06/24 - 13/06/24

Historias de usuario:

Como arquitecto de software, quiero modelar los objetos para definir sus atributos y comportamientos.

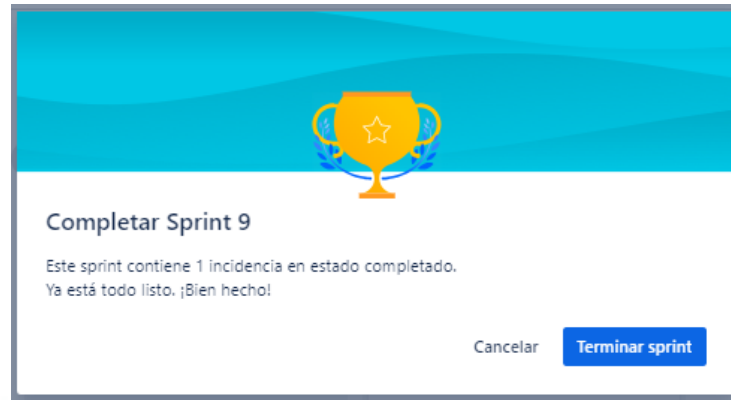
Durante el Sprint 9, el equipo se centró en el modelado de objetos para el sistema, partiendo de una sesión inicial el 07/06 para discutir la historia de usuario SCRUM-20.



A lo largo del sprint, se avanzó en la creación de los objetos clave del sistema, definiendo tanto sus atributos como sus comportamientos. Sin embargo, el 09/06, el equipo encontró desafíos al intentar modelar ciertas relaciones complejas entre los objetos, lo que no estaba previsto en las fases anteriores del proyecto.



Esto llevó a la necesidad de realizar reuniones adicionales con el equipo de arquitectura de software para resolver estas cuestiones. Para el 11/06, el equipo logró documentar completamente el modelo de objetos, realizando las revisiones finales al día siguiente.



El Sprint concluyó exitosamente el 13/06, con la entrega de un modelo de objetos que cumplía con los requisitos y expectativas del cliente.

Sprint 10: Diseño de la interfaz gráfica (5 días)

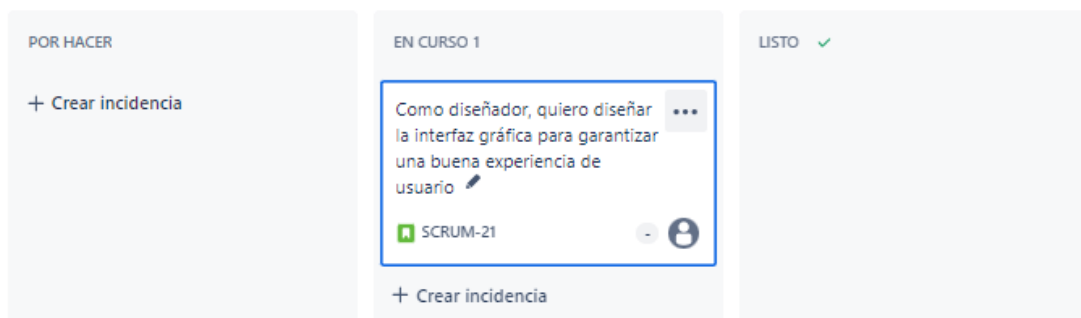
Fechas: 14/06/24 - 20/06/24

Historias de usuario:

Como diseñador, quiero diseñar la interfaz gráfica para garantizar una buena experiencia de usuario.

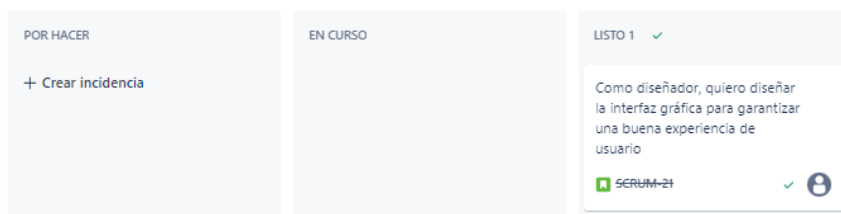
Durante el Sprint 10, el equipo se enfocó en el diseño de la interfaz gráfica del sistema.

