



FACULTAD DE TECNOLOGÍAS DE LA INFORMACIÓN Y LA COMUNICACIÓN
CARRERA: INGENIERÍA EN SOFTWARE

**DESARROLLO DE UN SISTEMA WEB DE BIENESTAR UNIVERSITARIO
PARA LA GESTIÓN DOCUMENTAL DEL ÁREA DE MEDICINA EN UNA
INSTITUCIÓN DE EDUCACIÓN SUPERIOR**

Trabajo de Integración Curricular para la obtención del Título de Ingeniero de Software

Autor (a):
Hector Steveen Ordoñez Chamba
Erick Fernando García Guerrero

Tutor (a):
Harry Carpio, Msc.

Quito, Ecuador
mayo, 2024

DECLARACIÓN DE AUTORÍA Y AUTORIZACIÓN PARA LA DIFUSIÓN DEL TRABAJO DE INTEGRACIÓN CURRICULAR

1. Yo, **Hector Steveen Ordoñez Chamba y Erick Fernando García Guerrero**, declaró en forma libre y voluntaria, que los criterios emitidos en el presente Trabajo de Integración Curricular, titulado: "**DESARROLLO DE UN SISTEMA WEB DE BIENESTAR UNIVERSITARIO PARA LA GESTIÓN DOCUMENTAL DEL ÁREA DE MEDICINA EN UNA INSTITUCIÓN DE EDUCACIÓN SUPERIOR**", previo a la obtención del título profesional de **Ingeniería en Software**, así como también los contenidos, ideas, análisis, conclusiones y propuestas son exclusiva responsabilidad de mi persona, como autor/a.
2. Declaro, igualmente, tener pleno conocimiento de la obligación que tiene la Universidad Iberoamericana del Ecuador, de conformidad con el artículo 144 de la Ley Orgánica de Educación Superior, de entregar a la SENESCYT, en formato digital una copia del referido Trabajo de Integración Curricular para que sea integrado al Sistema Nacional de Información de la Educación Superior del Ecuador para su difusión pública, respetando los derechos de autor.
3. Autorizó, finalmente, a la Universidad Iberoamericana del Ecuador a difundir a través del sitio web de la Biblioteca de la UNIB.E (Repositorio Digital Institucional), el referido Trabajo de Integración Curricular, respetando las políticas de propiedad intelectual de la Universidad Iberoamericana del Ecuador.

Quito, DM., a los 14 días del mes de febrero de 2025.



HECTOR STEVEEN ORDOÑEZ CHAMBA
1755516802



ERICK FERNANDO GARCIA GUERRERO
0401692249

AUTORIZACIÓN DE PRESENTACIÓN FINAL DEL TRABAJO DE INTEGRACIÓN CURRICULAR POR PARTE DEL TUTOR

Msc. Sandino Jaramillo Cartagena

Director(a) de la Carrera de Ingeniería en Software

Presente. -

Yo, **Harry Alberto Carpio Salvatierra, Magister**, Tutor del Trabajo de Integración Curricular realizado por los estudiantes **Hector Steveen Ordoñez Chamba y Erick Fernando García Guerrero** de la carrera de Ingeniería en software informó haber revisado el presente documento titulado **DESARROLLO DE UN SISTEMA WEB DE BIENESTAR UNIVERSITARIO PARA LA GESTIÓN DOCUMENTAL DEL ÁREA DE MEDICINA EN UNA INSTITUCIÓN DE EDUCACIÓN SUPERIOR**, el mismo que se encuentra elaborado conforme a lo establecido en el Reglamento de Titulación y el Manual de Estilo de la Universidad Iberoamericana del Ecuador, UNIB.E de Quito, por lo tanto, autorizo la entrega del Trabajo de Integración Curricular a la Unidad de Titulación para la presentación final ante el tribunal evaluador.

Atentamente,



Msc. Harry Alberto Carpio Salvatierra

Tutor

ACTA DE APROBACIÓN DEL TRABAJO DE INTEGRACIÓN CURRICULAR

Facultad: Ciencias Administrativas, Servicios, Comunicación y Tecnología

Carrera: Software

Modalidad: Híbrida

Nivel: 3er nivel de Grado

En el Distrito Metropolitano de Quito a los veintiocho días de marzo del 2025 a las 12:00 (12H00), ante el Tribunal de Presentación Oral, se presentó el señor: **GARCIA GUERRERO ERICK FERNANDO**, titular de la cédula de ciudadanía No. 401692249 a rendir la evaluación oral del Trabajo de Integración Curricular: "**DESARROLLO DE UN SISTEMA WEB DE BIENESTAR UNIVERSITARIO PARA LA GESTIÓN DOCUMENTAL DEL ÁREA DE MEDICINA EN UNA INSTITUCIÓN DE EDUCACIÓN SUPERIOR.**", previo a la obtención del Título de Ingeniera en Software. Luego de la exposición, el referido estudiante obtiene las calificaciones que a continuación se detallan:

	Calificación
Lectura del Trabajo de Integración Curricular	9,2 /10
Evaluación Oral del Trabajo de Integración Curricular	9,1 /10
Calificación Final del Trabajo de Integración Curricular	9,2 /10

Para constancia de lo actuado, los miembros del Tribunal de Presentación Oral del Trabajo de Integración Curricular, firman el presente documento en unidad de acto, a los veintiocho días de marzo del 2025 (28-03-2025).

Mgst. Andrea Guadalupe
DIRECTOR ACADÉMICO



Mgst. Sandino Jaramillo
DIRECTOR DE LA CARRERA DE
SOFTWARE



Mgst. Harry Carpio
TUTOR

Mgst. Marcela Parra
LECTOR

ACTA DE APROBACIÓN DEL TRABAJO DE INTEGRACIÓN CURRICULAR

Facultad: Ciencias Administrativas, Servicios, Comunicación y Tecnología

Carrera: Software

Modalidad: Híbrida

Nivel: 3er nivel de Grado

En el Distrito Metropolitano de Quito a los veintiocho días de marzo del 2025 a las 12:00 (12H00), ante el Tribunal de Presentación Oral, se presentó el señor: **ORDÓÑEZ CHAMBA HECTOR STEVEEN**, titular de la cédula de ciudadanía No. 1755516802 a rendir la evaluación oral del Trabajo de Integración Curricular: "**DESARROLLO DE UN SISTEMA WEB DE BIENESTAR UNIVERSITARIO PARA LA GESTIÓN DOCUMENTAL DEL ÁREA DE MEDICINA EN UNA INSTITUCIÓN DE EDUCACIÓN SUPERIOR.**", previo a la obtención del Título de Ingeniera en Software. Luego de la exposición, el referido estudiante obtiene las calificaciones que a continuación se detallan:

	Calificación
Lectura del Trabajo de Integración Curricular	92/10
Evaluación Oral del Trabajo de Integración Curricular	91/10
Calificación Final del Trabajo de Integración Curricular	92/10

Para constancia de lo actuado, los miembros del Tribunal de Presentación Oral del Trabajo de Integración Curricular, firman el presente documento en unidad de acto, a los veintiocho días de marzo del 2025 (28-03-2025).

Mgst. Andrea Guadalupe
DIRECTOR ACADÉMICO

Mgst. Harry Carpio
TUTOR

Mgst. Sandino Jaramillo
DIRECTOR DE LA CARRERA DE SOFTWARE

Mgst. Marcela Parra
LECTOR

ÍNDICE GENERAL

DECLARACIÓN DE AUTORÍA Y AUTORIZACIÓN PARA LA DIFUSIÓN DEL TRABAJO DE INTEGRACIÓN CURRICULAR.....	ii
AUTORIZACIÓN DE PRESENTACIÓN FINAL DEL TRABAJO DE INTEGRACIÓN CURRICULAR POR PARTE DEL TUTOR.....	iii
ACTA DE APROBACIÓN DEL TRABAJO DE INTEGRACIÓN CURRICULAR.....	iv
ÍNDICE GENERAL.....	vi
LISTA DE TABLAS.....	x
LISTA DE FIGURAS.....	xi
RESUMEN.....	xiv
ABSTRACT.....	xv
INTRODUCCIÓN.....	1
CAPÍTULO I.....	2
EL PROBLEMA.....	2
Planteamiento del problema.....	2
Objetivos de la investigación.....	4
Objetivo general.....	4
Objetivos específicos.....	4
Justificación e impacto de la investigación.....	5
Alcance de la investigación.....	7
CAPÍTULO II.....	8
MARCO TEÓRICO.....	8
Antecedentes de la Investigación.....	8
Bases teóricas.....	10
Sistemas de gestión documental.....	10
Sistema web.....	12
Metodologías de desarrollo de software.....	12
Patrones de diseño.....	13
Modelo Vista Controlador (MVC)	13
Monolítico.....	14
Framework de desarrollo web.....	14

Frameworks del Lado del Cliente.....	14
React.....	14
Angular.....	15
Frameworks del Lado del Servidor.....	16
Laravel.....	16
NestJS.....	16
Lenguajes de programación.....	17
JavaScript y TypeScript:	17
PHP.....	17
Java.....	17
Sistema de gestión de base de datos.....	18
PostgreSQL.....	19
MySQL.....	19
DynamoDB.....	20
Plataformas en la nube.....	22
Firebase.....	22
AWS (Amazon Web Services)	23
Modelo de estimación.....	26
COCOMO II.....	26
Function Point Analysis (FPA)	26
Bases Legales.....	28
Sistemas de Gestión documental.....	28
Evolución de los Sistemas de Gestión Documental.....	28
Impacto de la Tecnología en la Gestión Educativa.....	28
Reformas en la Ley de Educación Superior en Ecuador.....	28
CAPÍTULO III.....	30
MARCO METODOLÓGICO.....	30
Naturaleza de la investigación.....	30
Enfoque de la investigación.....	30
Nivel de la investigación.....	30

Diseño de la investigación.....	31
Tipo de investigación.....	31
Población.....	31
Técnicas e Instrumentos de Recolección de Datos.....	32
Técnica de Recolección de Datos.....	32
Operacionalización de la Variable.....	33
Instrumentos de Recolección de Datos.....	36
Validez y Confiabilidad.....	36
Técnicas de Análisis de Datos.....	37
Metodología del producto.....	38
CAPÍTULO IV.....	40
ANÁLISIS E INTERPRETACIÓN DE LOS RESULTADOS.....	40
Requerimientos funcionales y no funcionales del sistema.....	40
Diseño del sistema web.....	40
Desarrollo del sistema web.....	40
Validación del sistema.....	41
Resultados de la Investigación.....	41
Análisis de los Resultados.....	41
Resultados de la encuesta sobre requerimientos del sistema.....	42
Resultados de la encuesta sobre la validación del sistema.....	56
Metodología.....	57
Planificación.....	57
Prototipos.....	58
Desarrollo.....	58
Validación.....	59
Planificación del Proyecto de Software.....	60
Recursos Humanos.....	60
Recursos de Software.....	60
Implementación en AWS.....	62
Estimación del Proyecto.....	62

Ruta del proyecto.....	64
Herramientas de gestión del proyecto.....	65
Sistema de gestión de versiones.....	65
Análisis del diseño.....	65
Visión y Alcance.....	65
Nombre del producto.....	65
Cliente objetivo.....	66
Funcionalidades del sistema.....	66
Glosario de términos.....	71
Modelado UML.....	74
Diagrama de Clases.....	74
Diagrama de caso de uso.....	75
Diagrama de actividades.....	79
Diagrama de estados.....	82
Base de datos Amazon DynamoDB.....	83
Diseño de interfaz de usuario.....	84
Personal Administrativo.....	85
Diseño de arquitectura.....	96
Desarrollo.....	98
Tecnologías utilizadas.....	98
Producto de software desarrollado.....	98
Pruebas de Carga y Estrés.....	99
Metodología.....	99
Parámetros de prueba.....	99
Escenario 1: 40 Usuarios Concurrentes.....	100
Escenario 2: 80 Usuarios Concurrentes.....	101
Escenario 3: 120 Usuarios Concurrentes.....	103
Optimización del Rendimiento y Uso de Recursos Computacionales.....	105
Disponibilidad del sistema.....	107
Manual de Usuario.....	108

CAPÍTULO V.....	109
CONCLUSIONES Y RECOMENDACIONES.....	109
Conclusiones.....	109
Recomendaciones.....	110
REFERENCIAS BIBLIOGRÁFICAS.....	111
ANEXOS.....	114
Anexo 1. Cuestionario para recopilación de requerimientos.....	115
Anexo 2. Validación del producto/software.....	120
Anexo 3. Validación del Instrumento por parte de los expertos.....	121
Anexo 4. Documentación para desarrolladores.....	122
Anexo 5. Documentación para usuarios.....	122
Anexo 6. Prototipos.....	122
Anexo 7. Pruebas de Carga y Estrés.....	122
Anexo 8. Link del Sistema.....	122

LISTA DE TABLAS

Tabla 1. Descripción de las características de los frameworks del lado del cliente.....	15
Tabla 2. Descripción de las características de los frameworks del lado del servidor.....	16
Tabla 3. Descripción de las características de lenguajes de programación.....	18
Tabla 4. Descripción de las características de los sistemas de gestión de base de datos.....	21
Tabla 5. Descripción de las características de plataformas en la nube.....	24
Tabla 6. Descripción de las características de los modelos de estimación.....	27
Tabla 7. Operacionalización de Variables (Levantamiento de requerimiento/requisitos)....	34
Tabla 8. Operacionalización de Variables (Validación del producto/software).....	35
Tabla 9. Validadores del instrumento de investigación.....	37
Tabla 10. Porcentaje de respuestas afirmativas y negativas de la encuesta realizada para obtener los requerimientos funcionales y no funcionales.....	54
Tabla 11. Porcentaje de respuestas afirmativas y negativas de la encuesta realizada para la validación del producto/software.	57
Tabla 12. Responsable del trabajo de titulación.....	60

Tabla 13. Detalles del Hardware usado para el desarrollo.....	60
Tabla 14. Detalles del software utilizado para el desarrollo.....	61
Tabla 15. Ruta del proyecto.....	64
Tabla 16. Historias de Usuario Autenticación.....	66
Tabla 17. Historias de usuarios de la Gestión de proyectos.....	67
Tabla 18. Historias de Gestión de periodos.....	67
Tabla 19. Historias de usuarios de la Gestión de Actividades.....	68
Tabla 20. Historias de usuarios de Reportes.....	68
Tabla 21. Historias de usuarios de la gestión del calendario.....	69
Tabla 22. Historias de usuarios de la gestión de documentos.....	69
Tabla 23. Historias de usuarios búsqueda de archivos.....	70
Tabla 24. Historias de usuarios búsqueda de archivos.....	70
Tabla 25. Historias de usuarios interfaz.....	71
Tabla 26. Glosario de términos.....	72
Tabla 27. Tabla de Requerimientos funcionales.....	73
Tabla 28. Requerimientos no funcionales.....	73
Tabla 29. Tabla de casos de uso.....	78
Tabla 30. Pruebas de carga y estrés primer escenario 40 usuarios.....	100
Tabla 31. Métricas de Rendimiento para 80 Usuarios Concurrentes.....	102
Tabla 32. Métricas de Rendimiento para 120 Usuarios Concurrentes.....	104

LISTA DE FIGURAS

Figura 1. Datos de autentificación mediante correo y contraseña.....	42
Figura 2. Datos referentes a contraseñas.....	42
Figura 3. Datos de actualización de datos del usuario.....	43
Figura 4. Datos sobre la persistencia de la sesión.....	43
Figura 5. Datos Sobre permisos para controlar documentos.....	44
Figura 6. Datos Niveles de acceso a documentos.....	44
Figura 7. Datos sobre las versiones de los documentos.....	45
Figura 8. Datos sobre seguimiento de documentos.....	45

Figura 9. Métodos para compartir documentos en el sistema.....	46
Figura 10. Métodos para registrar documentos en la base de datos.....	46
Figura 11. Métodos para registrar metadatos en la carga de archivos.....	47
Figura 12. Métodos para categorizar los documentos.....	47
Figura 13. Métodos para permitir la modificación de documentos.....	48
Figura 14. Métodos para permitir la eliminación de documentos.....	48
Figura 15. Métodos para permitir la búsqueda de documentos.....	49
Figura 16. Métodos para descargar documentos.....	49
Figura 17. Métodos para permitir validaciones de contraseñas fuertes.....	50
Figura 18. Métodos para permitir niveles de acceso para los usuarios.....	50
Figura 19. Métodos para cifrar datos sensibles.....	51
Figura 20. Métodos para que el sistema sea capaz de manejar 100 usuarios.....	51
Figura 21. Accesos a dispositivos móviles.....	52
Figura 22. Datos para obtener la facilidad de uso del sistema.....	52
Figura 23. Datos para obtener el manejo de errores.....	53
Figura 24. Funcionamiento correcto bajo pruebas.....	55
Figura 25. Funcionamiento de forma consistente y sin errores.....	55
Figura 26. Disponibilidad.....	56
Figura 27. Documentación.....	56
Figura 28. Diagrama de clases.....	75
Figura 29. Diagrama de caso de uso (Administrador).....	76
Figura 30. Diagrama de caso de uso (Personal Médico).....	77
Figura 31. Diagrama de actividades Administrador.....	80
Figura 32. Diagrama de actividades personal médico.....	81
Figura 33. Diagrama de estados.....	82
Figura 34. Base de datos Amazon DynamoDB.....	84
Figura 35. Diseño de interfaz de usuario (Personal Administrativo Login).....	86
Figura 36. Diseño de interfaz de usuario (Personal Administrativo Inicio).....	86
Figura 37. Diseño de interfaz de usuario (Personal Administrativo Perfil).....	87
Figura 38. Diseño de interfaz de usuario (Personal Administrativo Perfil Crear	

Usuarios).....	87
Figura 39. Diseño de interfaz de usuario (Personal Administrativo Proyectos).....	88
Figura 40. Diseño de interfaz de usuario (Personal Administrativo Periodos).....	88
Figura 41. Diseño de interfaz de usuario (Personal Administrativo Actividades).....	89
Figura 42. Diseño de interfaz de usuario (Personal Administrativo Crear Actividades).....	89
Figura 43. Diseño de interfaz de usuario (Personal Administrativo Visualizar Actividades).....	90
Figura 44. Diseño de interfaz de usuario (Personal Administrativo Documentos).....	90
Figura 45. Diseño de interfaz de usuario (Personal Médico Login).....	92
Figura 46. Diseño de interfaz de usuario (Personal Médico Inicio).....	92
Figura 47. Diseño de interfaz de usuario (Personal Médico Perfil).....	93
Figura 48. Diseño de interfaz de usuario (Personal Médico Proyectos).....	93
Figura 49. Diseño de interfaz de usuario (Personal Médico Periodos).....	94
Figura 50. Diseño de interfaz de usuario (Personal Médico Actividades).....	94
Figura 51. Diseño de interfaz de usuario (Personal Médico Crear Actividades).....	95
Figura 52. Diseño de interfaz de usuario (Personal Médico Visualizar Actividades).....	95
Figura 53. Diseño de interfaz de usuario (Personal Médico Documentos).....	96
Figura 54. Arquitectura del sistema.....	97

Erick Fernando García Guerrero, Hector Steveen Ordoñez Chamba. *DESARROLLO DE UN SISTEMA WEB DE BIENESTAR UNIVERSITARIO PARA LA GESTIÓN DOCUMENTAL DEL ÁREA DE MEDICINA EN UNA INSTITUCIÓN DE EDUCACIÓN SUPERIOR.* Carrera Ingeniería en Software. Universidad Iberoamericana del Ecuador. Quito Ecuador. 2025. (138) pp.

RESUMEN

Este trabajo presenta el diseño y desarrollo de un sistema web para la gestión documental en el área de medicina de la Dirección de Bienestar Universitario en la Universidad Iberoamericana del Ecuador. El objetivo es optimizar la eficiencia operativa, reducir el uso de recursos humanos y mejorar la precisión y accesibilidad de la información.

Se empleó una metodología de investigación cuantitativa con un diseño no experimental y transversal. La población y muestra incluyeron al personal encargado del área de bienestar universitario, y los datos fueron recolectados mediante una encuesta.

El sistema se construyó utilizando tecnologías avanzadas como React y AWS, aplicando un enfoque de desarrollo en cascada. Incluye funcionalidades clave como la autenticación de usuarios, la creación de actividades y la generación de reportes relacionados con las actividades del proyecto médico, así como reportes por períodos y años de todas las actividades completadas. También ofrece un seguimiento completo de los estados de las actividades y la gestión de los documentos correspondientes. Se realizaron pruebas con el personal de Bienestar Universitario del área de medicina para validar su funcionamiento y asegurar que cumpla con los requisitos y necesidades del proyecto.

Los resultados mostraron una mejora significativa en la organización de las actividades y los documentos, lo que facilita una gestión más eficiente, reduce los errores humanos y optimiza los tiempos de recuperación de documentos. En conclusión, el sistema, validado por los usuarios finales, demuestra tener un impacto positivo en los procesos administrativos y se presenta como un modelo replicable en otras instituciones.

Palabras Clave: gestión documental, sistema web, tecnología, actividades, accesibilidad

ABSTRACT

This paper presents the design and development of a web-based document management system for the medical area of the University Welfare Directorate at the Universidad Iberoamericana del Ecuador. The objective is to optimize operational efficiency, reduce the use of human resources, and improve the accuracy and accessibility of information.

A quantitative research methodology was employed with a non-experimental and cross-sectional design. The population and sample consisted of the staff responsible for the university welfare area, and the data was collected through a survey.

The system was built using advanced technologies such as React and AWS, applying a waterfall development approach. It includes key functionalities such as user authentication, activity creation, and report generation related to medical project activities, as well as reports by periods and years of all completed activities. It also offers comprehensive tracking of the activity statuses and document management. Tests were conducted with the Welfare University staff in the medical area to validate its functionality and ensure it meets the project's requirements and needs.

The results showed a significant improvement in the organization of activities and documents, which facilitates more efficient management, reduces human errors, and optimizes document retrieval times. In conclusion, the system, validated by end users, demonstrates a positive impact on administrative processes and is presented as a replicable model for other institutions.

Keywords: document management, web system, technology, activities, accessibility

INTRODUCCIÓN

En la actualidad, la tecnología desempeña un papel fundamental en la optimización de los procesos administrativos dentro de las instituciones de educación superior. La digitalización y automatización de la gestión documental permiten mejorar la eficiencia operativa y reducir los errores derivados de los procesos manuales. En este contexto, una institución de educación superior enfrenta el reto de modernizar la gestión documental en el área de medicina de la Dirección de Bienestar Universitario, donde el volumen de documentos y la dependencia de procedimientos tradicionales afectan la accesibilidad, la organización y la precisión de la información. Para abordar esta problemática, se propone el desarrollo de un sistema web de bienestar universitario que facilite la gestión documental en esta área.

El presente estudio tiene como objetivo el desarrollo de un sistema web de bienestar universitario que permita la gestión eficiente de los documentos en el área de medicina. Para ello, se identificarán los requerimientos funcionales y no funcionales del sistema mediante encuestas al personal que integra el área de bienestar universitario, se seleccionará la metodología más adecuada para su implementación y se desarrollará un sistema que cumpla con los estándares de eficiencia operativa, reducción de recursos humanos y mejora en la accesibilidad a la información.

Desde el punto de vista metodológico, la investigación adoptará un enfoque cuantitativo, permitiendo medir y analizar los procesos de gestión documental a través de datos objetivos. La justificación de este estudio radica en la necesidad de optimizar la administración de documentos en el área de medicina, mejorando la eficiencia de los procesos y permitiendo que el personal universitario enfoque sus esfuerzos en actividades de mayor impacto.

El documento se estructura en varios capítulos que abordan de manera integral el desarrollo del sistema propuesto. En el Capítulo I, se presenta el planteamiento del problema, así como los objetivos y la justificación de la investigación. El Capítulo II expone el marco teórico, fundamentando la problemática con estudios previos y teorías relacionadas con la gestión documental y los sistemas web. En el Capítulo III, se describe la metodología empleada para el diseño y desarrollo del sistema. Posteriormente, en el Capítulo IV, se presentan y analizan los resultados obtenidos. Finalmente, el Capítulo V ofrece las conclusiones y recomendaciones derivadas del estudio.

CAPÍTULO I

EL PROBLEMA

En este capítulo se presenta el planteamiento del problema relacionado con el desarrollo de un sistema web de bienestar universitario para la gestión documental del área de medicina en una institución de educación superior. A lo largo del capítulo, se detallan los objetivos generales y específicos de la investigación, proporcionando una justificación clara y precisa del estudio. Asimismo, se delinea el alcance del proyecto, permitiendo una comprensión integral de su relevancia y los beneficios esperados. Esta sección establece la base necesaria para abordar la problemática y las soluciones propuestas.

Planteamiento del problema

Una institución de educación superior ha identificado la necesidad de modernizar sus procesos administrativos para mejorar la gestión de la información y la eficiencia operativa. Actualmente, la Dirección de Bienestar Universitario enfrenta desafíos importantes en la gestión documental del área de medicina, especialmente con las actividades y la carga de documentos asociados a cada una. Debido a esto, la institución está considerando implementar un sistema web que optimice estos procesos, mejorando la accesibilidad y precisión de la información.

A nivel mundial, los sistemas web han evolucionado y se han convertido en herramientas clave para gestionar de manera eficiente los procesos administrativos y educativos. Una de sus principales ventajas es la capacidad de sistematizar tareas repetitivas, lo que permite automatizar y actualizar los procesos administrativos de forma más efectiva.

En Ecuador, y especialmente en el ámbito educativo, la necesidad de adoptar estas tecnologías es cada vez más evidente. En este contexto, la institución de educación superior está avanzando hacia la sistematización de sus procesos administrativos con el objetivo de mejorar la gestión interna y la experiencia de los usuarios.

Como menciona Cedeno, Catuto y Rodas-Silva (2021):

Las aplicaciones web han recorrido un largo camino en su evolución, al punto de que hoy en día, se han convertido en una parte fundamental de las empresas debido a los beneficios que las TICs brindan a los usuarios, mejorando radicalmente la calidad, la rentabilidad y el tiempo de los procesos que se realizan dentro de las diferentes entidades a nivel mundial. (pág. 2)

Lo señalado por Cedeno, Catuto y Rodas-Silva resalta cómo la evolución de las aplicaciones o sistemas web, impulsada por las TICs, ha permitido a las empresas optimizar sus procesos, mejorar la rentabilidad y elevar la calidad de sus servicios. Todos estos avances en la tecnología han contribuido a mejorar los métodos de estudio y de organización de la información, lo que ha convertido a los sistemas web, en una de las herramientas más importantes para los negocios y la administración modernas.

Actualmente La Ley Orgánica de Educación Superior ha hecho recientemente cambios en esta entidad y ha aumentado sus tareas, por lo que la Institución de Educación Superior enfrenta diferentes problemas que afectan su operación y por ende requieren de más recursos humanos, derivados principalmente de la administración intensiva de documentos generados por las actividades que se realizan en esta área. Estos documentos se manejan manualmente y requieren de mucho esfuerzo, por esta razón surge la necesidad de optimizar y automatizar los procesos.

Lo estipulado en la LOE (2018) se asume de forma genérica en el Reglamento de Bienestar Universitario UNIB.E (2012 donde se expresa en su artículo 4 que:

La Unidad de Bienestar Universitario, se apunta con la finalidad de fortalecer un conjunto de programas y actividades destinados a todos los integrantes de la comunidad universitaria, con el objetivo de crear las condiciones favorables para el cumplimiento de los objetivos de la Educación Superior y la promoción de la Comunidad. (pág. 5)

Esta cita resalta la importancia de gestionar de manera eficiente las iniciativas y actividades establecidas en el reglamento. Sin embargo, uno de los principales desafíos es el gran volumen de documentos generados en estos procesos. Su manejo requiere un seguimiento minucioso, lo que consume tiempo y recursos, dificultando la organización y el acceso oportuno a la información.

Asimismo, la utilización de métodos manuales afecta negativamente la eficacia del funcionamiento. Estas acciones son susceptibles a equivocaciones humanas, lo que podría dar lugar a demoras y problemas. Con el incremento de la carga laboral, la posibilidad de cometer errores se eleva, extendiendo el tiempo requerido para finalizar las tareas y repercutiendo en la eficiencia de la organización.

Para abordar esta problemática, se emplea un enfoque cuantitativo que permitirá medir, analizar y optimizar las actividades en la gestión documental en base a datos concretos. La investigación cuantitativa se centrará en recopilar información para desarrollar el sistema.

Derivada de la problemática antes descrita se plantea la siguiente interrogante:

¿Cómo desarrollar un sistema web que optimice la gestión de actividades y documentos en el área de medicina?

Objetivos de la investigación

Objetivo general

Desarrollar un sistema web para la gestión documental del área de medicina en una institución de educación superior, con el objetivo de mejorar la accesibilidad a la información y su eficiencia en la ejecución de actividades.

Objetivos específicos

- Identificar los requerimientos funcionales y no funcionales del sistema web de bienestar universitario para la gestión documental del área de medicina en la institución de educación superior mediante encuestas al personal de bienestar universitario.
- Elegir la metodología más apropiada para el desarrollo del proyecto, teniendo en cuenta sus características y las necesidades del usuario.
- Diseñar el sistema web de gestión documental que cumpla con los requerimientos identificados.
- Desarrollar el sistema web de gestión documental empleando herramientas tecnológicas idóneas que garanticen una operación efectiva y la disminución de esfuerzos en los procesos realizados por los usuarios.
- Validar el sistema con el personal de Bienestar Universitario en cuanto a su funcionalidad y eficiencia.

Justificación e impacto de la investigación

La creación del trabajo de titulación se fundamenta en la urgencia de perfeccionar las labores ligadas a la administración de documentos en el sector de medicina de la Dirección de Bienestar Universitario de la institución de educación superior. En la actualidad, gestionar grandes cantidades de documentos implica un esfuerzo significativo en cuanto a tiempo y personal. La adopción del sistema sugerido facilitará la mejora de estos procedimientos, aumentando así la eficiencia y la disponibilidad de la información.

Para Chávarry y Sandoval (2023):

Esta investigación resalta el potencial transformador de las soluciones digitales para optimizar los procesos de gestión documental en instituciones educativas. La exitosa aplicación del sistema web, desarrollado con la metodología XP, condujo a mejoras considerables en la eficiencia del registro, localización y el proceso de atención, ahorrando tiempo, reduciendo la carga laboral y optimizando el uso de recursos. (pág. 1)

El desarrollo de un sistema de gestión documental específico para el área de medicina se presenta como una solución viable y eficiente para la institución. Este sistema no solo permitirá una gestión más adecuada de los documentos, sino que también proporcionará una interfaz amigable que facilitará a los usuarios la realización de tareas repetitivas y monótonas. Con el desarrollo de este sistema, se espera que las actividades cotidianas relacionadas con la documentación en el área de medicina de la Dirección de Bienestar Universitario se agilicen significativamente, reduciendo errores humanos y mejorando la precisión en el registro y manejo de información.

Asimismo, esto ayudará a localizar y acceder a la información de manera más sencilla, sino que también aumentará la disponibilidad de los documentos y el seguimiento de los procedimientos administrativos.

La implementación de un sistema de gestión documental mejorará significativamente la administración de documentos en el área de medicina. El personal de la Dirección de Bienestar Universitario podrá registrar y gestionar sus actividades de manera más eficiente, asegurando un control preciso de la documentación. Además, la automatización del proceso permitirá responder más rápido y con mayor precisión a las necesidades de la comunidad universitaria, optimizando el acceso a la información y mejorando la calidad del servicio.

El desarrollo de un sistema web para la organización de documentos en el área de medicina de la Dirección de Bienestar Universitario de la Institución de Educación Superior influirá favorablemente en la eficacia administrativa de la entidad. Esta iniciativa no sólo optimizará el manejo de datos dentro de la universidad, sino que también puede actuar como un ejemplo para otras instituciones y organizaciones en Ecuador. Al promover el uso de tecnologías modernas para la gestión de información, facilitará la modernización de los procesos administrativos en el sector educativo. Un mejor aprovechamiento de los recursos y la reducción de errores, permitirá ofrecer servicios de calidad, lo que llevará a una mejora en la eficiencia y efectividad de sus funciones.

La creación de este sistema permitirá mejorar las actividades diarias en el ámbito de la salud dentro de la Dirección de Bienestar Universitario. Estas actividades se realizan de manera más rápida y precisa, reduciendo la carga de trabajo y minimizando fallos. Esto permitirá a los empleados concentrarse en tareas de mayor relevancia, aumentando su efectividad. Además, el acceso y la recuperación de información se volverán más rápidos, creando un entorno de trabajo ordenado que beneficiará la toma de decisiones y la planificación estratégica. Finalmente, esto permitirá mejorar el seguimiento de las actividades y los documentos, garantizando una administración de información más eficiente y confiable.

Este sistema mejorará la eficiencia de la logística del área médica en la Dirección de Bienestar Universitario. Esto hará que los procesos sean más rápidos y precisos, reduciendo la carga de trabajo y los errores del personal. Así, los empleados podrán concentrarse en generar más valor. Además, la recuperación y el acceso a la información también serán más ágiles, asegurando un lugar de trabajo bien organizado que apoye la planificación estratégica y la toma de decisiones. A través de la sistematización, la gestión documental mejorará la transparencia y trazabilidad de los procesos, permitiendo una administración de la información mejor organizada y confiable.

La creación de un sistema de gestión documental proporcionará una estructura clara y eficiente para el manejo de documentos del área de medicina. Esto permitirá mantener un registro detallado y organizado de toda la información, facilitando su búsqueda y recuperación. Además, un sistema bien diseñado contribuirá a la transparencia y trazabilidad de los procesos administrativos, asegurando el cumplimiento de las normativas y políticas institucionales y externas. En resumen, el desarrollo de este sistema permitirá

mejorar los procesos realizados además de garantizar una gestión documental más eficiente y fiable.

Alcance de la investigación

Esta investigación tiene como propósito el diseño y desarrollo de un sistema web para la gestión documental en el área de medicina dentro de la Dirección de Bienestar Universitario de una Institución de Educación Superior. Para ello, se adoptará un enfoque de campo, permitiendo un análisis detallado de los procesos administrativos y operativos vigentes. El estudio abarca el periodo comprendido entre septiembre de 2024 y el 15 de febrero de 2025, durante el cual se llevará a cabo un diagnóstico de las necesidades del personal de bienestar universitario posteriormente a esto se diseñará y desarrollará el sistema en base a los requerimientos funcionales y no funcionales del área de medicina. Además, se procederá a la validación del sistema mediante la funcionalidad y disponibilidad del sistema permitiendo asegurar que el sistema cumple con los requerimientos solicitados. Con esto se espera contribuir a la modernización y eficiencia de la gestión documental en el área de medicina, favoreciendo la toma de decisiones basada en información organizada y accesible.

CAPÍTULO II

MARCO TEÓRICO

Este capítulo aborda antecedentes de estudios y proyectos realizados en los últimos 5 años sobre la gestión de documentación de medicina en instituciones educativas o contextos similares, así como las bases teóricas que sustentan el desarrollo de los objetivos planteados. En el contexto de las instituciones de educación superior, la gestión eficiente de la documentación de medicina es fundamental para garantizar el bienestar universitario. A continuación, se detallan los aspectos históricos y teóricos que fundamentan el desarrollo del sistema web propuesto.

Antecedentes de la Investigación

En primer lugar, se tiene la investigación realizada por Mendoza (2023), titulada "Desarrollo del sistema web para la automatización del registro de ficha socioeconómica en la unidad de bienestar universitario en la Universidad Nacional de Barranca 2020" con el objetivo de desarrollo de un sistema web de bienestar universitario para la gestión documental del área de medicina en una institución de educación superior, que optimice la eficiencia operativa, reduzca el uso de recursos humanos, y mejore la precisión y accesibilidad de la información. La investigación correspondió al tipo básica, con un nivel descriptivo y un diseño no experimental de enfoque cuantitativo. La muestra fue de tipo censal, siendo similar a la población, compuesta por 10 colaboradores de la unidad de bienestar universitario en la Universidad Nacional de Barranca. Asimismo, en relación a los objetivos específicos, las dimensiones basadas en el coeficiente de Spearman fluctúan entre magnitudes positivas altas: dimensión operación (0.716), dimensión plataforma (0.710) y dimensión gestión (0.772). La hipótesis planteada sugiere que una mayor aplicación en el desarrollo del sistema web conducirá a una mayor automatización en el proceso de registros y entrega de la ficha socioeconómica, lo cual está respaldado por los valores estadísticos obtenidos. Los valores de correlación de Spearman para las áreas de operaciones (0.716), plataforma (0.710) y gestión (0.772) muestran una conexión sólida y favorable entre la adopción del sistema en línea y el avance en la automatización de estas tareas.

En segundo lugar, la necesidad de contar con sistemas de información que faciliten la administración de procesos productivos y el análisis de la información para la toma de decisiones, tal como se evidencia en el área de Bienestar Universitario de la Universidad

del Norte (Acosta et al., 2024), guarda una estrecha relación con el presente estudio. Ambos comparten el objetivo de optimizar el tiempo en la realización de los procesos de la Dirección de Bienestar Universitario. El presente proyecto tiene como propósito principal implementar un sistema de gestión para agilizar los servicios: social, psicopedagógico, médico, odontológico, el proceso de inscripción al comedor universitario y la bolsa de trabajo en las unidades de Servicio Social, Unidad de Salud y Unidad de Servicios Alimentarios de la Dirección de Bienestar Universitario (DBU) de la UNASAM, que permita optimizar sus procesos de gestión y dar conformidad de satisfacción a todos los requerimientos de la DBU. Con el desarrollo de este proyecto se busca optimizar los procesos y solucionar las problemáticas en las diferentes unidades, las cuales están encargadas, a través de sus directivos, de realizar las actividades de cada proceso y la toma de decisiones en los distintos servicios que otorga la DBU.

En tercer lugar, se considera la investigación de Luna Alvarado (2024) de la Universidad Técnica de Babahoyo, titulada "Automatización de la Administración de Procesos y su Impacto en la Plataforma Tecnológica para la Gestión de Comunidades Académicas y Recreativas de la Universidad Técnica de Babahoyo". Este trabajo analiza la automatización, sus tecnologías y ventajas, centrándose en la gestión de procesos de negocios (BPM), la automatización robótica de procesos (RPA) y la automatización de procesos inteligentes (IPA). El objetivo de este proyecto es diseñar la automatización de los procesos administrativos en la plataforma tecnológica, con la finalidad de mejorar la eficiencia operativa, la productividad y la tolerancia a fallos humanos, además de ser escalable a diferentes niveles de automatización según los requerimientos y tecnologías a utilizar.

En cuarto lugar, la editorial académica liderada por Isabel Zambrano, Eneida Quindemil y Felipe León (2021), titulada "*Gestión documental en Universidades: Una mirada desde Latinoamérica*", analiza la implementación de un sistema de gestión documental (SGD) con un enfoque integral. Este es un modelo que propone un desarrollo basado en el cumplimiento de las leyes y regulaciones actuales, adopta un enfoque orientado a procesos y un papel integrado de las tecnologías de la información y la comunicación (TIC). Es probable que este sistema afecte las políticas de calidad y fomente la mejora continua de las instituciones de educación superior.

En lo que respecta al bienestar universitario, hay una grave escasez de sistemas de información o herramientas que puedan gestionar datos y procesos de manera dinámica. En la actualidad, este departamento opera almacenando su información de manera física, lo que conlleva altos gastos en papelería e impresión. Los servicios médicos también son limitados:

- Las actividades y documentos se registran de forma manual.
- No se registran los estados en que se encuentran las actividades y los documentos que pertenecen a las mismas.

Debido a esta limitación, no se puede hacer un seguimiento adecuado de los usuarios durante el proceso, lo que compromete el servicio. Las universidades carecen de un entorno digital que les permita tener una visión general de:

- Los tipos de actividades de bienestar.
- Los servicios universitarios.

Además, todas estas actividades son menos efectivas desde el punto de vista administrativo.

Este estudio destaca la importancia de desarrollar un sistema web que permita soluciones tecnológicas que mejoren la administración en estos procesos cruciales, eleven la calidad de los servicios y apoyen el fortalecimiento de las instituciones.

Bases teóricas

Sistemas de gestión documental

Un sistema de gestión de documentos es un conjunto de reglas, procedimientos y herramientas para el procesamiento eficiente de información en una organización. Su función principal es categorizar, clasificar, almacenar y recuperar los documentos para un acceso seguro y estructurado a la información. Manuales y digitales, estos sistemas agilizan su implementación para mejorar la eficiencia operativa, mantener el cumplimiento normativo y optimizar la gestión de documentos a lo largo de su ciclo de vida.

Las instituciones y organizaciones académicas manejan un alto volumen de documentación, por lo que encontrar formas eficientes de gestionarla es vital. Un Sistema de Gestión de Documentos (DMS) permite digitalizar la información en su organización y organizar, buscar y recuperar de manera digital, lo que garantiza un menor uso de papel y menos tiempo buscando información. Esto proporciona una gestión más ágil y eficiente de los recursos documentales.

Como menciona Suárez, Junco, Ruiz (2021):

Las universidades son organizaciones que generan un gran volumen de información, sobre todo con el vertiginoso desarrollo de las tecnologías de la información donde la producción documental va más allá de un simple documento en papel a convertirse en "toneladas" de información digital. Con ello surgen los Sistemas de Gestión Documental (SGD) que permiten, desde el punto de vista económico, una importante reducción de costes en recursos humanos y materiales. (pág. 01)

Además, un sistema de información de gestión mejora la transparencia y trazabilidad de los procesos administrativos y asegura el cumplimiento de las regulaciones y estándares institucionales. La toma de decisiones precisa y oportuna se hace posible gracias a la automatización de tareas y la reducción de errores humanos rutinarios, lo que ayuda a modernizar los procesos administrativos y proporciona una mayor eficiencia en la gestión de información orientada al desarrollo académico y la prerrogativa del desarrollo de procesos organizacionales.

Al llevar a cabo un análisis del "Sistema Web para la Gestión Documental en el Ministerio de Salud" desarrollado por Egusquiza, I. (2018), que utilizó la metodología SCRUM y herramientas como Python, Django y PostgreSQL, en comparación con la opción propuesta que usa AWS Amplify, se pueden notar diferencias notables en cuanto a su diseño y capacidad de expansión. Por un lado, el sistema del Ministerio de Salud se fundamenta en una infraestructura tradicional que incluye servidores dedicados y bases de datos gestionadas de manera independiente, mientras que la propuesta basada en AWS Amplify ofrece una solución en la nube que facilita la implementación de servicios como bases de datos escalables, autenticación de usuarios y funciones sin servidor en la nube.

Este enfoque moderno permite una gestión más eficiente y flexible de los recursos, ya que se adapta de forma dinámica a las necesidades del proyecto, brindando una base robusta y ampliable. De igual manera, la integración con otros servicios de AWS no solo mejora la eficacia de la aplicación, sino que también refuerza su capacidad de ajustarse a nuevas

demandas, algo esencial para el desarrollo y la ejecución de aplicaciones web en entornos cambiantes y de alta carga.

Sistema web

Los sistemas web pueden ser definidos como conjuntos de componentes y funcionalidades conectadas que funcionan en la red con el objetivo de brindar servicios a los usuarios. Estas estructuras han sido desarrolladas para ser accesibles a través de navegadores y pueden incluir un amplio rango de aplicaciones, desde sitios web simples hasta avanzados servicios en línea.

Como lo menciona Codina, Lluís, Marcos, Mari Carmen y Pedraza, Rafael (2023):

Los sistemas web pueden definirse como conjuntos de componentes y servicios interconectados que se ejecutan en internet para proporcionar funcionalidades específicas a los usuarios. Estos sistemas están diseñados para ser accesibles a través de navegadores web y pueden incluir una amplia gama de aplicaciones, desde sitios web simples hasta complejos servicios en línea. (pág. 10)

Como mencionan los autores de la cita los sistemas web se definen como un conjunto de componentes y servicios que permiten ser accesibles a través de navegadores y otros dispositivos lo que hoy en día es valorado ya que nos encontramos en mundo globalizado donde todos hacemos uso de sistemas web o plataformas que nos ofrecen servicios para satisfacer nuestras necesidades.

Para profundizar en el tema, es esencial resaltar el significado de la Web 2. 0, que denota la segunda fase de herramientas y servicios en internet centrados en la colaboración, el intercambio de datos y la generación de contenido por los usuarios. Componentes clave de la Web 2. 0 incluyen redes sociales, wikis, blogs y plataformas para el intercambio de contenido multimedia. La facilidad de uso y la interacción son aspectos cruciales, promoviendo una experiencia de usuario más participativa y cooperativa.

Metodologías de desarrollo de software

Los proyectos de software conllevan utilizar metodologías ágiles por lo cual vamos a analizar las tres siguientes y en base a esta investigación podremos seleccionar la adecuada a nuestro proyecto.

Scrum es una metodología ágil que divide el trabajo en pequeños incrementos de una a cuatro semanas llamados sprints. Actividades como la planificación, el desarrollo, la revisión y la entrega de características específicas se llevan a cabo siempre durante cada

sprint; para eso, es necesario conocer cuáles son los elementos clave que realmente incluyen: roles, actividades y el artefacto.

Haciendo énfasis en lo que menciona Medina Salgado (2024):

La revisión de los estudios analizados revela que la implementación de Scrum tiene un impacto positivo en la cohesión de los equipos de trabajo. Uno de los elementos clave que facilita este proceso es la estructura organizativa de Scrum, la cual fomenta una comunicación abierta y constante a través de reuniones periódicas, como las Daily Stand-ups y los Sprints de retroalimentación. Estas reuniones permiten que los miembros del equipo mantengan una comunicación fluida, lo que ayuda a identificar obstáculos y a tomar decisiones de manera conjunta, promoviendo un mayor nivel de colaboración y entendimiento mutuo. (pág. 07)

Como menciona Medina Salgado la comunicación dentro del marco de trabajo Scrum es fundamental. Al establecer roles específicos, artefactos y actividades, Scrum asegura que todos los miembros del equipo estén alineados y puedan colaborar de manera efectiva. Esto no solo facilita la coordinación de tareas, sino que también fortalece el compromiso y la responsabilidad individual y colectiva, elementos cruciales para el éxito de cualquier proyecto.

Por otra parte, la metodología Waterfall o Cascada es una metodología tradicional que sigue un enfoque secuencial donde cada fase del desarrollo debe completarse antes de pasar a la siguiente. Es ideal para proyectos con requisitos bien definidos desde el inicio, alguna de sus partes importantes Requerimientos, Diseño, Implementación, Verificación, Mantenimiento

Patrones de diseño

Para el enfoque de la investigación, se mencionan los siguientes patrones de diseño como fundamentales en el desarrollo de aplicaciones web.

Modelo Vista Controlador (MVC)

El patrón Modelo Vista Controlador es frecuentemente utilizado en el desarrollo de aplicaciones web debido a su capacidad para organizar y segmentar las funcionalidades dentro de una aplicación.

- **Modelo:** Representa la estructura de los datos y la razón detrás del software. Su propósito es administrar y trabajar con la información.
- **Vista:** Es la interfaz que muestra la información al usuario y capta las interacciones que éste realiza.

- **Controlador:** Actúa como un intermediario entre el Modelo y la Vista, gestionando las solicitudes del usuario y realizando los ajustes requeridos en el Modelo y la Vista cuando sea preciso.

Este método establece una clara separación entre las diferentes preocupaciones, lo que facilita el cuidado del código, la reutilización de componentes y la colaboración entre desarrolladores que participan en varias partes del sistema.

Monolítico

Este método ofrece una clara separación entre las diversas preocupaciones, lo que mejora la mantenibilidad del código, permite la reutilización de componentes y facilita la colaboración entre desarrolladores que trabajan en diferentes partes del sistema.

- **Componentes Integrados:** Todos los elementos de la aplicación, como el Modelo, la Vista y el Controlador, se llevan a cabo y se implementan conjuntamente como una unidad única.
- **Ventajas:** Facilita el proceso inicial al eliminar la necesidad de ajustar múltiples sistemas o servicios. Se integra con mayor comodidad en entornos de servidores estándar.
- **Desafíos:** Puede volverse complejo y difícil de expandir a medida que la aplicación se amplía en tamaño y funcionalidad. Los cambios en una parte del sistema pueden afectar a otras partes debido a su intensa interconexión.

Framework de desarrollo web

En el ámbito del desarrollo de aplicaciones web, los marcos de trabajo son recursos clave que ofrecen un esquema y características establecidas para simplificar la construcción y el cuidado de las aplicaciones. Este estudio se enfocará en examinar diferentes marcos significativos, tanto del lado del usuario como del lado del sistema.

Frameworks del Lado del Cliente

React

React es una herramienta de JavaScript creada por Facebook que se emplea para desarrollar interfaces de usuario dinámicas, fundamentada en la idea de componentes que se pueden usar varias veces. Sus rasgos principales comprenden un método declarativo para establecer interfaces y un manejo eficaz del estado a través de la estructura Flux o Redux.