El sprint comenzó con una reunión inicial el 14/06, donde se discutió la historia de usuario SCRUM-21.



A lo largo del sprint, se avanzó en la creación de los componentes gráficos clave, enfocándose en garantizar una experiencia de usuario intuitiva y visualmente atractiva. Sin

embargo, el 16/06, el equipo enfrentó dificultades para implementar ciertos elementos de diseño que no se habían previsto inicialmente.



Para abordar estos problemas, se organizaron sesiones adicionales con expertos en experiencia de usuario (UX) y diseño visual. El 18/06, el equipo finalizó la primera versión de la interfaz y completó la documentación necesaria, lo que permitió realizar revisiones finales el 19/06.



El Sprint concluyó exitosamente el 20/06, entregando una interfaz gráfica que cumplía con los requisitos y expectativas, cerrando así el Sprint 10 de forma satisfactoria.

Epic N°4: Implementación del sistema

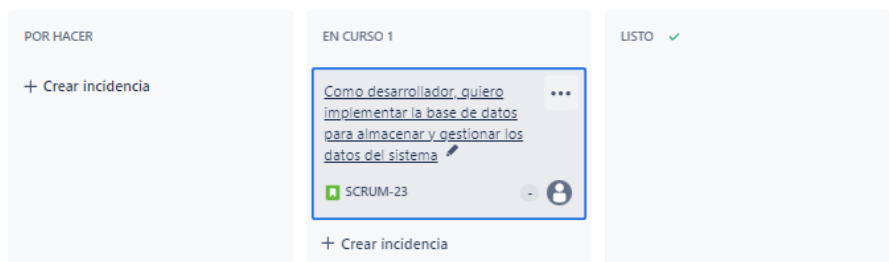
Sprint 11: Implementación de la base de datos (5 días)

Fechas: 21/06/24 - 27/06/24

Historias de usuario:

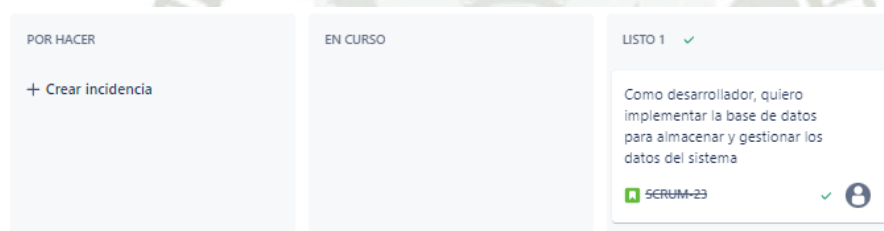
Como desarrollador, quiero implementar la base de datos para almacenar y gestionar los datos del sistema.

Durante el Sprint 11, el equipo se centró en la implementación de la base de datos para el sistema.



Fuente: Elaboración Propia

El sprint comenzó con una reunión inicial el 21/06 para discutir la historia de usuario SCRUM-23.



A lo largo del sprint, se realizaron avances importantes en la configuración y estructuración de la base de datos. No obstante, el 23/06, el equipo enfrentó dificultades técnicas para integrar algunos requerimientos de seguridad que no se habían considerado previamente.



Esto derivó en la necesidad de organizar sesiones adicionales con expertos en bases de datos y seguridad informática para resolver estas cuestiones. El 25/06, la estructura de la

base de datos fue completada, y se inició la documentación del proceso. Finalmente, el 26/06 se llevaron a cabo las revisiones finales.

El Sprint concluyó exitosamente el 27/06, entregando una base de datos funcional que cumplía con los requisitos establecidos, cerrando el Sprint 11 de manera satisfactoria.