Angular

Angular es un marco de trabajo creado por Google que emplea TypeScript y se adhiere al modelo MVVM (Modelo-Vista-Vista Modelo). Ofrece una base sólida para desarrollar aplicaciones de una sola página (SPA), incluyendo características como la vinculación de datos en ambas direcciones y la inyección de dependencias.

Tabla 1. Descripción de las características de los frameworks del lado del cliente.

Característica	React	Angular
Desarrollador	Facebook	Google
Lenguaje base	JavaScript (soporte para TypeScript)	TypeScript
Arquitectura	Basada en componentes reutilizables	MVVM (Modelo-Vista-Vista Modelo)
Curva de aprendizaje	Moderada	Pronunciada
Rendimiento	Eficiente (Virtual DOM)	Muy bueno (Change Detection)

Tras evaluar las características de React y Angular, hemos decidido utilizar React para el desarrollo de nuestro sistema. Aunque Angular ofrece una estructura más robusta con su patrón MVVM y funcionalidades como la inyección de dependencias, la curva de aprendizaje de React es más moderada, lo que facilita una adopción más rápida en el equipo. Además, el enfoque basado en componentes reutilizables y la eficiencia en el rendimiento gracias al Virtual DOM hacen que React sea ideal para construir interfaces de usuario interactivas y escalables, lo que se adapta mejor a los requerimientos de nuestro proyecto.

Frameworks del Lado del Servidor

Laravel

Laravel es un framework PHP que sigue el patrón MVC (Modelo-Vista-Controlador). Valorado por su simplicidad en su uso y variedad de funciones incorporadas, tales como enrutamiento RESTful, manejo de sesiones y verificación de identidad.

NestJS

NestJS representa un marco de trabajo para Node.js que utiliza TypeScript y sigue el patrón MVC. Fue diseñado para construir aplicaciones que sean tanto efectivas como escalables en Node.js, aprovechando las características de TypeScript, como la tipificación estática y la programación orientada a objetos.

Tabla 2. Descripción de las características de los frameworks del lado del servidor.

Característica	Laravel	NestJS
Lenguaje base	PHP	TypeScript/JavaScript
Patrón arquitectónico	MVC (Modelo-Vista-Controlador)	MVC (Modelo-Vista-Controlador)
Facilidad de uso	Alta (elegante y sintaxis sencilla)	Moderada (requiere conocimiento de TypeScript)
Casos de uso	Aplicaciones web, APIs, y sistemas completos	APIs escalables y aplicaciones backend.
Características clave	Enrutamiento, autenticación, ORM (Eloquent)	Tipificación estática, modularidad, inyección de dependencias.

Cada una de estas herramientas presenta beneficios y características distintas que pueden afectar la elección de la plataforma más adecuada para un proyecto determinado.

Laravel y NestJS son marcos de trabajo del lado del servidor que son eficientes y ofrecen una base firme, facilitando la creación de APIs y aplicaciones web que pueden escalar fácilmente. No obstante, si lo que buscas es una alternativa que sea rápida, escalable y completamente integrada con la nube, AWS Amplify se destaca como la mejor elección.

La metodología sin servidor, combinada con su simplicidad y su conexión con los servicios de AWS, facilita a los programadores el desarrollo y despliegue de aplicaciones que funcionan en plataformas web y móviles de manera ágil, al mismo tiempo que reduce la carga asociada con la gestión de la infraestructura y agiliza el proceso de creación.

Lenguajes de programación

En el desarrollo de sistemas web, la elección del lenguaje de programación es crucial, ya que determina la eficiencia, la mantenibilidad y la escalabilidad de la aplicación. A continuación, se exploran los lenguajes asociados a algunos de los principales frameworks de desarrollo web.

JavaScript y TypeScript:

Frameworks como React, Angular y Vue.js utilizan estos lenguajes de programación que permiten escribir componentes de interfaz de usuario utilizando JavaScript, un lenguaje ampliamente utilizado y compatible con todos los navegadores web modernos. También es posible integrar TypeScript en proyectos React para agregar tipado estático opcional y otras características avanzadas que mejoran la robustez del código.

PHP:

Laravel es un framework de PHP que sigue el patrón de arquitectura MVC. PHP es conocido por ser un lenguaje simple y poderoso para el desarrollo web, proporcionando una sintaxis clara y directa que facilita la creación de aplicaciones robustas y escalables.

Java:

Spring Boot es un framework de Java diseñado para desarrollar aplicaciones Java empresariales de forma rápida y eficiente. Java es un lenguaje conocido por su portabilidad, seguridad y rendimiento, siendo ideal para aplicaciones que requieren alta escalabilidad y procesamiento intensivo.

Tabla 3. Descripción de las características de lenguajes de programación.

Característica	JavaScript y TypeScript	PHP	Java
Frameworks asociados	React, Angular, Vue.js	Laravel	Spring Boot
Propósito principal	Interfaces de usuario y aplicaciones web interactivas	Desarrollo web completo (backend y frontend)	Aplicaciones empresariales y backend
Curva de aprendizaje	Suave (JavaScript), Moderada (TypeScript)	Suave	Moderada a pronunciada
Característica clave	Tipado dinámico (JavaScript), tipado estático opcional (TypeScript)	Sintaxis clara, orientación al desarrollo web	Portabilidad, rendimiento, y escalabilidad
Popularidad	Altamente popular (JavaScript es el estándar web)	Popular en desarrollo web backend	Popular en aplicaciones empresariales

Hemos elegido JavaScript junto con TypeScript como lenguaje de programación principal debido a su versatilidad y amplia adopción en el desarrollo web. Su compatibilidad con todos los navegadores modernos y su capacidad para crear interfaces interactivas lo convierten en la mejor opción para nuestro proyecto. Además, su integración con frameworks como React nos permite desarrollar una aplicación escalable y eficiente, optimizando tanto la experiencia del usuario como el rendimiento general del sistema.

Sistema de gestión de base de datos

En la creación de aplicaciones web, seleccionar el sistema de administración de bases de datos es fundamental, ya que influye directamente en la eficacia, capacidad de crecimiento y protección de la aplicación. A continuación, se describen tres de los sistemas de administración de bases de datos más comunes en el sector y sus particularidades: PostgreSQL, MySQL y MongoDB.

PostgreSQL

PostgreSQL es un sistema de administración de bases de datos relacional de código libre, famoso por su solidez, habilidad para gestionar grandes cantidades de información y su compatibilidad con transacciones ACID (Atomicidad, Consistencia, Aislamiento y Durabilidad). Entre sus características fundamentales se encuentran un modelo relacional clásico, gran escalabilidad y soporte para extensiones.

- **Modelo Relacional:** PostgreSQL utiliza un método relacional tradicional que brinda plena compatibilidad con el SQL estándar y funciones avanzadas extras.
- **Capacidad de Escala y Eficiencia:** Tiene una notable habilidad para crecer y es perfecto para aplicaciones que requieren manejar procesos complejos y transacciones.
- **Asistencia para Extensiones:** Permite el desarrollo de funciones y extensiones a medida para cumplir con las necesidades específicas del proyecto.
- **Comunidad Dinámica y Asistencia:** Cuenta con una comunidad activa de desarrolladores y una documentación sólida, lo que hace más fácil su uso y mantenimiento a largo plazo.

MySQL

MySQL es un sistema bien conocido para la administración de bases de datos relacionales que se utiliza mucho, famoso por su rapidez y simplicidad. Sus principales características son un alto rendimiento, adaptabilidad y capacidad de expansión, así como su amplia aceptación y conformidad con las normas SQL.

- **Rapidez y Eficiencia:** MySQL es famoso por su eficiente funcionamiento, sobre todo en contextos que manejan grandes cantidades de información.
- **Adaptabilidad y Capacidad de Ampliación:** Se puede expandir de manera vertical y horizontal, lo que lo convierte en una opción ideal para distintas aplicaciones y tipos de carga de trabajo.
- **Amplia Adopción:** Es utilizado por muchas organizaciones grandes y pequeñas debido a su fiabilidad y bajo costo de implementación.
- **Cumplimiento Normativo:** Se ajusta a las especificaciones SQL y brinda soporte para múltiples sistemas operativos y plataformas.

DynamoDB

Amazon DynamoDB es una base de datos NoSQL que no requiere servidor, famosa por su rapidez, adaptabilidad y capacidad de crecimiento. Entre sus rasgos destacados se encuentran un rendimiento de milisegundos de un solo dígito, un modelo de datos versátil y gran escalabilidad, junto con su diseño sin servidor.

- **Velocidad y Efectividad:** DynamoDB ofrece un rendimiento rápido con tiempos de respuesta en un solo dígito en milisegundos, incluso bajo condiciones de gran volumen.
- **Adaptabilidad y Crecimiento:** Siendo una base de datos NoSQL, DynamoDB proporciona un esquema de datos versátil, lo que facilita el almacenamiento de información estructurada, semiestructurada y no estructurada. Asimismo, se ajusta de manera automática para gestionar elevadas cantidades de datos y flujo de tráfico.
- **Modelo sin servidor:** Como un sistema de base de datos que no requiere servidor, DynamoDB ofrece a los usuarios la ventaja de abonar solo por el uso que tienen, eliminando la obligación de manejar servidores o llevar a cabo actualizaciones.
- **Ejemplos de aplicación:** Resulta perfecto para software que necesita rápida respuesta y gran capacidad de expansión, como sitios web, apps móviles, videojuegos y marketing digital. Industrias como la financiera y la de medios de comunicación emplean DynamoDB por su protección, capacidad de recuperación y eficacia.

Tabla 4. Descripción de las características de los sistemas de gestión de base de datos.

Característica	PostgreSQL	MySQL	DynamoDB
Modelo de Datos	Relacional (SQL)	Relacional (SQL)	NoSQL (Clave-Valor, Documentos)
Escalabilidad	Alta escalabilidad vertical y horizontal	Buena escalabilidad, especialmente en lectura	Escalabilidad automática y sin límites
Rendimiento	Excelente para operaciones complejas y transacciones ACID	Rápido en operaciones de lectura, especialmente en entornos web	Latencias de milisegundos de un solo dígito
Flexibilidad	Estructura de datos rígida, adecuada para datos estructurados	Estructura de datos rígida, adecuada para datos estructurados	Modelo flexible para datos estructurados y semiestructurados
Soporte de Extensiones	Amplio soporte para extensiones y funciones personalizadas	Soporte limitado para extensiones	Sin soporte para extensiones, pero altamente integrable con AWS
Comunidad y Soporte	Comunidad activa y amplia documentación	Comunidad activa y amplia documentación	Comunidad activa, soporte de AWS y documentación extensa
Casos de Uso Comunes	sistemas web, empresarial y gestión de contenido	Aplicaciones web, sistemas de gestión de contenido, comercio electrónico	Aplicaciones web y móviles, juegos, publicidad digital, servicios financieros

Hemos decidido optar por DynamoDB debido a sus características específicas que se ajustan a las necesidades de nuestro proyecto. Su diseño sin servidor, capacidad de escalar automáticamente y excelente rendimiento lo hacen ideal para gestionar nuestros datos de forma eficaz. Asimismo, su formato de almacenamiento optimizado para realizar consultas veloces y su conexión con el entorno de AWS nos facilitan la creación de un sistema web resistente y escalable.

Plataformas en la nube

En el contexto del desarrollo web y de aplicaciones, Firebase y Amazon Web Services (AWS) son dos plataformas ampliamente utilizadas que ofrecen una variedad de servicios para alojar, gestionar y escalar aplicaciones web y móviles. A continuación, se explican las características y beneficios clave de cada una:

Firebase

Firebase es una plataforma desarrollada por Google que ofrece una amplia gama de servicios backend como base de datos en tiempo real, autenticación de usuarios, alojamiento estático y funciones en la nube. Sus principales características incluyen:

- **Base de Datos en Tiempo Real:** Firebase Database es una base de datos NoSQL en tiempo real que facilita la sincronización de datos entre clientes en tiempo real.
- **Verificación de Usuarios:** Proporciona soluciones de autenticación que se pueden combinar fácilmente con plataformas como Facebook, Google y Twitter, entre otras.
- **Alojamiento de Aplicativos:** Permite guardar aplicaciones web estáticas de manera sencilla y escalable según los requerimientos.
- **Servicios en la Nube:** Permite ejecutar código del lado del servidor sin requerir un servidor, respondiendo a eventos generados por usuarios o cambios en la base de datos.
- **Conexión con Google Cloud:** Brinda una conexión directa con diversos servicios de Google Cloud Platform.

AWS (Amazon Web Services)

AWS (Amazon Web Services) es una plataforma de servicios en la nube líder proporcionada por Amazon, que incluye una amplia gama de servicios para computación, almacenamiento, bases de datos, análisis, aprendizaje automático, inteligencia artificial y más. Entre sus principales servicios, se encuentran:

- **AWS Amplify:** Un conjunto de herramientas y servicios diseñado para facilitar el desarrollo de aplicaciones web y móviles escalables, permitiendo una integración sencilla con servicios en la nube como autenticación, almacenamiento, bases de datos y APIs.
- **Nube de Cómputo Elástico (EC2):** Proporciona recursos informáticos escalables en la nube, permitiendo la configuración y supervisión de servidores virtuales.
- **Amazon RDS:** Es un servicio de base de datos relacional que es administrado y que es compatible con diferentes motores de bases de datos como MySQL, PostgreSQL, Oracle y SQL Server.
- **Lambda:** Permite la ejecución de código en el servidor en respuesta a eventos sin necesidad de administrar servidores.
- **S3 (Simple Storage Service):** Proporciona un espacio de almacenamiento flexible para información y material estático.
- **Amplia Gama de Servicios:** AWS ofrece más de 200 servicios complementarios, que abarcan desde herramientas de inteligencia artificial, análisis de datos, medidas de seguridad y mucho más.

Tabla 5. Descripción de las características de plataformas en la nube.

Característica	Firebase	AWS (Amazon Web Services)
Propósito Principal	Plataforma de desarrollo de aplicaciones móviles y web.	Plataforma integral de servicios en la nube para empresas de todos los tamaños.
Base de Datos	Firebase Realtime Database and Firestore: bases de datos NoSQL en tiempo real.	Amazon DynamoDB: base de datos NoSQL escalable y de alto rendimiento.
Autenticación de Usuarios	Servicios de autenticación integrados con proveedores como Google, Facebook y Twitter.	Amazon Cognito: servicio de autenticación y gestión de usuarios.
Alojamiento de Aplicaciones	Alojamiento de aplicaciones web estáticas y dinámicas.	AWS Amplify: Plataforma para crear, hospedar y lanzar aplicaciones móviles y web que son escalables; Amazon S3: servicio de almacenamiento de objetos; Amazon EC2: instancias virtuales para aplicaciones.
Funciones en la Nube	Cloud Functions: ejecución de código backend sin servidor.	AWS Lambda: ejecutar código en reacción a eventos sin requerir la gestión de servidores.
Escalabilidad	Escalabilidad automática según la demanda.	Escalabilidad amplia y adaptable, ideal para aplicaciones de diversos tamaños.

Facilidad de Uso	Interfaz intuitiva y fácil integración con otros servicios de Google.	Amplia gama de servicios con una curva de aprendizaje más pronunciada.
Precios	Modelos de precios más simples y económicos para proyectos pequeños.	Estructura de precios más compleja; puede ser más costoso dependiendo de los servicios utilizados.
Casos de Uso Comunes	Aplicaciones móviles y web de pequeña a mediana escala.	Aplicaciones empresariales, análisis de datos, inteligencia artificial, entre otros.

En general, Firebase y AWS son excelentes plataformas para la construcción de aplicaciones y sitios web, cada una con sus beneficios y casos de uso. La elección adecuada para cada proyecto variará en los requerimientos y habilidades del personal.

Después de considerar todas las opciones, se ha tomado la decisión unánime de construir el sistema en línea en el nivel gratuito de Amazon Web Services (AWS). Flexibilidad y control sobre la infraestructura: AWS ofrece una variedad de servicios que escalan según el tamaño de tu proyecto. Proporciona una red global con múltiples regiones y zonas de disponibilidad, garantizando alta disponibilidad y rendimiento, los cuales son críticos para que los sistemas de negocio sean confiables y escalables.

Este proyecto dependerá en gran medida de servicios esenciales como AWS Amplify, que ayudan en el diseño, alojamiento y lanzamiento de aplicaciones móviles y web escalables. AWS Amplify se integrará automáticamente con otros servicios de AWS, como almacenamiento, bases de datos y sistemas de autenticación, lo que hará que no solo el proceso de desarrollo, sino también el tiempo de entrega sean más eficientes.

Además de esto, Amazon ejecutará EC2 (Elastic Compute Cloud) para crear instancias de servidores virtuales que se pueden escalar según sea necesario y luego RDS (Relational Database Service) para manejar las necesidades de la base de datos (PostgreSQL o MySQL) con alta disponibilidad y copias de seguridad automáticas.

La gestión de usuarios y la autenticación se realizan a través de Amazon Cognito, mientras que AWS Lambda nos permitirá ejecutar nuestro código sin mantener servidores físicos, maximizando el uso de recursos.

La habilidad de AWS para combinar estos servicios de forma escalable y su adaptabilidad, resaltando AWS Amplify como un recurso esencial para facilitar el desarrollo, la convierten en una opción óptima para proyectos en expansión. Aunque el proceso de aprendizaje puede ser un poco más complicado, su fortaleza y capacidad de personalización respaldan su elección, garantizando que la plataforma web será segura, escalable y lista para enfrentar retos tecnológicos venideros.

Modelo de estimación

COCOMO II

COCOMO II, que significa Modelo de Costo Constructivo II, es una versión mejorada del modelo original COCOMO, que fue creado por Barry Boehm en 1995. Este modelo se emplea para calcular el costo y la cantidad de trabajo requerido en los proyectos de software, teniendo en cuenta el tamaño del software y otros aspectos que afectan la productividad.

El propósito de COCOMO II es calcular el trabajo necesario para el desarrollo, así como el costo y el tiempo que llevará un proyecto de software. Obtiene estas estimaciones a partir del tamaño del software junto con otros elementos que impactan el proceso de desarrollo.

Tras realizar un análisis detallado, hemos decidido que utilizar el modelo COCOMO II será adecuado para estimar las necesidades de nuestro proyecto.

Function Point Analysis (FPA)

El Análisis de Puntos de Función (FPA) es una técnica que se usa para calcular el tamaño del software dependiendo de las tareas que el sistema realiza para el usuario. Este enfoque se centra en enumerar las funciones del software, como entradas, salidas, consultas, archivos e interfaces. FPA resulta valioso para evaluar el esfuerzo y los recursos requeridos para construir sistemas, ofreciendo una valoración basada en la funcionalidad esperada del sistema, en lugar de en las líneas de código. Este enfoque hace posible obtener una

estimación más precisa cuando se tienen definiciones claras de los requerimientos funcionales del sistema.

Tabla 6. Descripción de las características de los modelos de estimación

Característica	COCOMO II	Function Point Analysis (FPA)
Enfoque de estimación	Basado en el tamaño del software en líneas de código (KLOC) y factores de coste.	Basado en el análisis de las funciones del sistema.
Tipos de proyectos adecuados	Proyectos de desarrollo de software a gran escala.	Proyectos con requisitos funcionales bien definidos.
Factores considerados	Tamaño del software, experiencia del equipo, complejidad del sistema, entre otros.	Funciones que realiza el sistema para el usuario.
Fases de estimación	Requiere una fase de definición del tamaño y una fase de asignación de factores.	Requiere contar las funciones del sistema.
Facilidad de implementación	Requiere más datos técnicos y detalles del software.	Más simple de aplicar y entender por los usuarios.
Precisión	Mayor precisión en proyectos de gran escala y complejidad.	Puede ser menos preciso si los requisitos funcionales no están bien definidos.

Elegimos utilizar el modelo COCOMO II para estimar el esfuerzo de nuestro proyecto, ya que puede analizar varios factores que impactan el proceso de desarrollo y proporciona una mayor exactitud en las tareas. Si bien el Análisis de Puntos de Función (FPA) es una opción

válida, especialmente cuando los requisitos funcionales son bien definidos, COCOMO II se adapta mejor a nuestras necesidades.

Bases Legales

Sistemas de Gestión documental

La administración de documentos en las instituciones de educación superior ha pasado por un notable desarrollo gracias a los progresos tecnológicos y la mayor necesidad de eficacia en la gestión de datos. Seguidamente, se exponen los antecedentes más importantes que han impactado la idea de crear un sistema en línea para la administración de documentos en el sector de medicina en la universidad.

Evolución de los Sistemas de Gestión Documental

Durante la última década, los métodos para administrar documentos han experimentado transformaciones importantes, impulsadas por el progreso en las tecnologías de la información y la comunicación. Estas soluciones permiten la digitalización, conservación, clasificación y búsqueda efectiva de documentos, reduciendo considerablemente el uso de papel y optimizando el tiempo necesario para encontrar información.

Como señala Ferriol Marchena y otros (2008):

"La implementación de sistemas de gestión documental no solo mejora la eficiencia en la producción y los servicios, sino que también aumenta la calidad de los productos y servicios". (Ferriol Marchena et al., 2008).

Impacto de la Tecnología en la Gestión Educativa

El uso de tecnologías avanzadas en la gestión educativa ha permitido a las instituciones mejorar sus procesos administrativos y educativos. Según Angulo, Mesías y Olmedo (2021):

"La integración de las nuevas tecnologías tiene una gran repercusión en el ámbito educativo, ya que el internet es un recurso útil para buscar información y elaborar tareas académicas en el nivel superior, proporcionando muchas oportunidades en la formación de futuros profesionales. La incorporación de estas tecnologías es fundamental para satisfacer las demandas actuales de los usuarios y mejorar la experiencia educativa". (pág. 9)

Reformas en la Ley de Educación Superior en Ecuador

La modificación de la Ley Orgánica de Educación Superior (LOES) en 2018 otorgó nuevas responsabilidades a las áreas de bienestar universitario, lo que incrementó la documentación necesaria y la exigencia de una gestión de información más efectiva. De acuerdo con lo que establece el Reglamento de Bienestar Universitario de UNIB. E (2012), la Unidad de Bienestar Universitario está encargada de organizar diversos programas y actividades para la comunidad académica, lo que conlleva un manejo considerable de documentos y la necesidad de un sistema que simplifique dichos procedimientos.

CAPÍTULO III

MARCO METODOLÓGICO

Este capítulo detalla el enfoque metodológico utilizado para alcanzar los objetivos establecidos. Se aborda la estructura investigativa, el perfil de la población y la muestra seleccionada, así como los métodos y herramientas aplicados para la recopilación de información. Además, se exploran los aspectos relacionados con la validez y fiabilidad del estudio y se describen las estrategias empleadas para el análisis de los datos obtenidos.

Naturaleza de la investigación

Enfoque de la investigación

El enfoque de investigación está relacionado con la característica fundamental del examen y puede ser de carácter cuantitativo, explicativo o una mezcla de las dos.

Según la Escuela Superior de Guerra del Ejército - Escuela de Postgrado (2023-2025), el enfoque cuantitativo refiere:

La elaboración de la tesis es un trabajo de investigación académica basada en lineamientos del método científico, considerando la doctrina en los diferentes campos del dominio militar, con gran sentido de responsabilidad social y ética. Por lo que para su elaboración es necesario estandarizar procedimientos con el fin de garantizar su originalidad y rigor científico. (pág. 5)

En este contexto, el estudio actual se sitúa dentro de un método cuantitativo, ya que fue realizado de forma objetiva y los datos recopilados se analizaron estadísticamente, aplicando los hallazgos a toda la población del análisis.

Nivel de la investigación

La investigación cuantitativa se clasifica en cuatro categorías: Exploratoria, Descriptiva, Correlacional y Explicativa. En este caso, el estudio se situó en el nivel descriptivo, que se caracteriza por medir, analizar o reunir información sobre diversas características o elementos del fenómeno en cuestión, con el fin de detallar su comportamiento (Escuela Superior de Guerra del Ejército - Escuela de Postgrado ,2023-2025). De ese modo, se obtuvieron los datos a partir de la muestra del análisis, los cuales se describieron tal como fueron recopilados para entender los requisitos funcionales y no funcionales del sistema, así como su verificación.

Diseño de la investigación

La investigación cuantitativa incluye dos tipos de diseños: el experimental y el no experimental. Este último se caracteriza por no implicar la manipulación de la variable independiente y se enfoca en observar los fenómenos en su contexto natural. Además, se divide en dos categorías: transversales, cuando los datos son recopilados en un solo instante, y longitudinales, cuando se obtienen en diferentes períodos (Escuela Superior de Guerra del Ejército - Escuela de Postgrado, 2023-2025). Por lo tanto, este estudio adoptó un diseño no experimental y transversal, dado que no se llevó a cabo manipulación de las variables y la recolección de datos se realizó en un único momento.

Tipo de investigación

El trabajo de campo se describe como “la obtención de información de manera directa de las personas que están siendo estudiadas, o del entorno donde suceden los eventos (datos primarios), sin interferir o regular ninguna variable, lo que significa que el investigador recoge la información sin modificar las circunstancias actuales (...)” (Arias, 2016, pág. 31).

En este sentido, el presente estudio se correspondió con una investigación de campo, debido a que los datos fueron recolectados directamente de las personas que manejan el bienestar universitario en el área de medicina de la institución de educación superior.

Población

Dado que nuestro proyecto la población objetivo es accesible en su totalidad,

(Arias, 2012) menciona:

Si la población, por el número de unidades que la integran, resulta accesible en su totalidad, no será necesario extraer una muestra. En consecuencia, se podrá investigar u obtener datos de toda la población objetivo, sin que se trate estrictamente de un censo. Esta situación debe explicarse en marco metodológico, en el que se obviará la sección relativa a la selección de la muestra. (pág. 83).

No es necesario recolectar una muestra para realizar el estudio. La cantidad de elementos que forman la población es accesible y facilita la obtención de información directamente de cada uno de los componentes relevantes. Por lo tanto, se decidirá estudiar a toda la población sin llevar a cabo un censo oficial, lo que se detallará en la parte metodológica del proyecto, omitiendo la sección correspondiente a la elección de la muestra.

La población, según Porfirio (2020), se define como:

"elementos accesibles o unidad de análisis que pertenecen al ámbito especial donde se desarrolla el estudio" (pág. 02).

En el marco de nuestro proyecto, la población incluye a todos los componentes que pertenecen al área concreta de investigación, los cuales son significativos y disponibles para los fines establecidos.

En la presente investigación la población estuvo conformada de la siguiente manera:

1.- Personal Administrativo de la institución de educación superior:

- **Población:** Las personas que integran el Personal Administrativo de la institución de educación superior. Estos individuos desempeñan un papel fundamental en la gestión y operación diaria de documentos y procesos en la institución, y su perspectiva es esencial para comprender los requerimientos funcionales y no funcionales del sistema web propuesto.

2. Personal Médico de la institución de educación superior:

- **Población:** Los profesionales que conforman el equipo médico de la entidad educativa. Estos profesionales tienen un papel esencial como usuarios directos del sistema en línea y compartieron datos importantes y necesarios sobre las necesidades funcionales y no funcionales de dicho sistema.

Este formato garantiza que se adquiere un enfoque completo y verificado de los usuarios finales, asegurando que los hallazgos de la investigación sean pertinentes y útiles.

Técnicas e Instrumentos de Recolección de Datos

Técnica de Recolección de Datos

Se entiende por técnica de investigación, el procedimiento o forma particular de obtener datos (Arias, 2012). La técnica de recolección de datos que se aplicará en la presente investigación será la entrevista, la cual va a posibilitar la obtención de datos fundamentales para el desarrollo del sistema propuesto. Monje (2011) citado por Sánchez y Murillo (2022) destaca que entre los métodos de recolección de datos cuantitativos se encuentra la entrevista estructurada, que emplea como instrumento de medición un cuestionario,

administrado de manera estandarizada y secuencial a los entrevistados, este último sugiere que sea estructurado.

Operacionalización de la Variable

La operacionalización de la variable consiste en el establecimiento de significados para los términos utilizados en el proceso de investigación, y permite transformar dichos términos en situaciones observables y/o medibles; es decir, que el procedimiento de operacionalización de variable, permite transformar las variables abstractas y generales, en variables concretas y específicas, esto es observables y medibles (Rafael Rangel, 2018).

En este sentido, en la presente investigación se llevó a cabo el proceso de operacionalización de la variable, a partir de los objetivos específicos. Se derivó la variable de estudio y, posteriormente, a partir de su definición, se establecieron las dimensiones, indicadores e ítems, tal como se muestra en la Tabla 7.

Tabla 7. Operacionalización de Variables (Levantamiento de requerimiento/requisitos)

Variable	Definición Conceptual	Definición Operacional	Dimensiones	Indicadores	Ítem
Requerimientos funcionales del sistema para la gestión documental del área de medicina	Actividades específicas que el sistema debe cumplir para gestionar eficientemente la documentación del área de medicina.	Conjunto de funcionalidades específicas del sistema que permiten la gestión eficiente de la documentación.	Gestión de usuarios	Autentificación de usuario	1,2,3,4
			Gestión documental	Seguridad de documentos	5,6,7,8,9
				Registrar documentos	10,11,12
				Modificar documentos	13
				Eliminar documentos	14
			Gestión de archivos	Búsqueda de archivos	15
				Descargar archivos	16
			Seguridad	Calidad de Contraseñas	17
				Controles de acceso	18
				Encriptación	19
Requerimientos no funcionales del sistema para la gestión documental del área de medicina.	Restricciones en los servicios del sistema, abarcando restricciones vinculadas al tiempo de operación, la fase de elaboración y el respeto a normas.	Conjunto de restricciones y características que determinan el rendimiento y calidad del sistema.	Rendimiento	Cantidad de usuarios concurrentes	20
			Usabilidad	Facilidad de uso	21,22,23

Tabla 8. Operacionalización de Variables (Validación del producto/software)

Variable	Definición Conceptual	Definición Operacional	Dimensiones	Indicadores	Ítem
Validar el funcionamiento del sistema web para la gestión documental del área de medicina	La prueba de software es el proceso de ejecución de un programa con la intención de encontrar errores. (Myers y Sanders, 2004)	La prueba de software es el proceso de ejecución de un programa con la intención de encontrar errores. (Myers y Sanders, 2004)	Rendimiento	Tiempo de respuesta del sistema	1
			Fiabilidad	Frecuencia de errores	2
			Disponibilidad	Porcentaje de disponibilidad	3
			Mantenibilidad	Documentación del código	4

Instrumentos de Recolección de Datos

Los métodos para recopilar información están diseñados para crear un entorno favorable que permita realizar mediciones exactas. Los conceptos teóricos que simbolizan los datos tienen su origen en la realidad física y perceptible, y pueden ser experimentados de forma directa o indirecta mediante los sentidos. De esta manera, cualquier elemento empírico puede ser medido. Así, las tácticas y acciones utilizadas en la recopilación de información forman los pasos que facilitan al investigador reunir los datos necesarios para responder a la cuestión de investigación de forma exacta. (Hernández, 2020).

El cuestionario se utiliza de manera muy común, así que es importante entender su propósito y atributos. Los cuestionarios son una serie de preguntas, abiertas y cerradas o ambas, dirigidas a estudiar una o más variables, siendo posiblemente el instrumento más comúnmente utilizado para recopilar datos (Hernández, 2018).

Para obtener información en este estudio, se utilizó un formulario que contenía 23 preguntas de respuesta cerrada con el fin de identificar los requisitos tanto funcionales como no funcionales del sitio web.

Validez y Confiabilidad

Según Castillo et al. (2021), la validez:

Corresponde al grado en que un instrumento mide lo que pretende medir. Una vez que ya se tiene claramente definidas de esta forma se garantiza la validez del instrumento. Dentro de los tipos de validez se encuentran: Validez de Contenido, Validez de Constructo (convergente, divergente y factorial) y Validez Predictiva; también se puede recurrir a la ayuda de personas expertas en el tema para que revisen el instrumento (juicio de expertos), a fin de determinar si cumple con la finalidad establecida. (pág. 77)

La validación del instrumento va a ser llevada a cabo por expertos en software e investigación de la institución de educación superior. Estos especialistas deben revisar el cuestionario y contribuirán con su valiosa experiencia para mejorarlo.

Tabla 9. Validadores del instrumento de investigación.

Cuestionario	
Expertos	Título profesional
Byron Moreno	Ingeniero en Sistemas
Sandino Jaramillo	Ingeniero en Sistemas

Por su parte, la confiabilidad según Castillo et al. (2021):

Se refiere a la capacidad del instrumento para arrojar datos o mediciones que correspondan a la realidad que se pretende conocer, o sea, la exactitud de la medición, así como la consistencia o estabilidad de la medición en diferentes momentos. Mientras más grande sea el nivel de confiabilidad de un instrumento, menor es la cantidad de error presente en los puntajes obtenidos (...). (pág. 77)

Por lo tanto, es fundamental garantizar un alto nivel de confiabilidad en los instrumentos de medición. Esto asegura que los datos recolectados sean precisos y consistentes a lo largo del tiempo, minimizando los errores en los resultados obtenidos y reflejando fielmente la realidad que se desea conocer.

Técnicas de Análisis de Datos

Las técnicas de análisis de datos son procesos y estrategias que se utilizan para examinar, ajustar e interpretar información con la finalidad de extraer datos útiles, identificar tendencias y tomar decisiones fundamentadas. Estas prácticas pueden aplicarse en diversas disciplinas como la estadística, el análisis comercial, la ciencia de datos y la investigación científica.

Según el artículo "Importancia del Análisis de Datos en la Toma de Decisiones" de Conexión ESAN:

El análisis de datos permite predecir tendencias futuras del mercado y responder rápidamente a ellas, ofreciéndole la posibilidad de una mejor ventaja competitiva a la empresa en su mercado objetivo. Para esto, es necesario que la compañía tenga la capacidad organizacional de la "agilidad". El análisis de datos brinda el conocimiento, pero si la organización no puede responder a ello, su adaptación no va a ocurrir. (*Importancia del Análisis de Datos En la Toma de Decisiones / Conexión ESAN*, 15 de octubre de 2021)

Después de analizar las diferentes técnicas disponibles, decidimos utilizar el análisis descriptivo y exploratorio de datos para examinar la información recopilada. Este enfoque nos permitirá obtener una visión clara y estructurada de las respuestas, identificando tendencias, patrones y distribuciones en los datos obtenidos a través de Google Forms.

Para conseguir esto, es fundamental contar con datos precisos y representativos, ya que su correcta interpretación nos ayudará a alcanzar conclusiones relevantes y sólidas. A través de gráficos y datos generados automáticamente, podremos visualizar la información de modo claro, lo que hará más sencillo tomar decisiones y desarrollar estrategias basadas en datos.

Metodología del producto

Para el desarrollo del sistema web relacionado con este proyecto de titulación, se ha adoptado un enfoque en cascada con la intención de mantener un proceso ordenado y progresivo. Esta estrategia asegura que la entrega de la aplicación se realice de forma programada, segmentando el proyecto en etapas bien delimitadas y concretas. Es necesario finalizar cada etapa antes de avanzar a la siguiente, lo que asegura que todos los requisitos y especificaciones sean cumplidos de manera adecuada. De este modo, se promueve un desarrollo del sistema que es tanto metódico como controlado, contribuyendo al cumplimiento de los objetivos planteados.

Según Delgado Olivera, Lisdania de la Caridad; Díaz Alonso, Lexys Manuel (2021), refiere que:

Este modelo toma las actividades fundamentales del proceso de especificación, desarrollo, validación y evolución y las representa como fases separadas del proceso. (pág. 11)

La metodología en cascada desempeña un papel fundamental en el proyecto del sistema web para la gestión de historias clínicas al brindar una serie de beneficios significativos. En primer lugar, la metodología en cascada ofrece un enfoque estructurado que permite una organización clara y secuencial de las fases del proyecto. Esto facilita la planificación detallada y la documentación exhaustiva en cada etapa del desarrollo. Al seguir un proceso lineal, se garantiza que cada fase esté completamente desarrollada y validada antes de pasar a la siguiente, promoviendo así un control riguroso y la reducción de riesgos a lo largo del proyecto, como lo menciona Alava García y Suarez Vera (2022):

La primordial característica del modelo de cascada es que sigue una sucesión lineal, esto posibilita detectar unas fases concretas a las cuales se les debería ofrecer cumplimiento en orden, a medida tal que se van aplicando las adecuaciones pertinentes y las pruebas para mitigar los errores que se logren exponer... (pág. 33)

Una vez establecidos los elementos mencionados anteriormente, el proyecto actual se organiza en torno a un conjunto de fases secuenciales que se centran en las funcionalidades necesarias para el sistema web de la historia clínica. El autor del proyecto actúa como el gerente del proyecto y tiene la autoridad para planificar y coordinar las actividades clave en cada fase del ciclo de vida del desarrollo. A partir de ahí, se definen los requisitos detallados y se realiza una estimación del esfuerzo necesario para cada etapa. Durante cada fase, se llevan a cabo revisiones para monitorear el progreso y asegurar que se cumplan los objetivos establecidos.

Durante el desarrollo de este proyecto, se organizaron encuentros con los principales beneficiarios del sistema web de gestión documental. Estas reuniones tuvieron lugar en el centro médico, así como en los campus de la Universidad. El propósito de estos encuentros fue recopilar información y obtener datos relevantes sobre aspectos cruciales del sistema. Adicionalmente, se efectuaron encuestas que proporcionan resultados precisos, los cuales se describen en el capítulo IV de este estudio.

Al finalizar la etapa de desarrollo, se lleva a cabo una revisión final en la que se analizan los resultados obtenidos. En este punto, se examinan los logros conseguidos y se identifican áreas de mejora para futuros proyectos. La metodología lineal garantiza que este proceso se realice de forma ordenada, permitiendo un seguimiento minucioso y una documentación exhaustiva en cada etapa, lo que contribuye a la efectividad y calidad del sistema web diseñado para la gestión documental.

CAPÍTULO IV

ANÁLISIS E INTERPRETACIÓN DE LOS RESULTADOS

En este capítulo se presentan los resultados obtenidos durante el desarrollo de un sistema web de bienestar universitario para la gestión documental del área de medicina en una institución de educación superior. Esta solución tecnológica fue diseñada para optimizar la administración de documentos y procedimientos médicos, mejorando la eficiencia operativa en el ámbito universitario.

A lo largo del proceso de desarrollo, se alcanzaron de manera efectiva todos los objetivos concretos que se habían establecido:

Requerimientos funcionales y no funcionales del sistema

Se identificaron las características tanto funcionales como no funcionales del sistema a través de encuestas dirigidas al grupo de Bienestar Universitario de la Institución de Educación Superior. Esta tarea permitió recolectar información fundamental para el diseño y desarrollo del sistema, asegurando que la propuesta cumpla con las necesidades operativas y preferencias de los usuarios. Metodología Seleccionada

Diseño del sistema web

La elaboración del sistema en línea se realizó considerando los requisitos que se habían señalado anteriormente, empleando Figma para el desarrollo de prototipos dinámicos. Esta etapa facilitó la comprobación de la operatividad del sistema con los usuarios esenciales antes de su construcción, garantizando que la interfaz y la vivencia del usuario fueran claras y efectivas.

Desarrollo del sistema web

Para llevar a cabo la instauración del sistema, se utilizaron herramientas modernas que maximizan el uso del personal y mecanizan las tareas administrativas. Esto no solo incrementó la eficacia operativa de la organización, sino que también asegura un sistema resistente, ampliable y que se ajusta a requerimientos venideros.

Validación del sistema

El sistema fue comprobado junto al equipo de Bienestar Universitario con el fin de confirmar que satisfacía las necesidades y requerimientos del área, asegurando su efectividad y operatividad en el contexto universitario.

Resultados de la Investigación

La verificación del sistema fue realizada en asociación con el equipo de Bienestar Universitario, buscando asegurar que la solución satisficiera sus necesidades y expectativas. Para esto, se implementó una encuesta para validar el producto, que sirvió para medir el funcionamiento del sistema en un contexto real. Los resultados alcanzados hicieron posible confirmar su efectividad, eficiencia y habilidad para mejorar la administración universitaria.

Análisis de los Resultados

En esta parte se muestran los resultados derivados de las encuestas realizadas a los usuarios finales en dos momentos importantes: la recogida de requerimientos y la comprobación del software. Durante la primera etapa, se establecieron las exigencias concretas del sector de la salud, además de las características fundamentales para el sistema web de administración documental. Estas encuestas facilitaron el estudio de los métodos actuales de gestión documental y los problemas asociados con la clasificación y el manejo de la información médica.

En la segunda etapa, se llevó a cabo la verificación del software con los usuarios finales, con el propósito de analizar su funcionalidad y asegurar que cumplía con las demandas del sistema, garantizando así su efectividad y su capacidad de ajuste a las necesidades de los usuarios.

Resultados de la encuesta sobre requerimientos del sistema

1.- ¿El sistema debe permitir la autenticación de usuarios mediante correo electrónico y contraseña?

3 respuestas



Figura 1. Datos de autentificación mediante correo y contraseña.

Los tres encuestados (100%) están de acuerdo en que el sistema debe permitir la autenticación de usuarios mediante correo electrónico y contraseña.

2.- ¿El sistema debe soportar la recuperación de contraseñas mediante correo electrónico?

3 respuestas



Figura 2. Datos referentes a contraseñas.

Los tres encuestados (100%) están de acuerdo en que el sistema debe soportar la recuperación de contraseñas mediante correo electrónico.

3.- ¿El sistema debe permitir a los usuarios actualizar su correo electrónico o contraseña desde su perfil?

3 respuestas

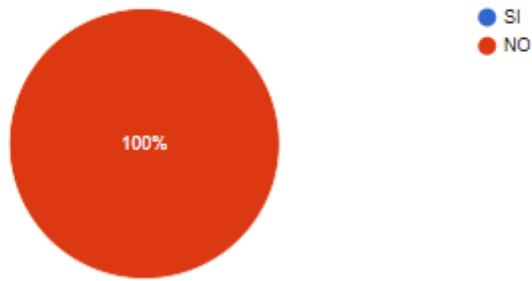


Figura 3. Datos de actualización de datos del usuario.

Los tres encuestados (100%) responden que "NO", indicando que no se debe permitir a los usuarios actualizar su información de correo electrónico o contraseña desde su perfil.

4.- ¿El sistema debe mantener la sesión del usuario activa después de cerrar la aplicación (sesión persistente)?

3 respuestas

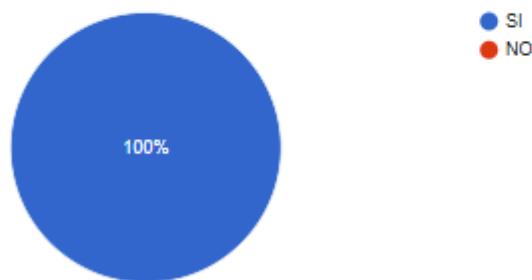


Figura 4. Datos sobre la persistencia de la sesión.

Los tres encuestados (100%) están de acuerdo en que el sistema debe mantener la sesión del usuario activa después de cerrar la aplicación.

5.- ¿El sistema debe permitir asignar permisos específicos a cada usuario para controlar el acceso a documentos?

3 respuestas

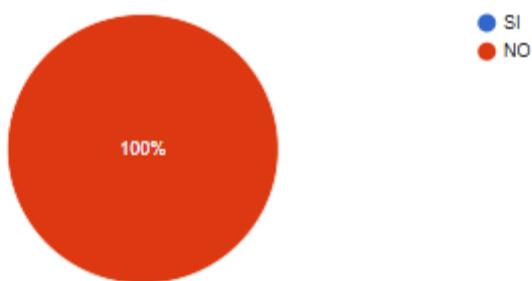


Figura 5. Datos Sobre permisos para controlar documentos.

Los tres encuestados (100%) responden que "NO", indicando que no se debe permitir asignar permisos específicos a cada usuario.

6.- ¿El sistema debe manejar diferentes niveles de acceso a los documentos (por ejemplo, lectura, edición, eliminación)?

3 respuestas

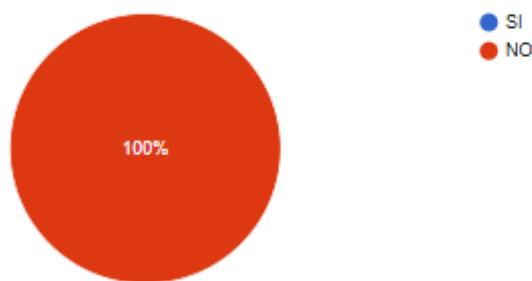


Figura 6. Datos Niveles de acceso a documentos.

Los tres encuestados (100%) responden que "NO", indicando que no se debe manejar diferentes niveles de acceso.

7.- ¿El sistema debe permitir el control de versiones de los documentos para mantener un historial de cambios?

3 respuestas

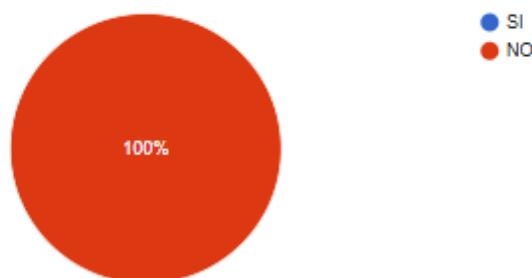


Figura 7. Datos sobre las versiones de los documentos.

Los tres encuestados (100%) responden que "NO", indicando que no se debe permitir control de versiones.

8.- ¿El sistema debe permitir el seguimiento del estado de los documentos (borrador, revisión, aprobado, archivado)?

3 respuestas



Figura 8. Datos sobre seguimiento de documentos.

Los tres encuestados (100%) están de acuerdo en que el sistema debe permitir el seguimiento del estado de los documentos.

9.- ¿El sistema debe permitir que los documentos sean compartidos con usuarios específicos o grupos?

3 respuestas

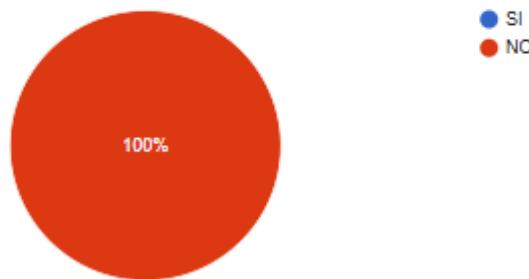


Figura 9. *Métodos para compartir documentos en el sistema.*

Los tres participantes (100%) responden con un "NO", lo que indica que no se debería facilitar el intercambio de documentos.

10.- ¿El sistema debe permitir el registro de nuevos documentos en la base de datos?

3 respuestas

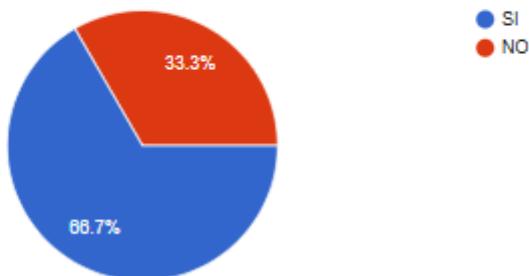


Figura 10. *Métodos para registrar documentos en la base de datos.*

Dos de los tres encuestados (66.67%) están de acuerdo en que el sistema debe permitir el registro de nuevos documentos.

11.- ¿El sistema debe permitir el registro de metadatos al momento de la carga de los archivos?

3 respuestas



Figura 11. *Métodos para registrar metadatos en la carga de archivos.*

Los tres encuestados (100%) están de acuerdo en que el sistema debe permitir el registro de metadatos.

12.- ¿El sistema debe permitir la categorización de documentos por tipo y fecha?

3 respuestas



Figura 12. *Métodos para categorizar los documentos.*

Los tres encuestados (100%) están de acuerdo en que el sistema debe permitir la categorización de documentos por tipo y fecha.

13.- ¿El sistema debe permitir la modificación de documentos existentes?

3 respuestas

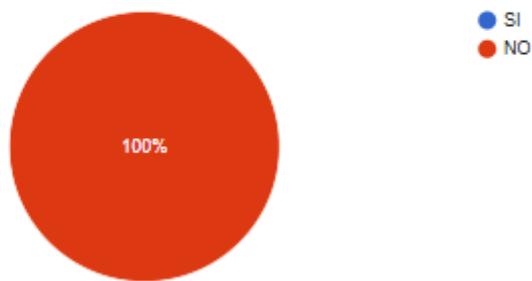


Figura 13. Métodos para permitir la modificación de documentos.

Los tres encuestados (100%) responden que "NO", indicando que no se debe permitir la modificación de documentos.

14.- ¿El sistema debe permitir la eliminación de documentos?

3 respuestas

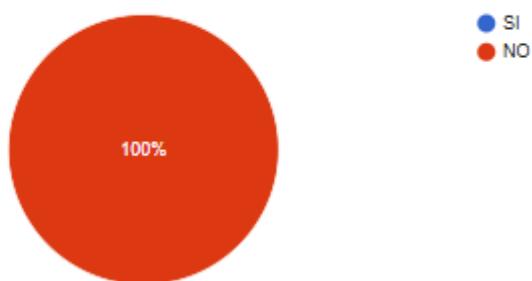


Figura 14. Métodos para permitir la eliminación de documentos.