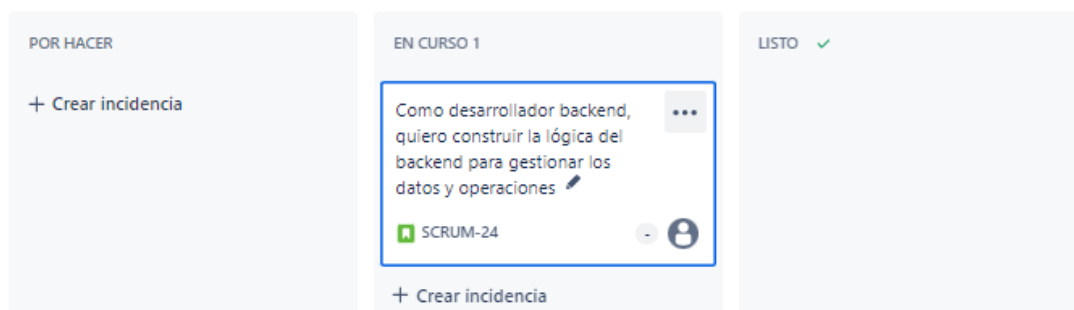
Sprint 12: Desarrollo del backend (35 días)

Fechas: 28/06/24 - 15/08/24

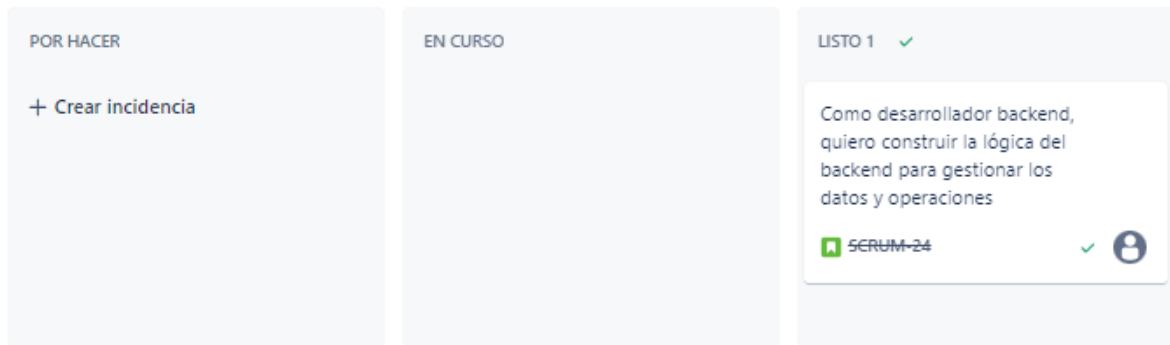
Historias de usuario:

Como desarrollador backend, quiero construir la lógica del backend para gestionar los datos y operaciones.

Durante el Sprint 12, el equipo se enfocó en el desarrollo del backend del sistema. El sprint comenzó con una reunión inicial el 28/06, en la que se discutió la historia de usuario SCRUM-24.



A lo largo del sprint, se lograron avances significativos en la construcción de la lógica del backend, asegurando que las operaciones y la gestión de datos se ejecutaran de manera eficiente. Sin embargo, el 18/07, el equipo enfrentó desafíos técnicos inesperados relacionados con la integración de APIs externas y la validación de datos, lo cual no había sido contemplado previamente.



Para resolver estos problemas, se organizaron sesiones de trabajo adicionales con ingenieros especializados en integración y seguridad de datos. Para el 05/08, el desarrollo principal del backend estaba completo, y el equipo comenzó la fase de pruebas y optimización.



El 13/08 se realizaron las revisiones finales, y el sprint concluyó con éxito el 15/08, entregando un backend completamente funcional que cumplía con los requisitos y expectativas establecidos, cerrando así el Sprint 12 de manera satisfactoria.