Los tres encuestados (100%) contestan que "NO", lo que indica que la eliminación de documentos no debe ser aceptada.

15.- ¿El sistema debe permitir la búsqueda de documentos mediante palabras clave?

3 respuestas



Figura 15. Métodos para permitir la búsqueda de documentos.

Los tres encuestados (100%) están de acuerdo en que el sistema debe permitir la búsqueda de documentos mediante palabras clave.

16.- ¿El sistema debe permitir la descarga de archivos?

3 respuestas

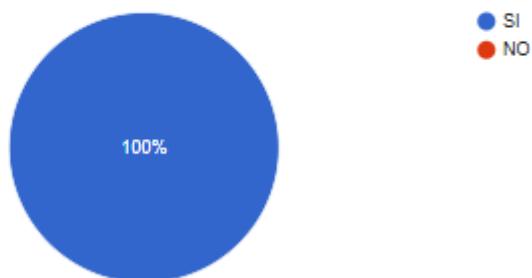


Figura 16. Métodos para descargar documentos.

Los tres encuestados (100%) están de acuerdo en que el sistema debe permitir la descarga de archivos.

17.- ¿El sistema debe requerir contraseñas fuertes para la autenticación de usuarios?

3 respuestas



Figura 17. Métodos para permitir validaciones de contraseñas fuertes.

Los tres encuestados (100%) están de acuerdo en que el sistema debe requerir contraseñas fuertes.

18.- ¿El sistema debe tener niveles de permisos para usuarios (por ejemplo, administrador, usuario regular)?

2 respuestas

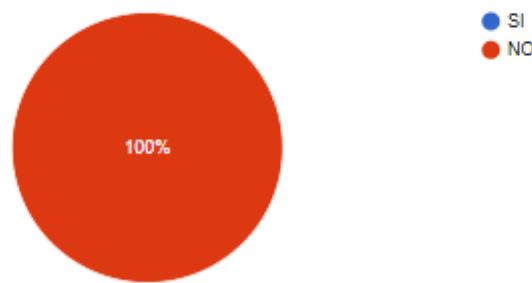


Figura 18. Métodos para permitir niveles de acceso para los usuarios.

Los tres encuestados (100%) responden que "NO", indicando que no se debe tener niveles de permisos.

19.- ¿El sistema debe cifrar los datos sensibles almacenados?

3 respuestas



Figura 19. *Métodos para cifrar datos sensibles.*

Los tres encuestados (100%) están de acuerdo en que el sistema debe cifrar los datos sensibles.

20.- ¿El sistema debe ser capaz de manejar al menos 100 usuarios concurrentes sin degradar su rendimiento?

3 respuestas

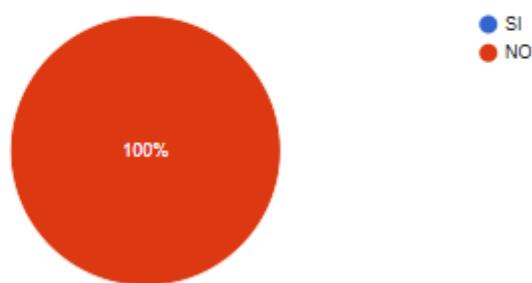


Figura 20. *Métodos para que el sistema sea capaz de manejar 100 usuarios.*

Los tres encuestados (100%) responden que "NO", indicando que el sistema no necesita manejar 100 usuarios concurrentes.

21.- ¿El sistema debe ser accesible desde dispositivos móviles?

3 respuestas

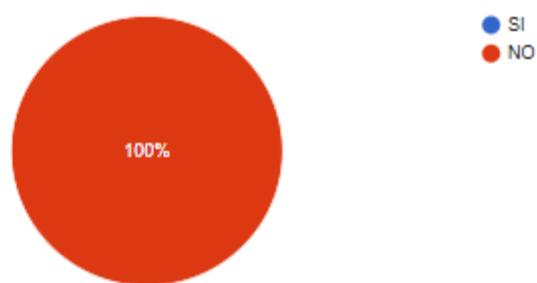


Figura 21. Accesos a dispositivos móviles.

Los tres encuestados (100%) responden que "NO", indicando que el sistema no necesita ser accesible desde dispositivos móviles.

22.- ¿El sistema debe ser fácil de usar y entender para usuarios con conocimientos básicos de informática?

3 respuestas



Figura 22. Datos para obtener la facilidad de uso del sistema.

Los tres encuestados (100%) están de acuerdo en que el sistema debe ser fácil de usar.

23.- ¿El sistema debe proporcionar mensajes de error claros y detallados en caso de fallos?

3 respuestas



Figura 23. Datos para obtener el manejo de errores.

Los tres encuestados (100%) están de acuerdo en que el sistema debe proporcionar mensajes de error claros.

Según los hallazgos del estudio de necesidades llevado a cabo, es posible deducir las conclusiones que se enumeran a continuación para la puesta en marcha del sistema:

Requerimientos Funcionales Clave:

- A. Autenticación: Hay acuerdos comunes sobre que el sistema debería facilitar la comprobación de los usuarios utilizando su dirección de correo electrónico y una clave.
- B. Recuperación de contraseñas: Todos los encuestados coinciden en que el sistema necesita poder recuperar contraseñas a través del correo electrónico.
- C. Sesión Persistente: Es fundamental preservar la actividad de la sesión del usuario una vez que la aplicación se ha cerrado.
- D. Seguimiento de Estado: Es unánime la necesidad de permitir el seguimiento del estado de los documentos a través de diferentes etapas (borrador, revisión, aprobado, archivado).
- E. Registro de Metadatos: Existe acuerdo en que el sistema debe permitir el registro de metadatos al momento de la carga de archivos.
- F. Categorización: La categorización de documentos por tipo y fecha es un requerimiento unánime.
- G. Búsqueda: Es fundamental permitir la búsqueda de documentos mediante palabras clave.
- H. Descarga: Todos coinciden en que el sistema debe permitir la descarga de archivos.

Requerimientos No Funcionales Clave:

- A. Contraseñas Fuertes: Se considera indispensable requerir contraseñas robustas para la autenticación de usuarios.
- B. Cifrado de Datos: Existe acuerdo unánime en que el sistema debe cifrar los datos sensibles almacenados.
- C. Facilidad de Uso: Es crucial que el sistema sea sencillo de manejar y comprensible para personas con habilidades básicas en tecnología.
- D. Mensajes de Error: Es fundamental que el sistema ofrezca errores claros y fáciles de comprender.

Tabla 10. *Porcentaje de respuestas afirmativas y negativas de la encuesta realizada para obtener los requerimientos funcionales y no funcionales.*

Tipo de Respuesta	Número de Preguntas	Porcentaje	Conclusión
Afirmativas	13	56.52%	Más de la mitad de las interrogantes recibió respuestas positivas, lo que indica una inclinación notable hacia las características presentadas en la encuesta.
Negativas	10	43.48%	Una cantidad considerable de preguntas recibe respuestas negativas, subrayando características que no se perciben como esenciales para el sistema.

Resultados de la encuesta sobre la validación del sistema

1.- ¿El sistema funciona correctamente bajo pruebas de carga y estrés?

3 respuestas



Figura 24. Funcionamiento correcto bajo pruebas.

Los tres encuestados (100%) están de acuerdo en que el sistema funciona correctamente bajo pruebas de carga y estrés.

2.- ¿El sistema replica de manera confiable las funcionalidades y navegación del prototipo, asegurando que todos los flujos de interacción y componentes visuales funcionen de forma consistente y sin errores?

3 respuestas



Figura 25. Funcionamiento de forma consistente y sin errores.

Los tres involucrados (100%) afirman que el sistema refleja de manera precisa las características y la navegación del modelo, asegurando que todos los procesos de interacción y componentes visuales funcionen de manera coherente y sin errores.

3.- ¿El sistema está disponible al menos el 95% del tiempo?

3 respuestas



Figura 26. Disponibilidad.

Los tres encuestados (100%) confirman que el sistema está disponible al menos el 95% del tiempo.

4.- ¿El sistema tiene documentación completa para desarrolladores y usuarios?

3 respuestas



Figura 27. Documentación.

Los tres involucrados (100%) en el sondeo afirman que el sistema cuenta con una documentación completa destinada a desarrolladores y usuarios.

El sistema examinado ha mostrado ser sólido, fiable y correctamente documentado, según lo indicado por las respuestas de los encuestados. En esta revisión no se hallaron elementos desfavorables, lo que indica que el producto está preparado para su puesta en marcha sin requerir ajustes esenciales.

Tabla 11. Porcentaje de respuestas afirmativas y negativas de la encuesta realizada para la validación del producto/software.

Tipo de Respuesta	Número de Preguntas	Porcentaje	Conclusión
Afirmativas	4	100%	Todas las preguntas recibieron contestaciones afirmativas, lo que indica que el sistema es sólido, confiable y tiene un buen desempeño.
Negativas	0	0%	No se presentaron preguntas negativas, lo que sugiere que no hay aspectos esenciales que necesiten ajustes en esta revisión.

Metodología

El sistema fue desarrollado utilizando un enfoque de desarrollo en cascada. Esto implica un proceso secuencial donde cada fase debe completarse antes de que comience la siguiente. Las fases típicamente incluyen:

1. **Recopilación de Requerimientos:** Identificación de requerimientos funcionales y no funcionales a través de encuestas y consultas con las partes interesadas.
2. **Diseño:** Diseño de la arquitectura del sistema, el esquema de la base de datos y la interfaz de usuario.
3. **Implementación:** Escritura del código y construcción del sistema.
4. **Pruebas:** Pruebas del sistema para asegurar que cumple con los requerimientos y está libre de errores.
5. **Despliegue:** Despliegue del sistema en un entorno de producción.
6. **Mantenimiento:** Provisión de mantenimiento y soporte continuo.

Planificación

Dado el uso de la metodología en cascada, la fase de planificación involucra:

- **Definición del Alcance del Proyecto:** Definir claramente los objetivos y entregables del sistema.
- **Creación de una Línea de Tiempo:** Definir un calendario para cada etapa del ciclo de desarrollo.

- **Asignación de Recursos:** Asignar recursos (desarrolladores, testers, hardware, software) a cada tarea.
- **Gestión de Riesgos:** Identificar posibles riesgos y desarrollar estrategias de mitigación.
- **Definición de Entregables:** Definir los resultados específicos de cada fase.

Prototipos

Se desarrollaron prototipos del sistema utilizando la herramienta digital Figma, que permitió la creación de representaciones visuales interactivas y detalladas de las diferentes vistas del sistema. A lo largo del proceso, se elaboraron varias versiones de los prototipos con el fin de explorar diversas perspectivas en cuanto a diseño, interactividad y el flujo del usuario. Figma ofreció una plataforma colaborativa que facilitó la retroalimentación continua de los miembros del equipo, permitiendo la actualización y perfeccionamiento de los prototipos de manera instantánea. Gracias a sus capacidades avanzadas de prototipado y a sus herramientas de diseño intuitivas, se logró una representación precisa de la experiencia del usuario, lo que mejoró la calidad de las decisiones para la implementación final del sistema. En el anexo 6 se incluye un listado de los prototipos desarrollados.

Desarrollo

De acuerdo con lo que se eligió en los fundamentos teóricos, se procederá de la siguiente manera: la selección de los instrumentos y tecnologías se basa en los principios, cualidades y necesidades que se mencionaron anteriormente, garantizando que cada parte del sistema satisfaga los requerimientos de escalabilidad, rendimiento y eficacia operativa. A continuación, se describen los instrumentos utilizados en cada parte:

- **Backend:** AWS Amplify proporciona una opción eficiente y ampliable para manejar el backend, incorporando servicios como autenticación, almacenamiento y APIs de manera simple. En relación a Firebase, Amplify ofrece mayor control y flexibilidad sobre la infraestructura, particularmente en aplicaciones de gran tamaño, y se conecta sin dificultad con otros servicios de AWS. En contraste con Laravel, Amplify disminuye el tiempo necesario para el desarrollo al presentar servicios listos para utilizar, mientras que Laravel demanda más ajustes manuales. Optamos por AWS Amplify debido a su velocidad en la implementación, capacidad de escalabilidad y la sencillez en la gestión de servicios en la nube.

- **Base de datos:** DynamoDB es una plataforma de base de datos NoSQL que proporciona rapidez y seguridad, perfecta para iniciativas que requieren crecimiento. A diferencia de bases de datos como PostgreSQL y MySQL, que funcionan con esquemas de datos fijos, DynamoDB facilita el manejo de información desestructurada y semi desestructurada de manera más adaptable. Elegimos DynamoDB por su capacidad de escalar por sí mismo, su integración con AWS y su bajo coste de mantenimiento.
- **Frontend:** Se eligió a React debido a su habilidad para desarrollar interfaces que son tanto interactivas como adaptables, ofreciendo una experiencia de usuario efectiva y cautivadora. A diferencia de Angular, que es un marco de trabajo completo con una estructura más estricta y una curva de aprendizaje más alta, React se presenta como una biblioteca más ligera y adaptable, adecuada para iniciativas que requieren rapidez en la creación y mayor control sobre el manejo de datos. Optamos por React por su facilidad de uso, su amplia comunidad, y la capacidad de integrarse sin problemas con otras bibliotecas o marcos, conforme a las exigencias del proyecto, lo que facilita un desarrollo más rápido y personalizado en contraste con Angular.

Validación

La validación del sistema consistió en evaluaciones realizadas por el equipo del departamento de salud del área de Bienestar Universitario. Esto se realizó con el propósito de:

- **Asegurar la Funcionalidad:** Verificar que el sistema funcione según lo establecido y cumpla con las condiciones definidas.
- **Evaluar la Usabilidad:** Establecer si el sistema es sencillo de usar y comprender para los usuarios finales.
- **Confirmar el Impacto Positivo:** Asegurar que el sistema tenga un impacto beneficioso en la gestión administrativa y aumente su efectividad.
- **Documentación usuarios:** los usuarios pudieron visualizar la documentación de la aplicación.
- **Comprobar que el sistema estaba disponible:** Esto se realizó mediante las pruebas ya que el sistema estuvo disponible una semana en la cual los usuarios no reportaron ninguna novedad.

Planificación del Proyecto de Software

Recursos Humanos

Los recursos humanos desempeñan un papel fundamental en el desarrollo de software. A continuación, se detallan los profesionales que participaron y se mantuvieron en el proyecto:

Tabla 12. *Responsable del trabajo de titulación.*

Recursos Humanos	Cargo
Steveen Ordoñez	Desarrollador
Erick Garcia	Desarrollador

Durante la creación de este proyecto, se emplearon diferentes herramientas tanto de hardware como de software para satisfacer las necesidades del desarrollo y la plataforma web. Estos elementos aseguran el éxito del desarrollo. A continuación, se especifican los requerimientos del hardware utilizado en el proceso de desarrollo.

Tabla 13. *Detalles del Hardware usado para el desarrollo.*

Equipo	
Procesador	AMD Ryzen 7 5700G with Radeon Graphics, 3.80 GHz
Memoria RAM	8,00 GB
Sistema operativo	Windows 11
Memoria de almacenamiento	Disco SSD 500Gbs NVMe 500Gbs

Recursos de Software

Para asegurar que la aplicación funcione correctamente, se utilizaron varias herramientas y servicios informáticos, que incluían la creación del sistema en línea. Dentro de los recursos utilizados, destacan las soluciones que ofrece AWS, aprovechando su opción sin costo. Estas herramientas ayudan a reducir los gastos a lo largo del desarrollo.

Tabla 14. Detalles del software utilizado para el desarrollo.

Herramientas/servicios	Descripción	Uso específico
AWS Amplify	Plataforma de desarrollo en la nube que simplifica la creación de aplicaciones web y ofrece todos los servicios de AWS y un backend dedicado.	Configuración de backend (autenticación, almacenamiento).
AWS EC2	Servicio de computación en la nube en la capa gratuita.	Implementación de la aplicación para llevar a cabo pruebas de la misma.
React	Biblioteca de JavaScript para el desarrollo de interfaces de usuario.	Desarrollo del frontend de la aplicación web.
Git	Sistema de control de versiones.	Gestión del código fuente y trabajo en equipo.
Visual Studio Code	Editor de texto y entorno de desarrollo integrado.	Edición y depuración del código a lo largo de la fase de desarrollo.
Figma	Instrumento para el diseño y desarrollo de interfaces de usuario.	Creación de prototipos interactivos del sistema web. Diseño de la interfaz de usuario asegurando accesibilidad y usabilidad.
Jira	Software de gestión de proyectos y seguimiento de tareas.	Planificación y asignación de tareas para el desarrollo. Seguimiento del progreso de las funcionalidades del sistema.

Implementación en AWS

Para la implementación y administración de los servicios en la nube, se aplicaron las configuraciones que siguen:

- Se utilizó **AWS Amplify** para supervisar todo el proceso de desarrollo, abarcando la configuración de la autenticación y el almacenamiento de información.
- **EC2 en la capa gratuita** permitió utilizar una instancia **t2.micro**, lo que fue suficiente para manejar la computación y el tráfico de pruebas de la aplicación.
- **DynamoDB** se integraron como soluciones de almacenamiento y bases de datos, respectivamente, para el manejo eficiente de la información del proyecto.

Estimación del Proyecto

La valoración en el ámbito de proyectos informáticos es una tarea esencial que demanda un examen exhaustivo del proyecto, una organización preliminar meticulosa y un entendimiento claro de los recursos disponibles. Gómez y López indican que "Una de las actividades más cruciales en la gestión de proyectos de software es la valoración de costos. Aunque se lleva a cabo entre las primeras acciones, justo después de definir los requisitos, se aplica continuamente a medida que avanza el proyecto para mejorar la exactitud en la valoración" (pág. 4). En este marco, el trabajo actual expone el modelo de valoración conocido como COCOMO II, desarrollado por Barry Boehm. Este modelo facilita la realización de valoraciones basadas en el tamaño del software y un conjunto de factores de costo y de escalabilidad.

El modelo COCOMO II se utilizó para realizar la evaluación del esfuerzo, tiempo y costo en la elaboración de software, adaptándose a métodos actuales y proyectos más ágiles. En este esquema, los actores en el procedimiento son el gestor del proyecto y el analista de costos.

A continuación, se utilizó la segunda fase del modelo COCOMO II (Early Design Model) con el objetivo de realizar una estimación precisa del esfuerzo necesario. Para ello se aplicó la siguiente fórmula:

$$PM = AAA \times (KSLOC)^B \times \prod EM_i$$

Donde:

AA (Coeficiente de calibración) = 2.94

KSLOC (Kilo Source Lines of Code) = 200 (que equivale a 0.2 KLOC)

BB (Exponente de escala) = 1.12

EM1 (Factores de esfuerzo - Effort Multipliers) = 3.67

EM2 (Factores de esfuerzo - Effort Multipliers) = 1.0

EM3 (Factores de esfuerzo - Effort Multipliers) = 1.2

$$PM \approx 2.94 \times 0.158 \times 3.67 \times 1.0 \times 1.2$$

$$PM \approx 2.94 \times 0.158 \approx 0.46572$$

$$PM \approx 0.46572 \times 3.67 \approx 1.71$$

$$PM \approx 1.71 \times 1.2 \approx 2.052$$

R = 2.05 meses-persona

Después de calcular el esfuerzo en persona-meses, PM nos permitió estimar:

Duración del proyecto (TDEV):

$$TDEV = C \times (PM)^D$$

C (Coeficiente de calibración para la duración) = 2.5

D (Exponente de escala para la duración) = 0.35

$$TDEV = 2.5 \times 1.29 \approx 3.225$$

R = 3.23 meses

Tamaño del equipo:

$$\text{Tamaño del Equipo} = \frac{PM}{TDEV}$$

$$\text{Tamaño del Equipo} = \frac{2.05}{3.23} \approx 0.635$$

R = 0.64 personas

Esfuerzo estimado (PM): 2.05 meses-persona

Duración estimada (TDEV): 3.23 meses

Tamaño del equipo: 0.64 personas

La estimación del esfuerzo total requerido para la ejecución del proyecto es de aproximadamente 2.05 meses-persona, lo que sugiere que se necesitarán alrededor de 2 meses de trabajo de una persona o su equivalente en un equipo. Sin embargo, la duración proyectada del proyecto es de 3.23 meses, lo que indica que, aunque el esfuerzo es relativamente bajo, la finalización de las tareas podría extenderse debido a la naturaleza del trabajo y la disponibilidad del equipo.

Respecto a la cantidad de miembros en el grupo, se estima que se requieren cerca de 0.64 personas, lo que sugiere que una sola persona podría llevar a cabo todo el proyecto. Sin embargo, se sugiere que se explore la opción de obtener apoyo extra para garantizar la calidad del trabajo y cumplir con los plazos establecidos.

Ruta del proyecto

Para representar de manera visual el plan de actividades del proyecto, se elaboró un diagrama de esfuerzo y duración que detalla la distribución del esfuerzo estimado (en meses-persona) y el tiempo de desarrollo (TDEV en meses). Este diagrama proporciona una visualización clara de cómo se asignan los recursos a lo largo de las iteraciones o fases del proyecto.

Tabla 15. Ruta del proyecto.

Fase del Proyecto	Duración (Meses)	Esfuerzo (Meses-Persona)	Actividades
Fase de Planificación	0.5	0.5	Establecimiento de requerimientos, diseño inicial.
Fase de Diseño Detallado	1.0	1.0	Desarrollo detallado, documentación técnica.
Fase de Implementación	1.5	1.0	Codificación, pruebas unitarias.
Fase de Pruebas y Ajustes	0.23	0.23	Pruebas integrales, ajustes finales.
Total	3.23 meses	2.05 meses-persona	

La tabla presentada muestra una distribución clara y detallada del esfuerzo y la duración estimados para cada fase del proyecto, de acuerdo con el modelo COCOMO II. A través de esta tabla, se puede observar que el esfuerzo total estimado es de 2.05 meses-persona, mientras que la duración total del proyecto se proyecta en 3.23 meses. Las etapas del proyecto, como la Planificación, Diseño Detallado, Implementación y Pruebas y Ajustes, están claramente establecidas con sus correspondientes tiempos y esfuerzos.

Herramientas de gestión del proyecto

En esta investigación, se seleccionará Jira como la herramienta para llevar a cabo la gestión de la implementación de la aplicación web. Esta plataforma ofrece una organización flexible y facilita el manejo de las tareas relacionadas con el avance del proyecto. Además, Jira cuenta con capacidades para el seguimiento del tiempo y la elaboración de informes de rendimiento, convirtiéndose en una herramienta fundamental para garantizar que el proyecto se ejecute de manera efectiva y ordenada.

Sistema de gestión de versiones

La plataforma seleccionada para la gestión de versiones en el desarrollo de la aplicación web es GitHub, la cual facilita el trabajo en equipo y el seguimiento de cada funcionalidad añadida durante el proceso de construcción. GitHub permite registrar todos los cambios realizados en el código, corregir errores y ofrece alojamiento en un repositorio remoto para la gestión remota de los cambios. De esta manera, se asegura una colaboración eficaz y un control detallado de las versiones del proyecto.

Análisis del diseño

Visión y Alcance

En esta sección se detalla la visualización del aplicativo web en desarrollo y se define su alcance.

Nombre del producto

Desarrollo de un sistema web de bienestar universitario para la gestión documental del área de medicina en una institución de educación superior.

Cliente objetivo

Personal administrativo del área de bienestar universitario de instituciones de educación superior.

Funcionalidades del sistema

Las funcionalidades del aplicativo web fueron gestionadas según los resultados obtenidos de la encuesta aplicada y validados por las personas involucradas. Para describir los requisitos del sistema, se elaboraron historias de usuarios.

Tabla 16. Historias de Usuario Autenticación

Historia de Usuario	
Número: 1	Usuario: Personal Administrativo y Médico
Nombre historia: Autenticación de Usuarios	
Programador responsable: Steveen Ordoñez	
Descripción:	
Como administrador o médico del sistema, quiero que los usuarios puedan acceder mediante credenciales seguras para garantizar la seguridad y confidencialidad de la información.	
Criterios de aceptación:	
<ul style="list-style-type: none">● Inicio de sesión con usuario y contraseña.● cambio de contraseña● recuperación de contraseñas● Validación de credenciales seguras.● Mensajes de error en caso de credenciales incorrectas.● Cierre de sesión seguro.	
Observaciones: Ninguna	

Tabla 17. Historias de usuarios de la Gestión de proyectos

Historia de Usuario	
Número: 2	Usuario: Personal administrativo
Nombre historia: Gestión de Proyectos	
Programador responsable: Steveen Ordoñez	
Descripción:	
Como administrador del sistema, quiero poder gestionar los proyectos para garantizar su disponibilidad en el sistema.	
Criterios de aceptación:	
<ul style="list-style-type: none">● Crear proyectos, editar y eliminar.● campos obligatorios: (nombre) opcionales: (descripción, foto).	
Observaciones: Ninguna	

Tabla 18. Historias de Gestión de periodos

Historia de Usuario	
Número: 3	Usuario: Personal administrativo
Nombre historia: Gestión de Periodos	
Programador responsable: Steveen Ordoñez	
Descripción:	
Como administrador del sistema, quiero poder gestionar los periodos para garantizar su disponibilidad en el sistema.	
Criterios de aceptación:	
<ul style="list-style-type: none">● Crear Periodos, editar y eliminar.● campos obligatorios: (semestre, año) opcionales: (descripción).	
Observaciones: Ninguna	

Tabla 19. Historias de usuarios de la Gestión de Actividades

Historia de Usuario	
Número: 4	Usuario: Personal Administrativo y Médico
Nombre historia: Gestión de Actividades	
Programador responsable: Steveen Ordoñez	
Descripción:	
Como administrador o médico del sistema, quiero poder gestionar las actividades para garantizar su disponibilidad en el sistema.	
Criterios de aceptación:	
<ul style="list-style-type: none">• Crear Periodos, actualizar su estado, descargar reportes por actividades, descargar documentos dentro de las actividades.	
Observaciones: Ninguna	
Tabla 20. Historias de usuarios de Reportes	
Historia de Usuario	
Número: 5	Usuario: Personal Administrativo y Médico
Nombre historia: Gestión Reportes	
Programador responsable: Steveen Ordoñez	
Descripción:	
Como administrador o médico del sistema, quiero poder descargar los reportes para garantizar su disponibilidad en el sistema.	
Criterios de aceptación:	
<ul style="list-style-type: none">• Descargar los reportes por actividades y por semestre y año.	
Observaciones: Ninguna	

Tabla 21. Historias de usuarios de la gestión del calendario

Historia de Usuario	
Número: 6	Usuario: Personal Administrativo y Médico
Nombre historia: Gestión Calendario	
Programador responsable: Steveen Ordoñez	
Descripción:	
Como administrador o médico del sistema, quiero poder visualizar las actividades realizadas para visualizar los estados y fechas en que se realizaron.	
Criterios de aceptación:	
<ul style="list-style-type: none">• Deben estar todas las actividades realizadas con el estado en que se encuentran además de señalar el estado bajo colores que demuestren el estado en el que se encuentra.	
Observaciones: Ninguna	

Tabla 22. Historias de usuarios de la gestión de documentos

Historia de Usuario	
Número: 7	Usuario: Personal Administrativo y Médico
Nombre historia: Gestión de Documentos	
Programador responsable: Steveen Ordoñez	
Descripción:	
Como administrador o médico del sistema, quiero registrar y descargar nuevos documentos en el sistema para garantizar su almacenamiento y disponibilidad.	
Criterios de aceptación:	
<ul style="list-style-type: none">• Subida de documentos en distintos formatos (.pdf, .docx, .jpg) y sus metadatos de (nombre, fecha, tipo de documento, tags, fecha de creación y actualización).	
Observaciones: Ninguna	

Tabla 23. Historias de usuarios búsqueda de archivos

Historia de Usuario	
Número: 8	Usuario: Personal Administrativo y Médico
Nombre historia: Búsqueda de Archivos Médicos	
Programador responsable: Erick García	
Descripción:	
Como personal médico y administrativo, quiero poder buscar documentos específicos utilizando filtros y palabras clave para acceder rápidamente a la información relevante.	
Criterios de aceptación:	
<ul style="list-style-type: none">• Filtros de búsqueda por nombre, fecha, tipo de documento, tags, periodo, año, nombre del proyecto.	

Observaciones: Ninguna

Tabla 24. Historias de usuarios búsqueda de archivos

Historia de Usuario	
Número: 9	Usuario: Personal Administrativo y Médico
Nombre historia: Descarga de Documentos	
Programador responsable: Erick García	
Descripción:	
Como personal médico y administrativo, quiero poder descargar documentos médicos almacenados en el sistema para su consulta o impresión cuando sea necesario.	
Criterios de aceptación:	
<ul style="list-style-type: none">• Descarga de documentos en su formato original.• Registro de cada documento en el sistema.	

Observaciones: Ninguna

Tabla 25. *Historias de usuarios interfaz*

Historia de Usuario

Número: 10

Usuario: Personal Administrativo y Médico

Nombre historia: Interfaz de usuarios

Programador responsable: Erick García

Descripción:

Como personal médico y administrativo, quiero que la interfaz del sistema sea clara y fácil de usar para poder realizar mis tareas sin dificultades ni necesidad de capacitación extensa.

Criterios de aceptación:

- Diseño limpio y organizado con accesos rápidos a funciones principales.
 - Etiquetas y explicaciones precisas en los botones y formularios.
-

Observaciones: Ninguna

Glosario de términos

El cuadro siguiente muestra algunas de las palabras fundamentales vinculadas a la carrera de software que se señalan en este trabajo de titulación. Actúa como guía para aquellos que no estén versados en el lenguaje técnico o que se topen con términos desconocidos. Así, se ofrece una respuesta para cubrir la curiosidad y potenciar la comprensión.

Tabla 26. Glosario de términos.

Término	Significado	Alias
Modelo Vista Controlador	El patrón MVC es un patrón de arquitectura que administra la separación de la lógica de negocio de la interfaz del usuario facilitando la mantenibilidad y escalabilidad de una aplicación. (Cahuana, 2022, pág. 39).	MVC
Front-end	El Front-end trabaja la interfaz visual, y hace que el usuario pueda interactuar con nuestro sitio o sistema. Está orientado al lenguaje de marcas y al lenguaje de programación web de ejecución en equipos clientes. (Pérez, Quispe, Mullicundo & Lamas, 2021, pág. 01).	Front-end
Back-end	El Backend se encarga de la manipulación de los datos, un Back-end no sirve de mucho si no existe un Front-end de por medio, el desarrollador Back-end debe de conocer de bases de datos, frameworks y aspectos de seguridad. (Pérez, Quispe, Mullicundo & Lamas, 2021, pág. 01).	Back-end
Framework	Framework es una colección de comandos o funciones básicas que forman ciertas reglas e interactúan entre sí, de modo que al crear una aplicación web, debemos seguir las reglas del Framework. (Adriano, 2022, pág. 18).	Framework
React	React es una librería open-source de Javascript la cual sirve para la construcción de interfaces, está siendo apoyada por Facebook. (Murgueytio, Galarza, P. J., & Barrientos, 2022, Pág. 2)	React
Cascada	Este modelo toma las actividades fundamentales del proceso de especificación, desarrollo, validación y evolución y las representa como fases separadas del proceso. (Olivera, de la C, & Díaz Alonso, 2021)	Cascada
AWS Amplify	AWS Amplify es un conjunto de herramientas y servicios que permite a los desarrolladores crear y escalar aplicaciones móviles sin tener que preocuparse por la infraestructura subyacente. (Martín, Chavez, Flores, Zogbe, Rodríguez, & Murazzo, 2023, Pág. 3)	AWS Amplify

Tabla 27. Tabla de Requerimientos funcionales

ID	Requerimiento	Descripción
RF1	Autentificación de usuario	Permitir el acceso solo a usuarios registrados mediante credenciales seguras.
RF2	Registrar documentos	Permitir a los usuarios subir y registrar nuevos documentos en el sistema.
RF3	Modificar documentos	Habilitar la edición de documentos previamente registrados según los permisos del usuario.
RF4	Eliminar documentos	Permitir la eliminación de documentos con las autorizaciones correspondientes.
RF5	Búsqueda de archivos	Proveer una funcionalidad para localizar documentos mediante filtros y palabras clave.
RF6	Descargar archivos	Habilitar la descarga de documentos según los permisos del usuario.
RF7	Facilidad de uso	Crear una interfaz que sea fácil de usar y accesible para optimizar la vivencia del usuario.

Tabla 28. Requerimientos no funcionales

ID	Requerimiento	Descripción
RF1	Seguridad de documentos	Garantizar la protección de los documentos contra accesos no autorizados.
RF2	Calidad de Contraseñas	Establecer normas de protección para generar contraseñas seguras.
RF3	Controles de acceso	Configurar funciones y permisos para limitar el acceso a propiedades y archivos.
RF4	Cantidad de usuarios concurrentes	Permite que varios usuarios accedan al mismo tiempo sin comprometer el rendimiento.

Modelado UML

Esta sección presenta diversos diagramas UML que ilustran de manera visual la estructura y el comportamiento del aplicativo web, facilitando su comprensión y análisis.

Diagrama de Clases

Este diagrama resulta sumamente beneficioso, pues facilita la definición precisa de la estructura de un sistema, representando sus clases, atributos, operaciones y relaciones de forma exacta. Para la creación del esquema, se emplea la herramienta AWS Amplify, que comprende las clases siguientes:

- **Documents:** esta clase hace referencia a los documentos que se gestionan dentro del sistema, incluyendo atributos como el nombre, la ruta de almacenamiento, etiquetas y tipo de documento. Además, mantiene una relación con las actividades, permitiendo asociar cada documento a una actividad específica dentro del sistema.
- **Period:** esta clase hace referencia a los períodos académicos durante los cuales se realizarán los proyectos en el sistema. Incorpora elementos como el año, el semestre y una descripción del intervalo. Igualmente, establece una relación con los proyectos, simplificando la organización y gestión de estos conforme al período académico correspondiente.
- **Project:** esta clase hace referencia a los proyectos gestionados dentro del sistema. Incorpora elementos como el título, el resumen y la localización del archivo, lo que simplifica la organización y el acceso eficiente a los datos relevantes de cada proyecto.
- **ActivityTasks:** esta clase hace referencia a las obligaciones asociadas a las acciones realizadas dentro del sistema. Incorpora elementos como el título y la descripción de cada obligación. Asimismo, establece una relación con las acciones, simplificando la organización y gestión de las obligaciones dentro de una acción específica.

- **Activity:** esta clase se relaciona con las acciones administradas en el sistema. Incorpora aspectos fundamentales como la situación actual de la acción, la fecha en que se realiza, los periodos de inicio y finalización, la entidad que la lleva a cabo, el responsable del proyecto, el cargo ocupado, la unidad pertinente, el objetivo global, el número de participantes y el gasto destinado. Además, crea conexiones con los proyectos, períodos, tareas vinculadas y documentos, lo que permite una gestión organizada y metódica de cada acción en el sistema.

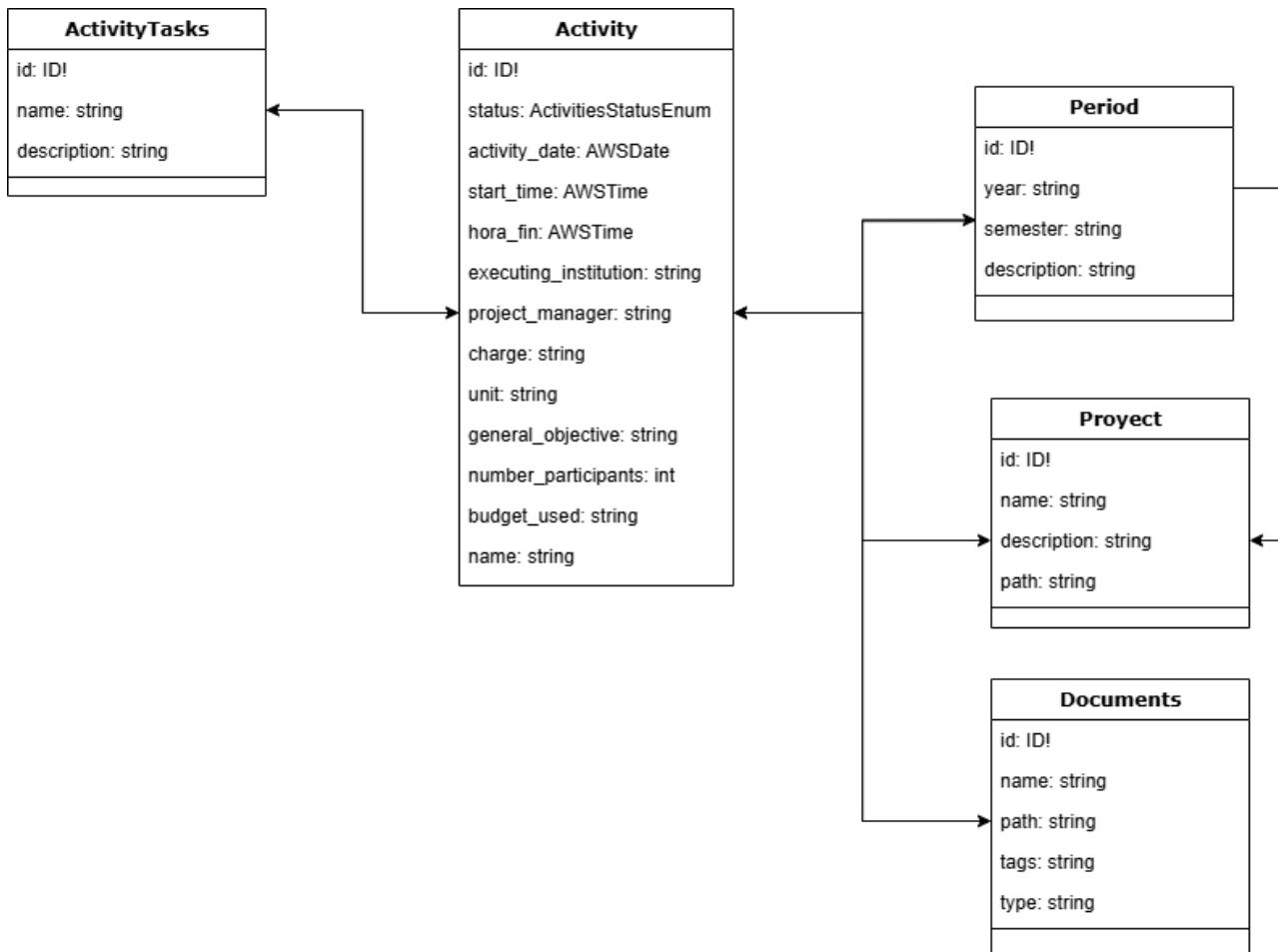


Figura 28. Diagrama de clases.

Diagrama de caso de uso

El siguiente diagrama muestra la interacción de los actores con las funcionalidades del sistema. Estos actores se dividen en dos grupos: Administrador y Personal Médico. El Administrador tiene acceso completo a todas las funcionalidades del sistema, mientras que el Personal Médico sólo podrá acceder a ciertas funciones específicas.

Se utilizó un diagrama de caso de uso porque permite representar visualmente las interacciones entre los actores y el sistema, diferenciando sus niveles de acceso. A diferencia de las historias de usuario, que describen necesidades individuales, los casos de uso modelan escenarios complejos y detallan flujos alternativos, excepciones y precondiciones, facilitando el análisis y la comunicación del sistema.

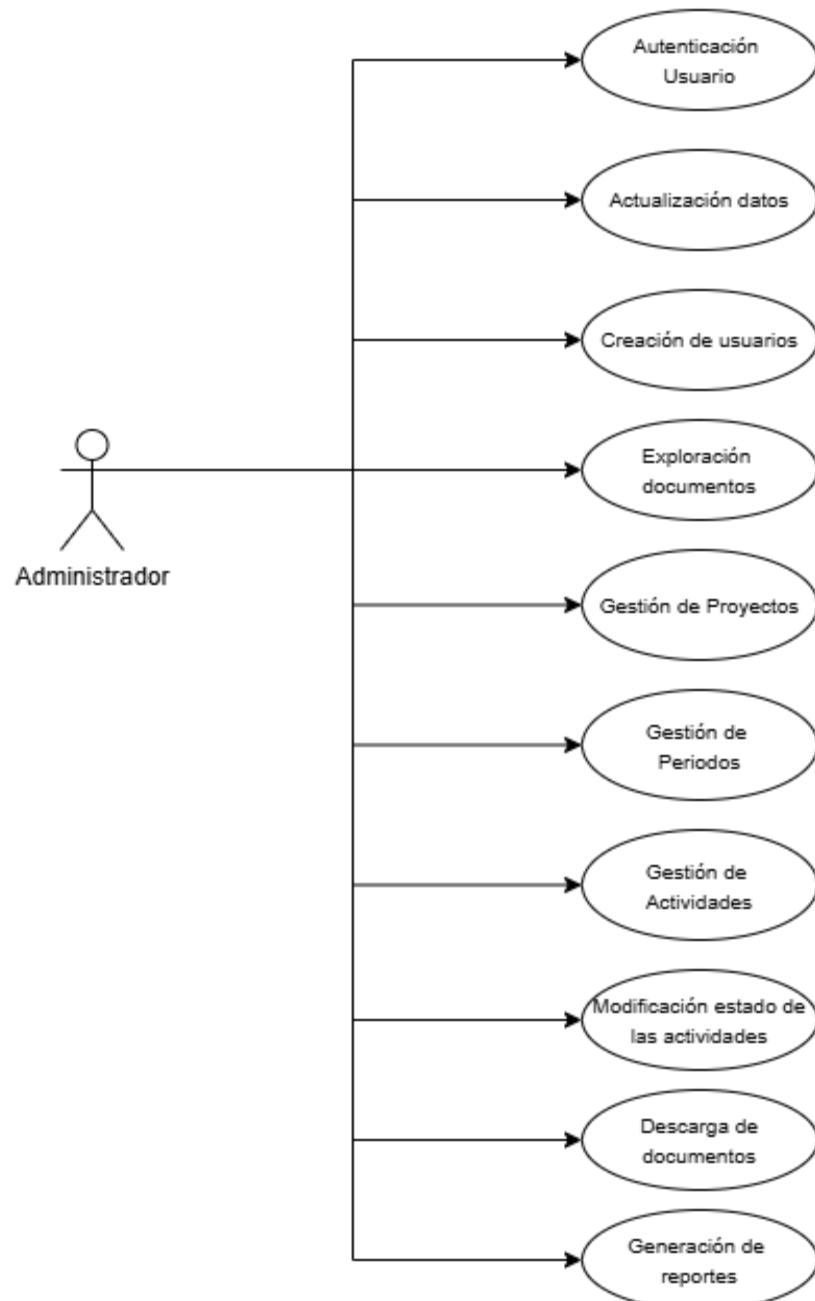


Figura 29. Diagrama de caso de uso (Administrador).

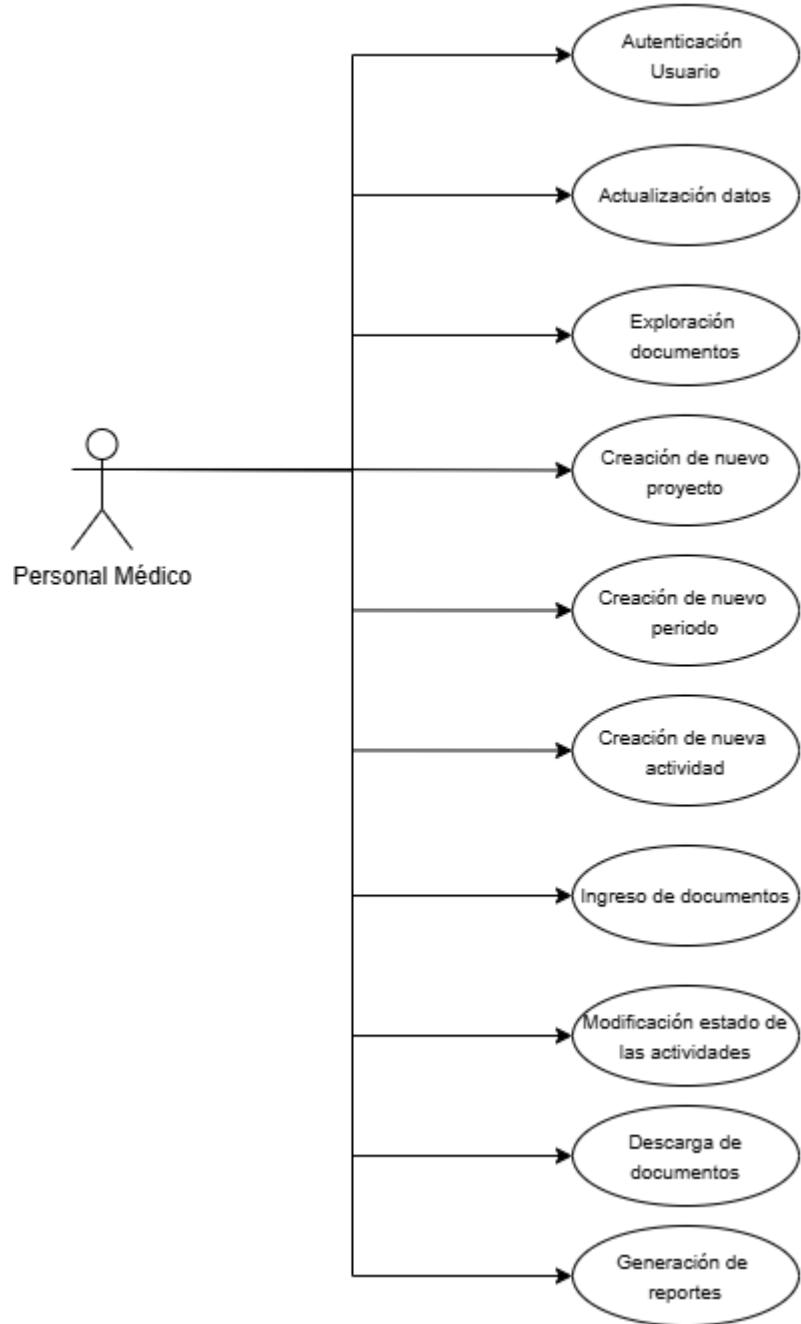


Figura 30. Diagrama de caso de uso (*Personal Médico*).

Tabla 29. Tabla de casos de uso

Nº	Nombre de Caso de Uso
1	Autenticación de Usuarios
2	Actualización de datos
3	Creación de usuarios
4	Exploración de usuarios
5	Gestión de Proyectos
	5.1 Creación de nuevo proyecto
	5.2 Eliminar proyecto
	5.3 Editar proyecto
6	Gestión de Periodos
	6.1 Creación de nuevo periodo
	6.2 Eliminar periodo
	6.3 Editar periodo
7	Gestión de Actividades
	7.1 Creación de nueva actividad
	7.2 Eliminar actividad
	7.3 Editar actividad
8	Ingreso de documentos
9	Modificar estado de las actividades
	9.1 Cambiar estado de la actividad (Pendiente, Completada, Cancelada, En Progreso)

10 Descarga de documentos

11 Generación de reportes

11.1 Generación de reporte semestral

11.2 Generación de reporte anual

Diagrama de actividades

El siguiente diagrama muestra las actividades disponibles para el usuario según su rol dentro de la aplicación. Este modelo permite representar de manera clara el flujo de acciones que se llevan a cabo dentro del sistema.