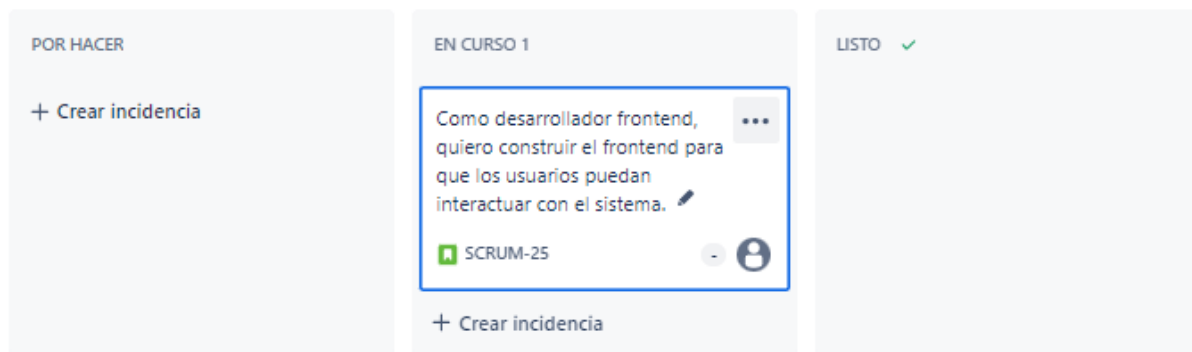
Sprint 13 - Sprint 25: Desarrollo del frontend (35 días)

Fechas: 16/08/24 - 03/10/24

Historias de usuario:

Como desarrollador frontend, quiero construir el frontend para que los usuarios puedan interactuar con el sistema.

Durante el Sprint 13, el equipo se enfocó en el desarrollo del frontend del sistema. El sprint comenzó con una reunión inicial el 16/08, donde se discutió la historia de usuario SCRUM-25.



A lo largo del sprint, se lograron avances importantes en la construcción de la interfaz de usuario, asegurando que las interacciones fueran intuitivas y atractivas. Sin embargo, el 01/09, el equipo enfrentó desafíos al implementar ciertas funcionalidades que requerían una integración más profunda con el backend, lo que no se había previsto en las fases anteriores.



Para abordar estos problemas, se organizaron sesiones de trabajo adicionales con desarrolladores de backend y diseñadores UX. Para el 25/09, se completó la mayor parte del desarrollo del frontend, y el equipo comenzó a realizar pruebas de usabilidad y optimización de rendimiento.



Las revisiones finales se llevaron a cabo el 02/10, y el sprint concluyó exitosamente el 03/10, entregando un frontend funcional que cumplía con los requisitos y expectativas del cliente, cerrando así el Sprint 13 de manera satisfactoria.

Sprint 14: Implementación de pruebas (10 días)

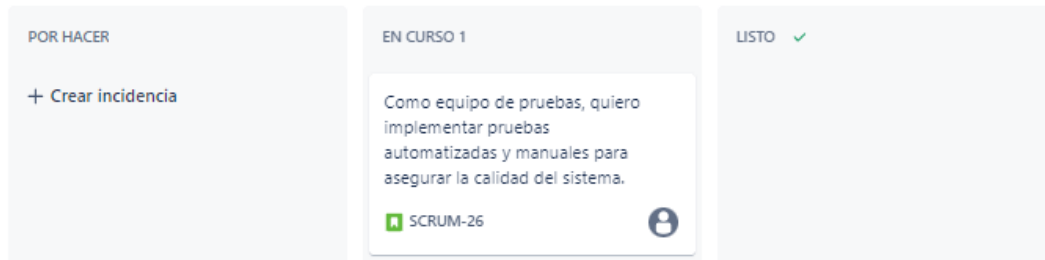
Fechas: 04/10/24 - 17/10/24

Historias de usuario:

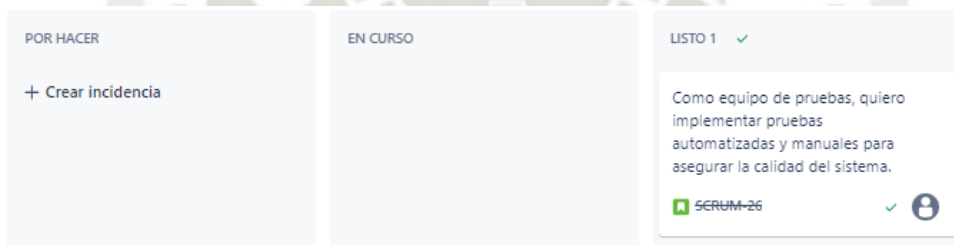
Como equipo de pruebas, quiero implementar pruebas automatizadas y manuales para asegurar la calidad del sistema.