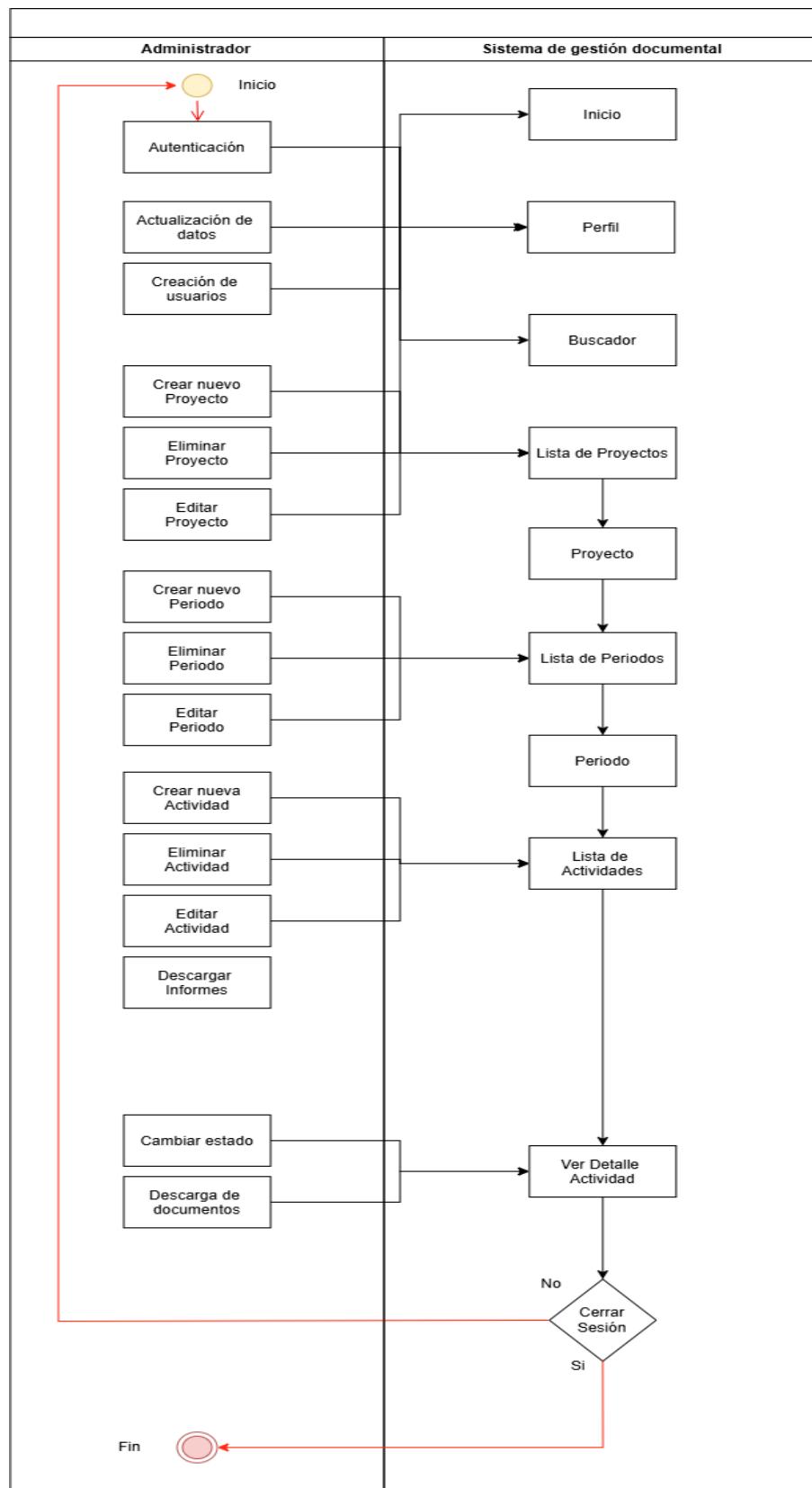


Figura 31. Diagrama de actividades Administrador.

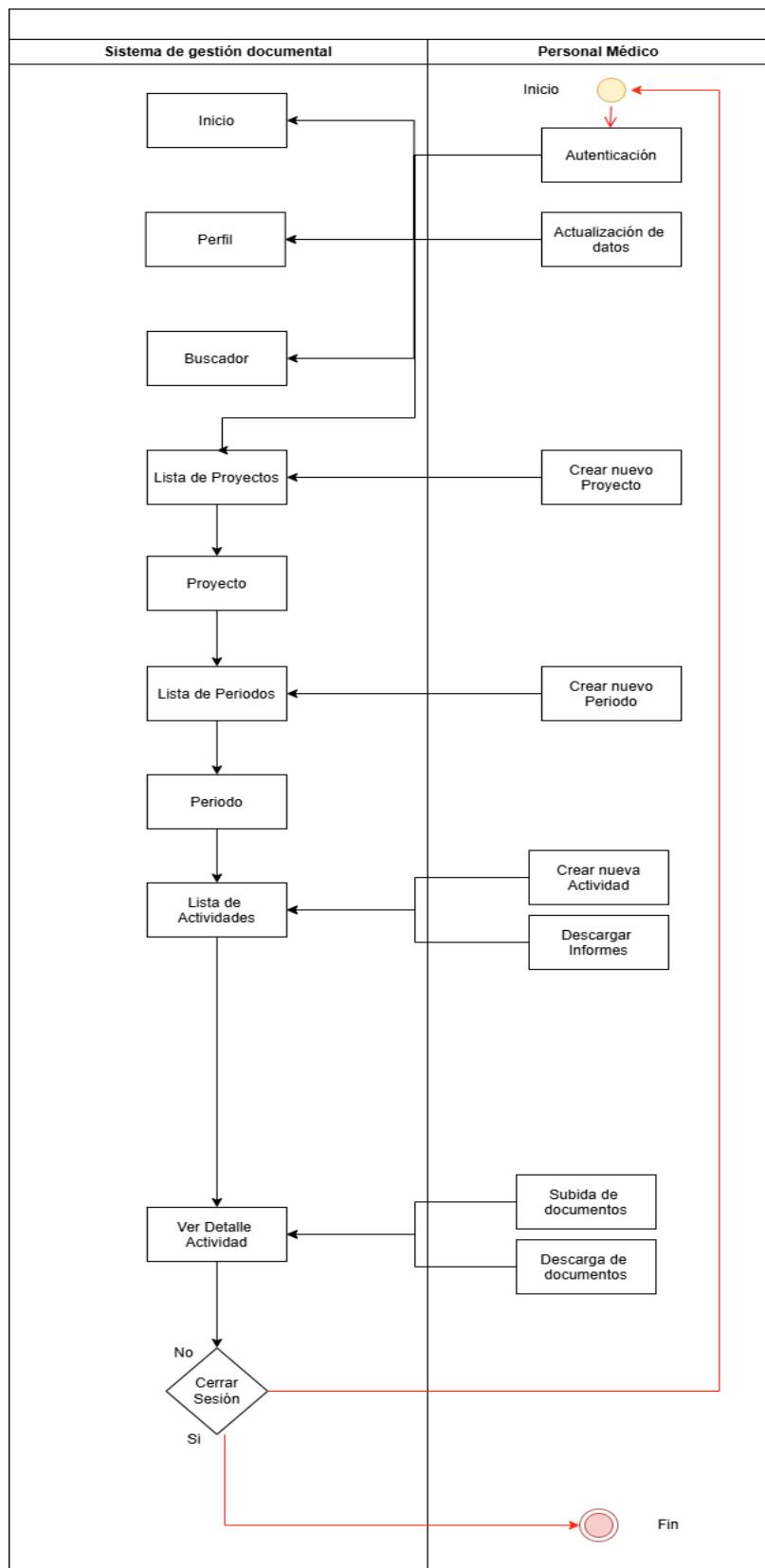


Figura 32. Diagrama de actividades personal médico.

Diagrama de estados

Este diagrama permite definir el procesamiento de datos y la interacción del usuario con las entidades externas que forman parte del sistema, proporcionando una visión clara de su funcionamiento.

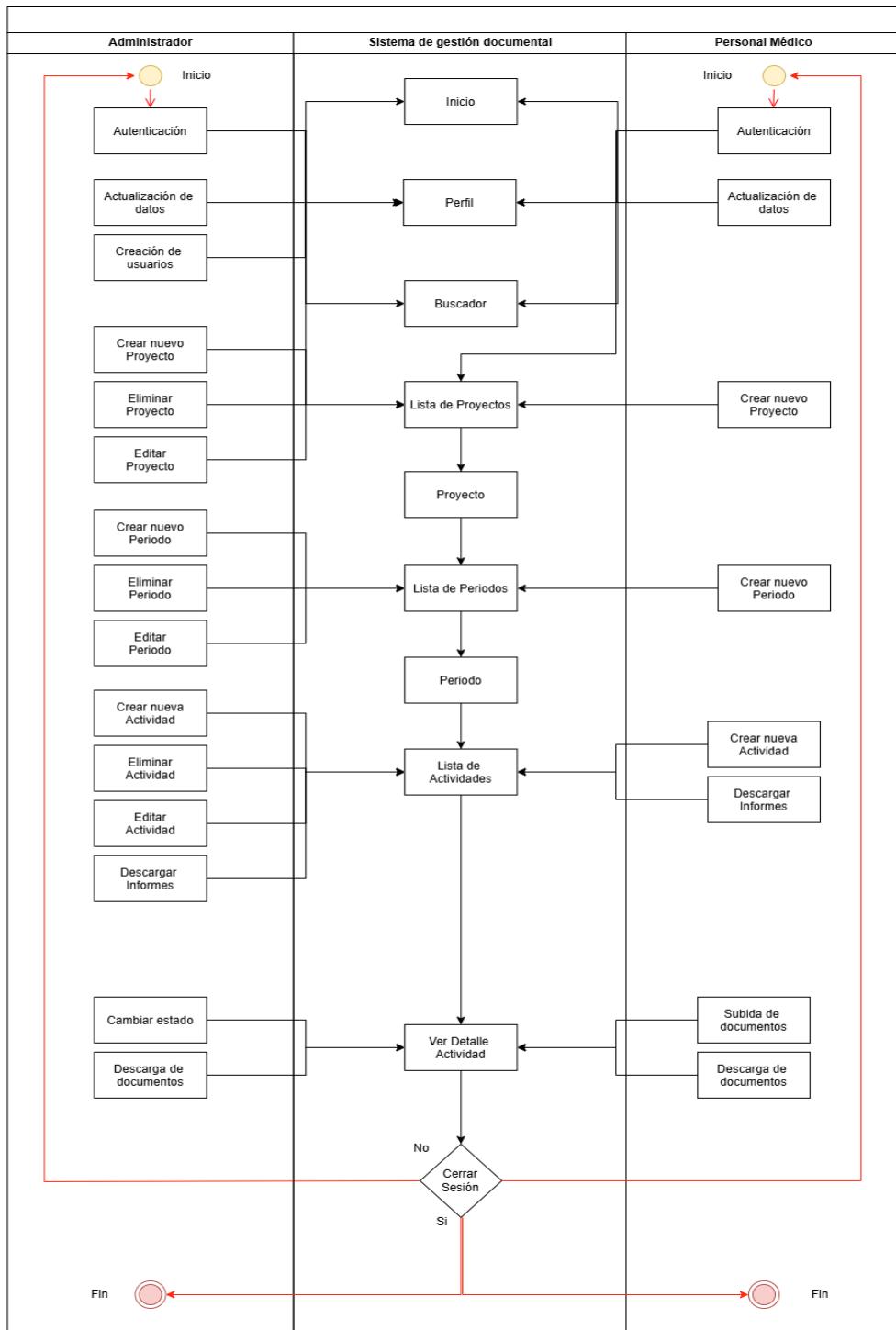


Figura 33. Diagrama de estados.

Base de datos Amazon DynamoDB

Para guardar los datos del proyecto, se optó por Amazon DynamoDB, un servicio de base de datos NoSQL que es totalmente administrado y proporciona altos niveles de rendimiento y capacidad de escalamiento automático. DynamoDB facilita la organización de los datos en tablas que cuentan con claves primarias ajustables, mejorando así la búsqueda y obtención de información. Su conexión con otros servicios de AWS asegura una disponibilidad excelente, protección y facilidad en la gestión de datos, ofreciendo una experiencia eficaz y fiable en el almacenamiento y manejo de la información.

The screenshot shows the AWS DynamoDB schema configuration interface with four tables defined:

- Documents**:
 - Fields: id (ID!), name (String), path (String), tags (String), type (String).
 - Relationships: name → Activities (Activity, m:n man...).
- Period**:
 - Fields: id (ID!), year (String), semester (String), description (String).
 - Relationships: name → Project (Project, 1:1 one P...).
- Project**:
 - Fields: id (ID!), name (String), description (String), path (String).
 - Relationships: None.
- ActivitiesStatusEnum**:
 - Values: EARRING, COMPLETED, CANCELADA, IN_PROGRESS.
 - Relationships: None.

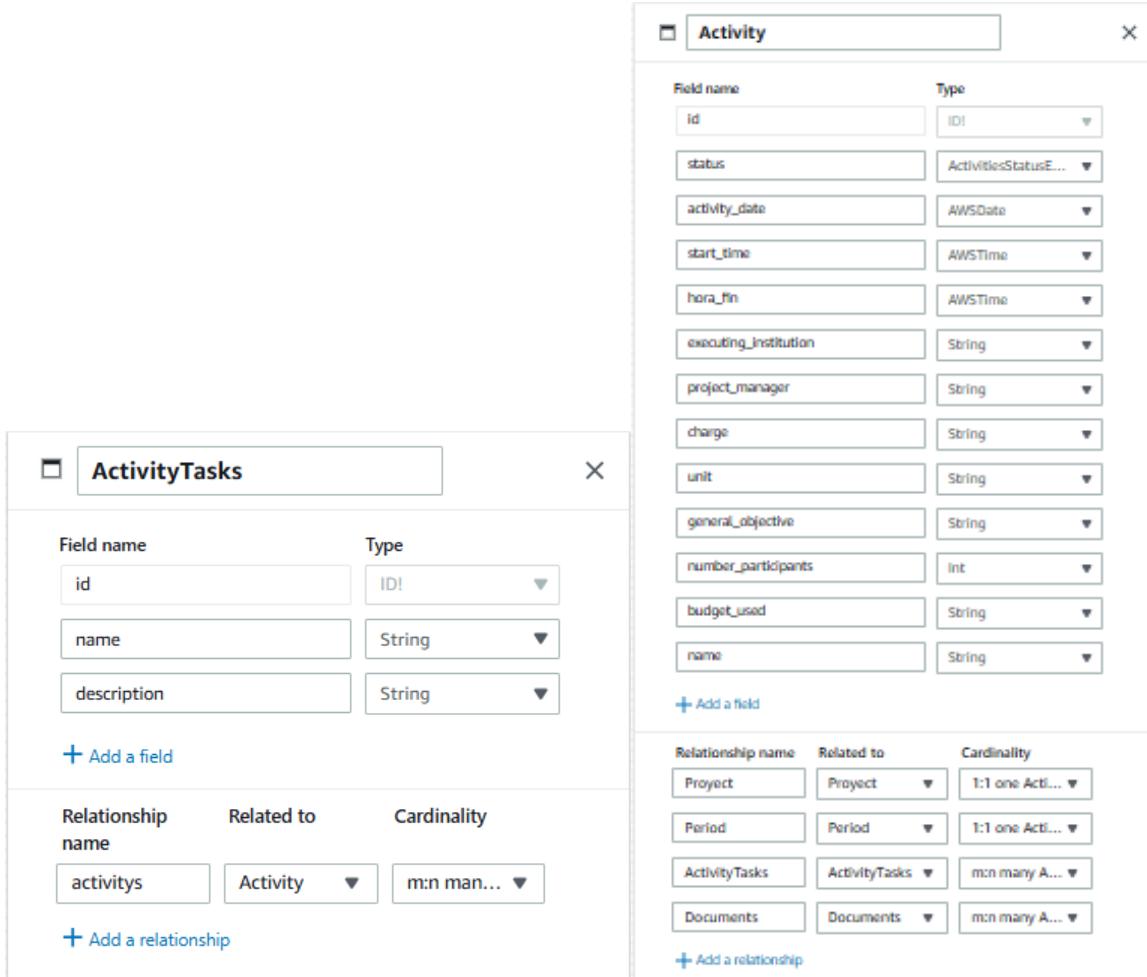


Figura 34. Base de datos Amazon DynamoDB.

La Figura 34 muestra el esquema de base de datos utilizado para la elaboración de la aplicación web. Este diagrama permite visualizar las distintas entidades y sus vínculos, lo que promueve una gestión eficaz de los datos y una ejecución correcta de las consultas.

Diseño de interfaz de usuario

En esta sección se muestra el flujo de pantallas del sistema web de bienestar universitario para la administración de documentos del campo de medicina. El medio por el cual los usuarios se relacionan con el sistema es la interfaz de usuario. Se empleó la herramienta Draw.io para su representación. El sistema incluye dos funciones clave: el personal de administración y el personal médico. A continuación, se describen los flujos correspondientes a cada uno.

Personal Administrativo

El personal administrativo se conecta al sistema introduciendo su información de acceso, compuesta por su dirección de correo y su clave. Si llega a olvidar su clave, tiene la opción de ir a la sección de recuperación, donde deberá ingresar su dirección de correo para recibir un enlace para restablecerla.

Tras lograr el inicio de sesión, el usuario accede al panel de tareas, donde se presenta un menú que proporciona diversas funciones. Entre estas se encuentra la alternativa de búsqueda de documentos, que asiste en la localización de archivos mediante filtros concretos, facilitando un acceso rápido a la información.

Además, el sistema cuenta con una interfaz diseñada para los usuarios, en la que pueden realizar una actualización eficaz de sus datos personales. Esta parte también les facilita la creación de nuevos perfiles, facilitando la gestión y gestión de cuentas en la plataforma.

La pantalla principal cuenta con herramientas sofisticadas que facilitan a los usuarios una visión nítida y exhaustiva de las estadísticas de los documentos introducidos. A través de gráficos interactivos y datos pertinentes, la plataforma ofrece un análisis activo, simplificando la interpretación de los datos y el proceso de decisión en el sistema.

Además, el sistema proporciona una lista de proyectos, desde la cual se pueden observar los distintos proyectos en ejecución. En cada proyecto, el usuario puede acceder a la lista de períodos, dispuestos en orden cronológico para un seguimiento más efectivo de la información.

En cada periodo de tiempo, existen una serie de actividades que facilitan la supervisión y la observación de las actividades. El usuario tiene la posibilidad de seleccionar una tarea específica para obtener información exhaustiva, en la que se muestran explicaciones, archivos asociados y el estado pertinente.

Para facilitar la toma de decisiones y la gestión de datos, el sistema ofrece una vista previa del informe, brindando al usuario la posibilidad de ver un resumen ordenado de la información antes de elaborar el documento final. Esta característica optimiza la valoración y asegura la precisión de los datos proporcionados.

El sistema incluye una pantalla de calendario, donde los usuarios pueden visualizar eventos, actividades y vencimientos de documentos organizados cronológicamente. Esta función permite una mejor planificación y gestión del tiempo, asegurando que las tareas

sean atendidas de manera eficiente. El calendario se integra con el resto del sistema, proporcionando notificaciones y recordatorios automáticos.

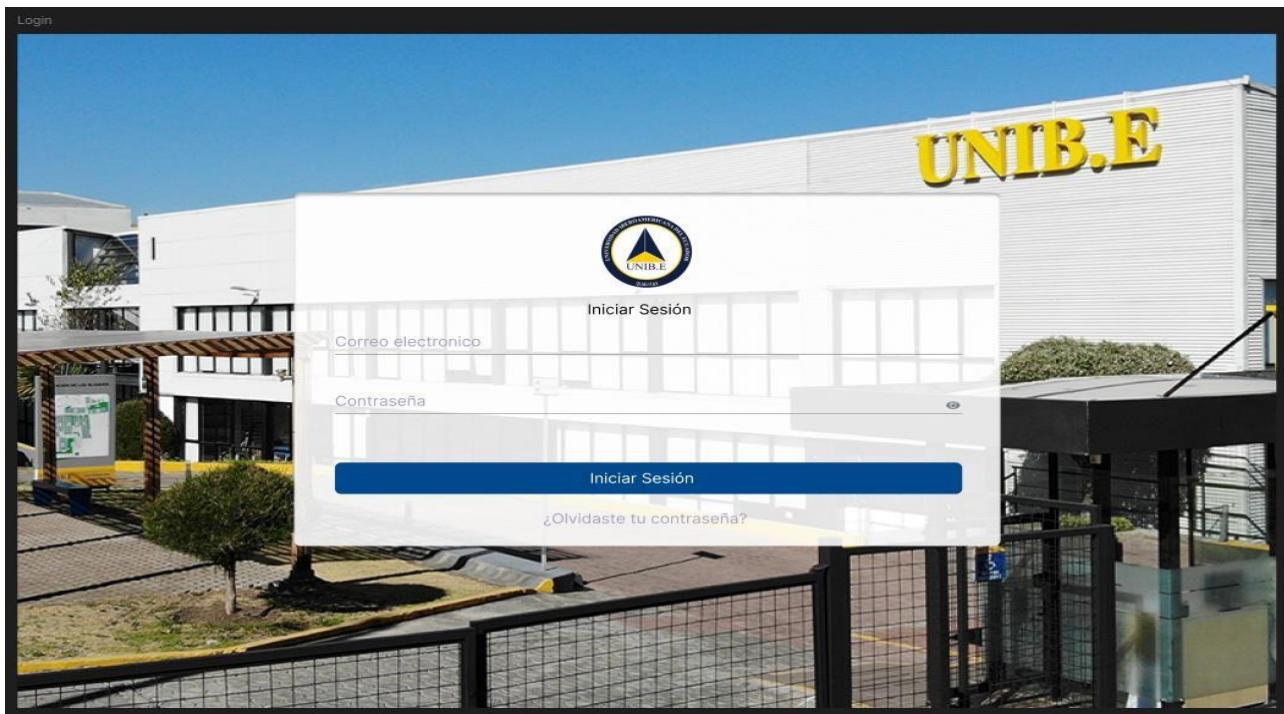


Figura 35. Diseño de interfaz de usuario (Personal Administrativo Login).

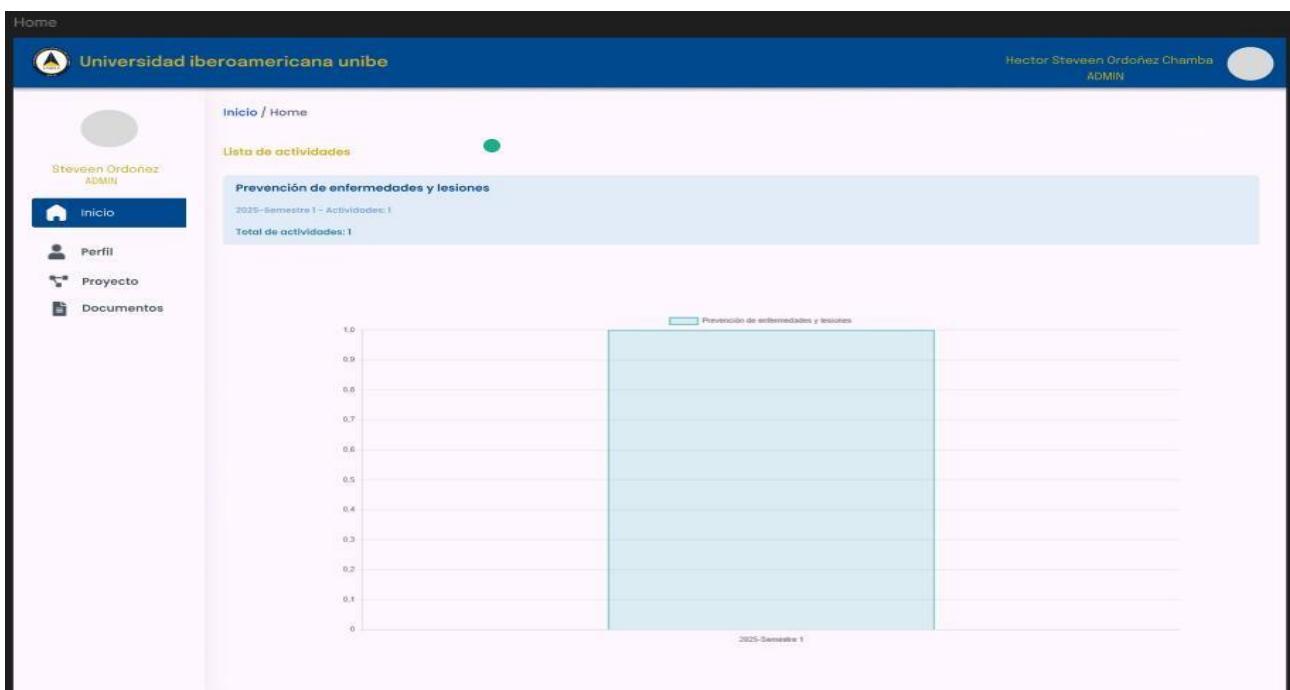


Figura 36. Diseño de interfaz de usuario (Personal Administrativo Inicio).