Durante el Sprint 14, el equipo se centró en la implementación de pruebas para el sistema.

El sprint comenzó con una reunión inicial el 04/10, donde se discutió la historia de usuario SCRUM-26.



A lo largo del sprint, se lograron avances significativos en la creación de pruebas automatizadas y manuales para garantizar la calidad del sistema. Sin embargo, el 10/10, el equipo enfrentó dificultades al intentar cubrir ciertos casos de prueba que no habían sido considerados en las fases de desarrollo previas.



Esto llevó a la necesidad de realizar sesiones de trabajo adicionales con los desarrolladores para identificar y documentar estos casos no previstos. Para el 15/10, se completó la implementación de las pruebas, y el equipo llevó a cabo revisiones finales para asegurar que todos los aspectos del sistema fueran evaluados.



Epic N° 5: Pruebas del sistema

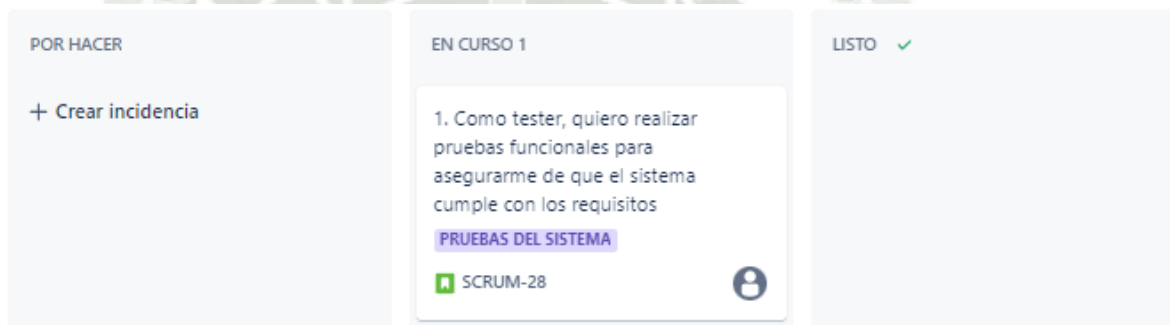
Sprint 15: Pruebas funcionales (10 días)

Fechas: 18/10/24 - 31/10/24

Historias de usuario:

Como tester, quiero realizar pruebas funcionales para asegurarme de que el sistema cumple con los requisitos.

Entre el 7 y el 11 de octubre, otro equipo apoyó en el desarrollo de las pruebas, permitiendo lograr avances significativos en la ejecución de pruebas funcionales para verificar que el sistema cumpliera con los requisitos establecidos. Sin embargo, el 24/10, el equipo principal enfrentó dificultades al encontrar algunos errores en la funcionalidad que no se habían detectado en sprints anteriores.



Esto llevó a la necesidad de realizar sesiones de trabajo adicionales con los desarrolladores para solucionar estos problemas. Para el 29/10, se completaron todas las pruebas funcionales y se documentaron los resultados.



Las revisiones finales se llevaron a cabo el 30/10, y el sprint concluyó exitosamente el 31/10, entregando un informe de pruebas que confirmaba que el sistema cumplía con los requisitos funcionales, cerrando así el Sprint 15 de manera satisfactoria.



Las revisiones finales se llevaron a cabo el 29/10, y el sprint concluyó exitosamente el 30/10, entregando un informe de pruebas que confirmaba que el sistema cumplía con los requisitos funcionales, cerrando así el Sprint 15 de manera satisfactoria.

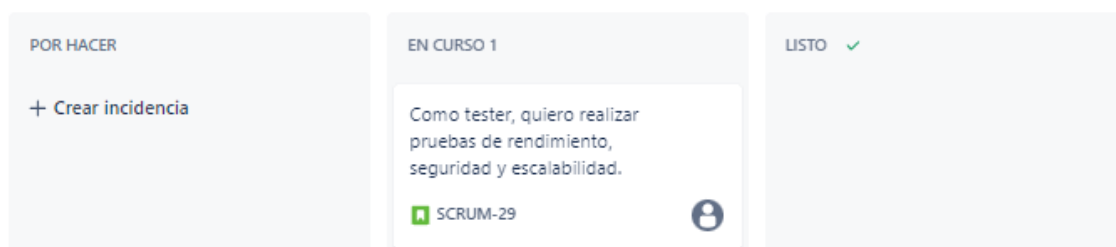
Sprint 16: Pruebas no funcionales (10 días)

Fechas: 01/11/24 - 14/11/24

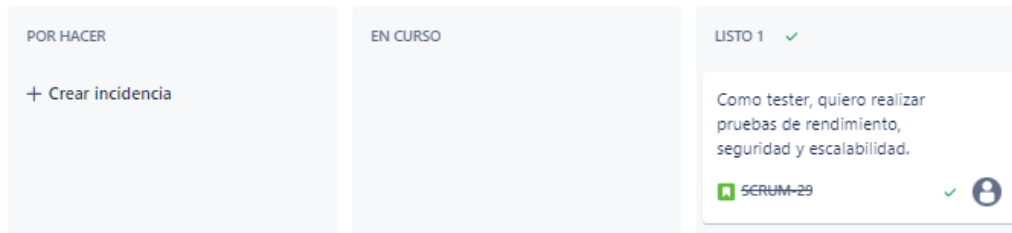
Historias de usuario:

Como tester, quiero realizar pruebas de rendimiento, seguridad y escalabilidad.

Gracias a la optimización en la planificación de los sprints anteriores, fue posible adelantar el Sprint 16, originalmente programado para noviembre, a las semanas del 14 al 27 de octubre. Este adelanto permitió al equipo concentrarse en pruebas no funcionales para garantizar que el sistema no solo cumpla con los requisitos funcionales, sino también con criterios de rendimiento, seguridad y escalabilidad.



A través de una gestión eficiente de recursos y colaboración con otros equipos, se logró una preparación sólida para abordar estas pruebas con antelación.



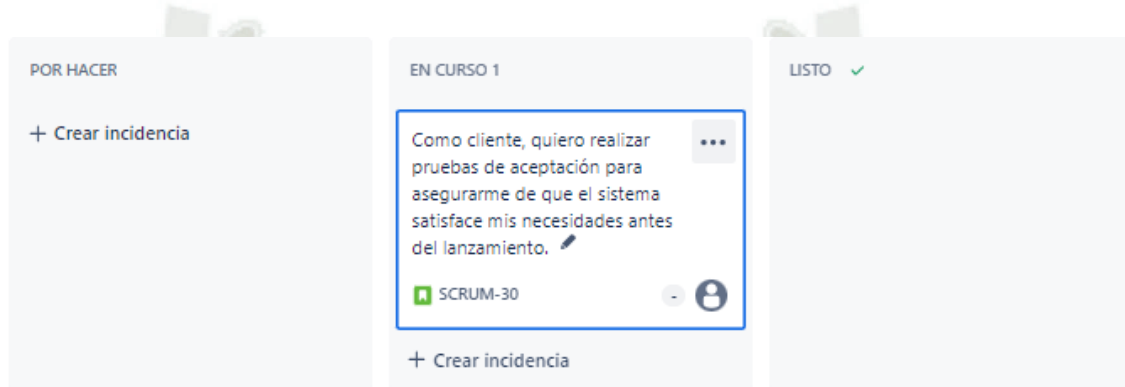
Durante el Sprint 16, el equipo de pruebas se dedicó a evaluar el sistema en condiciones de carga, verificando su capacidad de respuesta y la estabilidad del rendimiento bajo escenarios intensos. También se llevaron a cabo pruebas de seguridad en distintos módulos críticos para detectar posibles vulnerabilidades. La participación de un equipo especializado en seguridad durante este proceso fue esencial, ya que permitió identificar áreas que requerían ajustes adicionales para fortalecer la integridad del sistema.



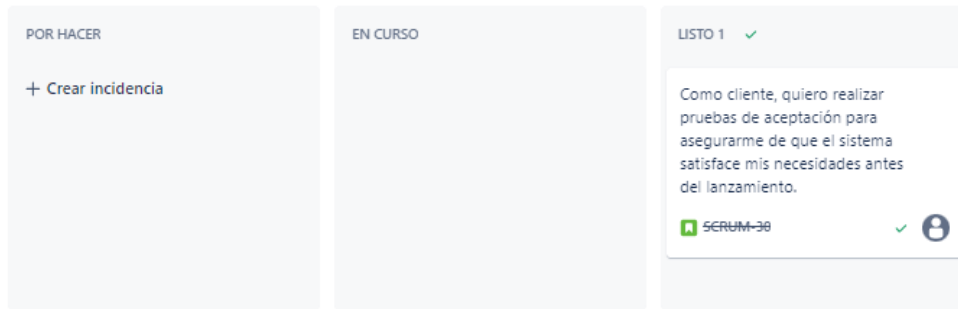
Este adelanto en el calendario no solo favoreció la optimización de recursos, sino que también dio al equipo de desarrollo un tiempo extra para realizar mejoras basadas en los resultados obtenidos. Para el 27/10, se había completado la fase inicial de pruebas, y se compartió un informe preliminar con los responsables de los módulos afectados, proporcionando así una ventaja significativa para afinar el sistema antes de las fechas originalmente programadas.

Sprint 17: Pruebas de Aceptación (6 días)**Fechas:** 15/11/24 - 22/11/24**Historia de usuario:**

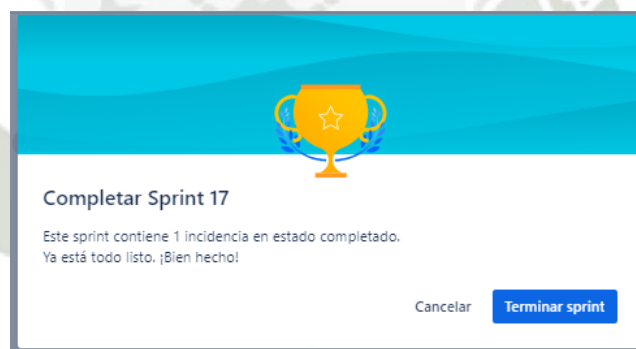
Como cliente, quiero realizar pruebas de aceptación para asegurarme de que el sistema satisface mis necesidades antes del lanzamiento.



Dado el progreso anticipado en sprints previos, fue posible adelantar el Sprint 17, originalmente programado para noviembre, al período del 21 al 27 de octubre. Este cambio permitió al cliente involucrarse antes de lo previsto en el proceso de pruebas de aceptación, verificando directamente que el sistema respondiera a sus necesidades y expectativas antes del lanzamiento oficial. Esta colaboración temprana no solo facilitó una retroalimentación más oportuna, sino que también ayudó al equipo a detectar y ajustar detalles específicos en función de las observaciones del cliente.



Durante esta fase, el cliente participó activamente en la validación de las funcionalidades clave del sistema, realizando pruebas de aceptación sobre varios módulos importantes. Este enfoque permitió identificar ajustes menores y confirmar que los criterios establecidos en las historias de usuario se cumplían en su totalidad. Además, el equipo de desarrollo se mantuvo disponible para realizar las modificaciones solicitadas de manera ágil, logrando una comunicación efectiva y una respuesta rápida a las observaciones planteadas.



Este adelanto de las pruebas de aceptación ofreció una ventaja significativa, ya que al cierre del 27 de octubre se contaba con un sistema casi listo para el lanzamiento. La validación previa permitió que cualquier retoque final se pudiera realizar sin presión, y el cliente manifestó satisfacción al ver que el sistema cumplía con sus expectativas antes de la fecha de entrega original.