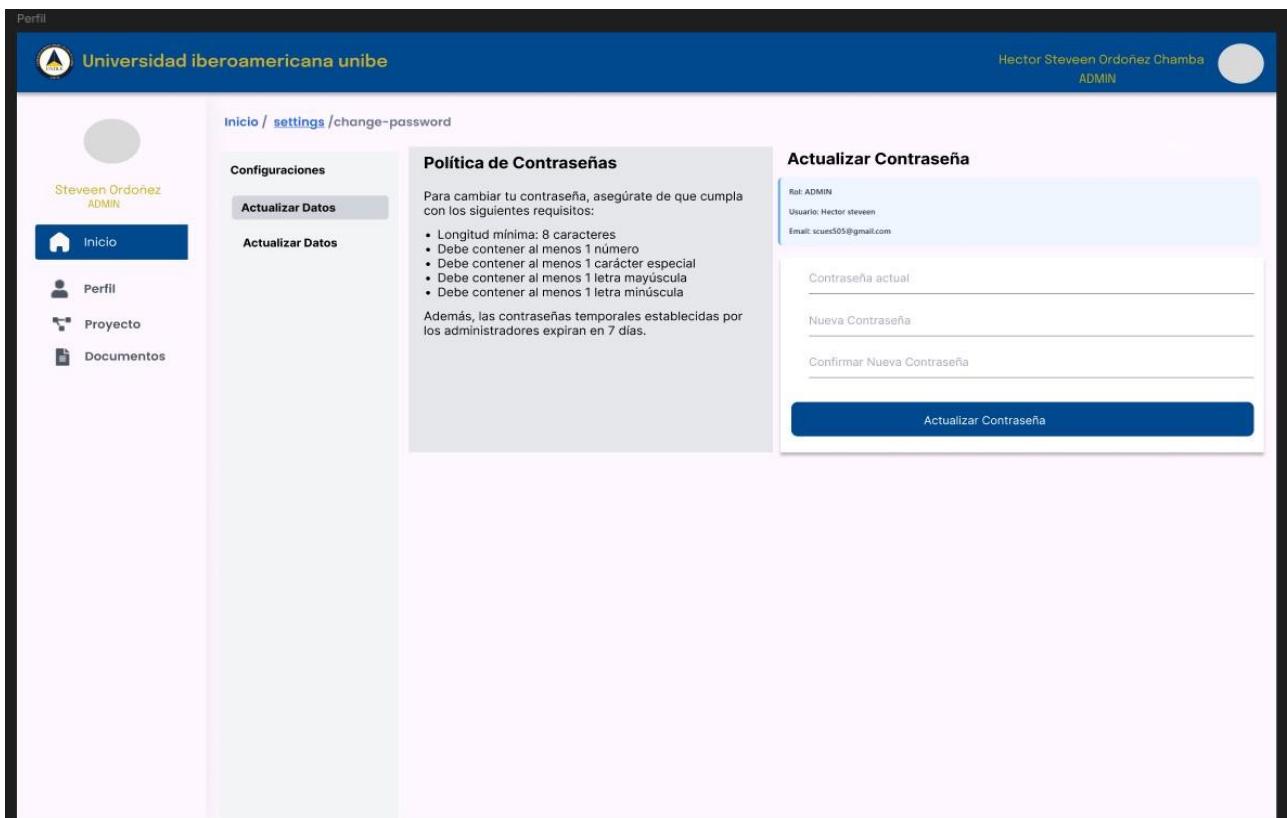


Figura 37. Diseño de interfaz de usuario (Personal Administrativo Perfil).

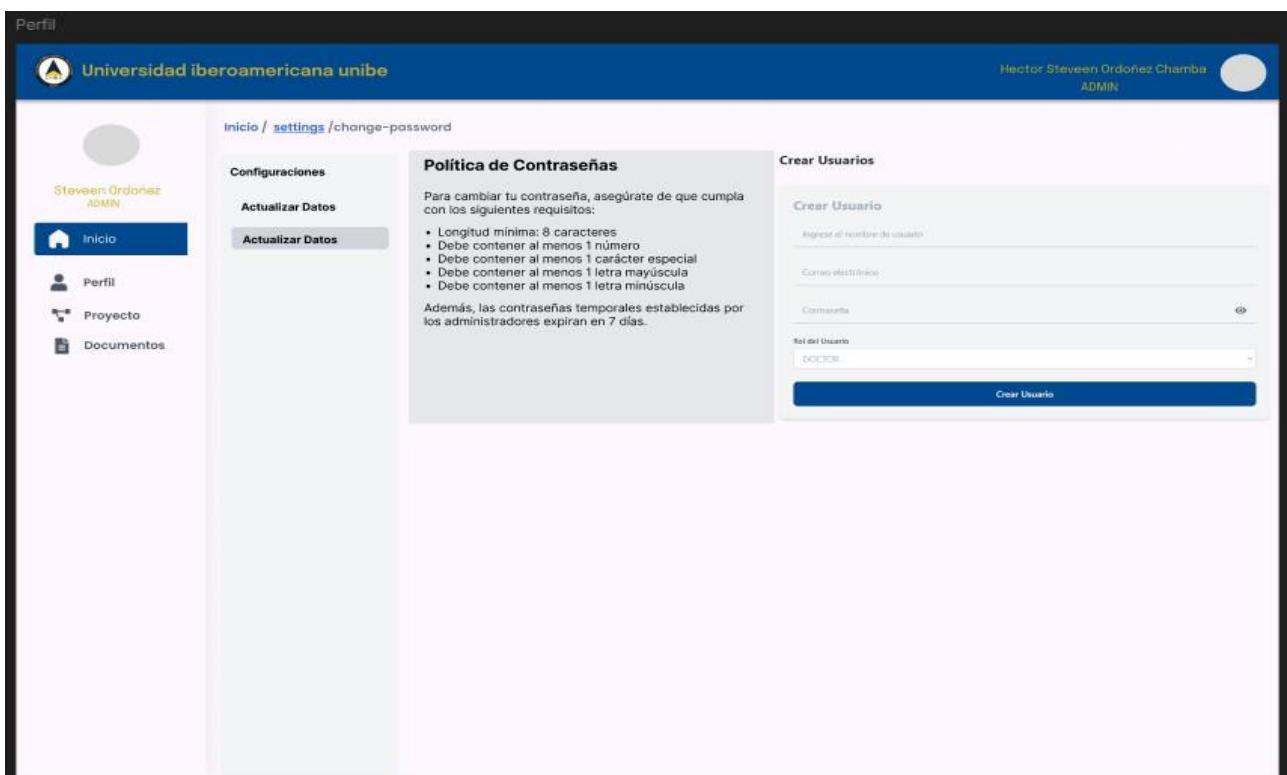


Figura 38. Diseño de interfaz de usuario (Personal Administrativo Perfil Crear Usuarios).

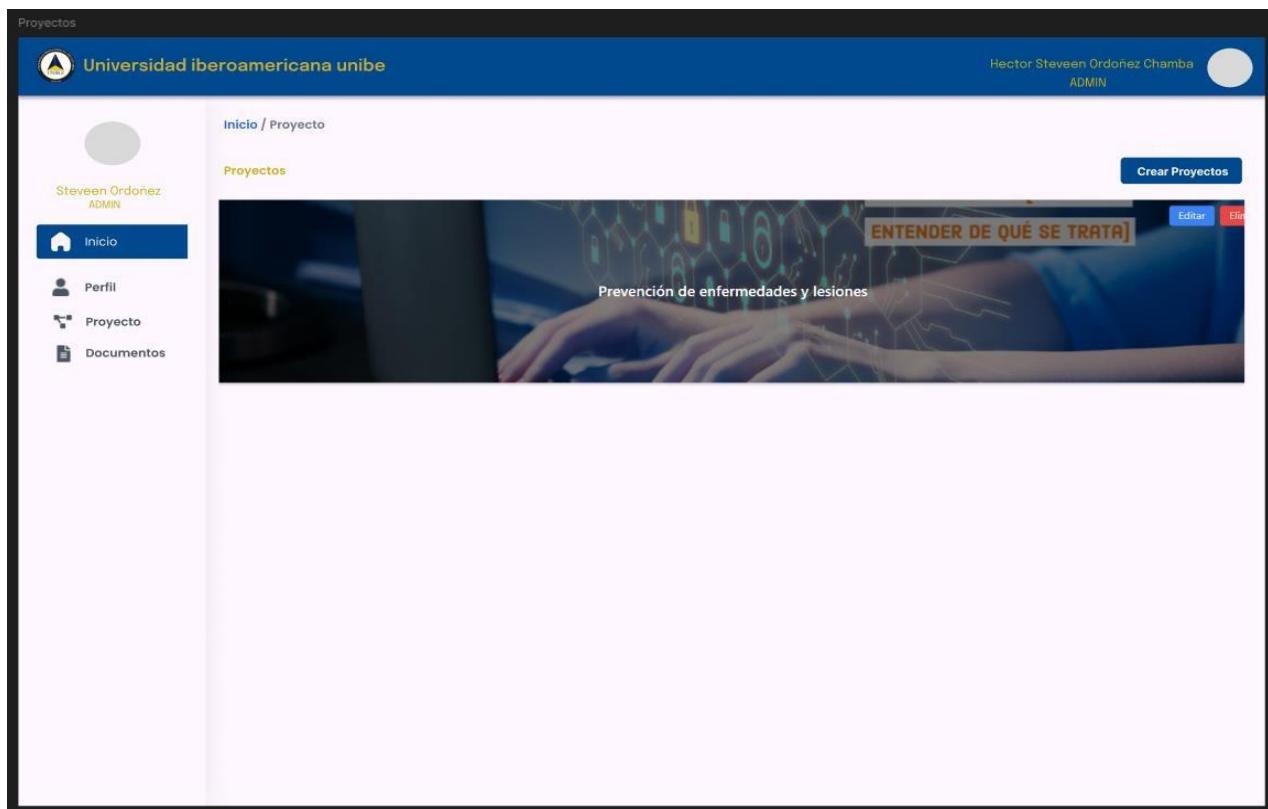


Figura 39. Diseño de interfaz de usuario (Personal Administrativo Proyectos).

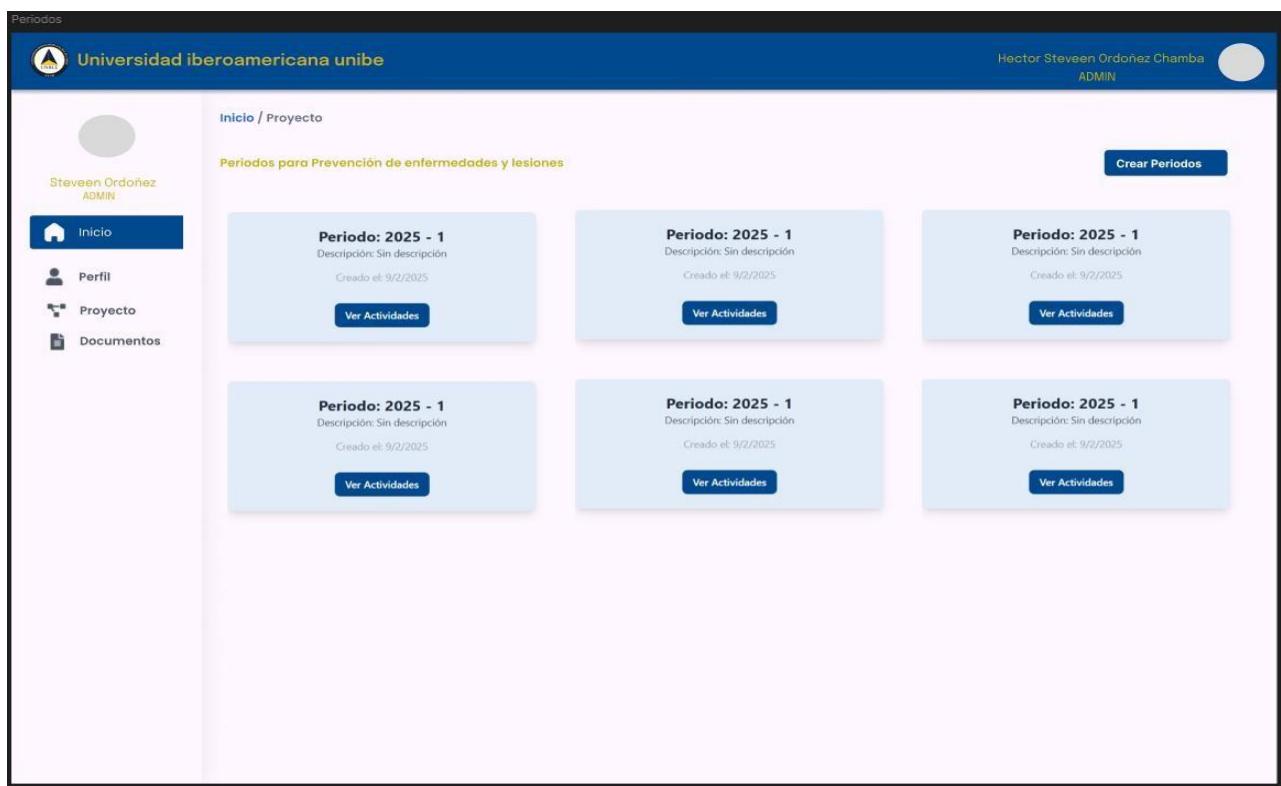


Figura 40. Diseño de interfaz de usuario (Personal Administrativo Periodos).

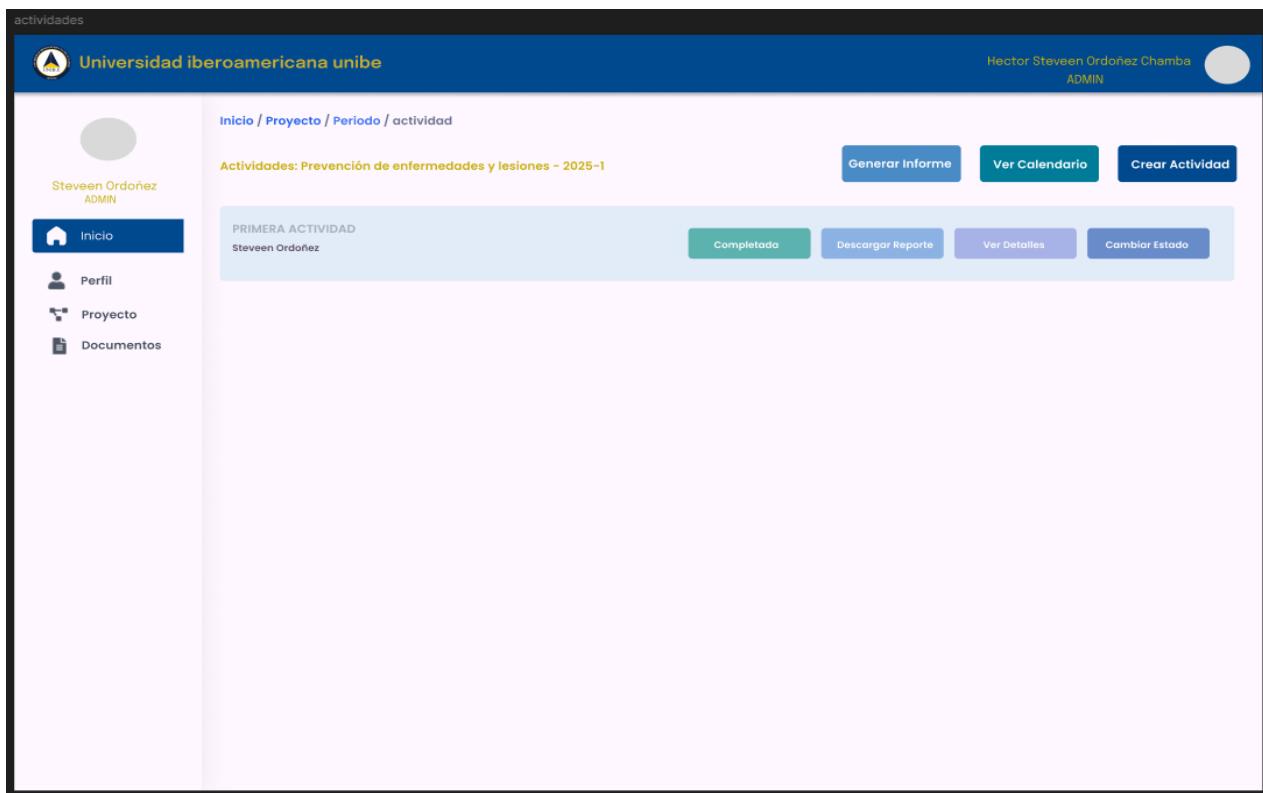


Figura 41. Diseño de interfaz de usuario (Personal Administrativo Actividades).

Figura 42. Diseño de interfaz de usuario (Personal Administrativo Crear Actividades).

Detalles de la actividad

Universidad iberoamericana unibe

Hector Steveen Ordoñez Chamba
Doctor

Steveen Ordoñez
Doctor

Inicio

Perfil

Proyecto

Documentos

Detalles de la actividad

Inicio / Proyecto / Periodo / actividad

Detalles de la actividad

	BENESTAR UNIVERSITARIO	CÓDIGO: PÁGINA: 1 DE 2
FECHA DE ACTUALIZACIÓN: 2025-02-09	INFORME DE ACTIVIDADES REALIZADAS	VERSIÓN: 001

1. Datos Generales

Nombre del Proyecto: Prevención de enfermedades y lesiones
Institución Ejecutora: unibe
Responsable del Proyecto: Steveen ordoñez
Categoría: Bienestar
Unidad: unicef
Periodo del informe: 2025 - 1

2. Objetivo General

objetivo

3. Actividades Realizadas

Nombre: SEGUNDA
Descripción: asas

Nombre: primera
Descripción: asas

4. Número de Participantes

344

5. Presupuesto utilizado

3434

6. Anexos

Presupuesto
Este archivo no es una imagen o está en formato no soportado.
[Descargar Presupuesto](#)

Artes gráficos de difusión
Este archivo no es una imagen o está en formato no soportado.
[Descargar](#)

Figura 43. Diseño de interfaz de usuario (Personal Administrativo Visualizar Actividades).

DOCUMENTOS

Universidad iberoamericana unibe

Hector Steveen Ordoñez Chamba
ADMIN

Steveen Ordoñez
ADMIN

Inicio

Perfil

Proyecto

Documentos

Inicio / Proyecto / Periodo / actividad / crear-actividad

Lista de Documentos

2025 ▾ Semester 1 ▾ Todos los tipos ▾ Todos los proyectos ▾ Buscar por nombre o tags:

Proyecto: Prevención de enfermedades y lesiones periodo: 2025-1 Nombre del archivo: Actividad 1 software.docx Tags: docx Creado: 02/02/2025, 10:14:01 Última actualización: 02/02/2025, 10:14:01 Descargar	Proyecto: Prevención de enfermedades y lesiones periodo: 2025-1 Nombre del archivo: comparative exercises 5.pdf Tipo Archivo: PDF para imprimir Tags: pdf Creado: 02/02/2025, 10:14:01 Última actualización: 02/02/2025, 10:14:01 Descargar	Proyecto: Prevención de enfermedades y lesiones periodo: 2025-1 Nombre del archivo: actividades.pdf Tipo Archivo: PDF para imprimir Tags: pdf Creado: 02/02/2025, 10:14:01 Última actualización: 02/02/2025, 10:14:01 Descargar
Proyecto: Prevención de enfermedades y lesiones periodo: 2025-1 Nombre del archivo: comparative exercises 5.docx Tipo Archivo: Microsoft Word Tags: docx Creado: 02/02/2025, 10:14:01 Última actualización: 02/02/2025, 10:14:01 Descargar	Proyecto: Prevención de enfermedades y lesiones periodo: 2025-1 Nombre del archivo: apli.png Tipo Archivo: Imagen Tags: png Creado: 02/02/2025, 10:14:01 Última actualización: 02/02/2025, 10:14:01 Descargar	Proyecto: Prevención de enfermedades y lesiones periodo: 2025-1 Nombre del archivo: gct1.png Tipo Archivo: Imagen Tags: png Creado: 02/02/2025, 10:14:01 Última actualización: 02/02/2025, 10:14:01 Descargar

Figura 44. Diseño de interfaz de usuario (Personal Administrativo Documentos).

Personal Médico

El personal médico también inicia sesión en el sistema ingresando sus credenciales, conformadas por su correo electrónico y contraseña. En caso de olvidar la contraseña, puede acceder a la pantalla de recuperación de contraseña, donde deberá ingresar su correo electrónico para recibir un enlace de restablecimiento.

Una vez verificada la autenticación, el usuario ingresa al panel de tareas, donde se presenta un menú con múltiples funciones. El buscador de documentos es una de estas herramientas, que posibilita la búsqueda de archivos a través de filtros concretos, simplificando el acceso inmediato a la información.

Asimismo, el sistema cuenta con una pantalla especialmente diseñada para los usuarios, donde pueden actualizar su información personal de manera sencilla y eficiente.

La pantalla de inicio está diseñada con una funcionalidad avanzada que permite a los usuarios visualizar de manera clara y detallada las estadísticas relacionadas con los documentos ingresados. A través de gráficos interactivos y métricas relevantes, la plataforma ofrece un análisis dinámico de la información, facilitando la interpretación de los datos y mejorando la toma de decisiones dentro del sistema.

El sistema también ofrece una lista de proyectos, desde donde se pueden visualizar los diferentes proyectos en curso. Dentro de cada proyecto, el usuario tiene acceso a la lista de períodos, organizados cronológicamente para un mejor seguimiento de la información.

En cada intervalo de tiempo, hay un conjunto de tareas que ayuda a supervisar y observar las actividades. El usuario tiene la opción de elegir una tarea particular para ver información detallada, donde se presentan explicaciones, archivos relacionados y el estado correspondiente.

Con el fin de simplificar la toma de decisiones y la administración de datos, el sistema proporciona una vista previa del informe, permitiendo al usuario examinar un compendio ordenado de la información antes de crear el documento definitivo. Esta característica mejora la evaluación y garantiza la exactitud de los datos presentados.

El sistema incluye una pantalla de calendario, donde los usuarios pueden visualizar eventos, actividades y vencimientos de documentos organizados cronológicamente. Esta función permite una mejor planificación y gestión del tiempo, asegurando que las tareas

sean atendidas de manera eficiente. El calendario se integra con el resto del sistema, proporcionando notificaciones y recordatorios automáticos.

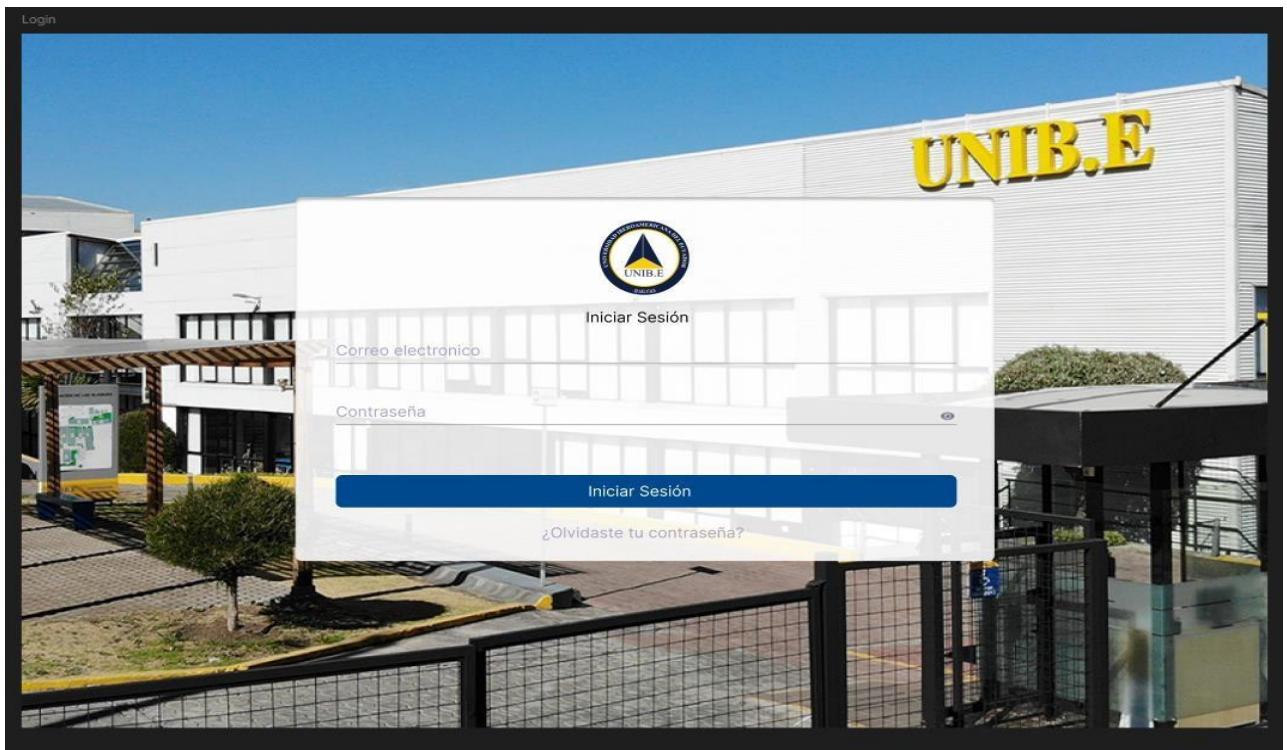


Figura 45. Diseño de interfaz de usuario (Personal Médico Login).

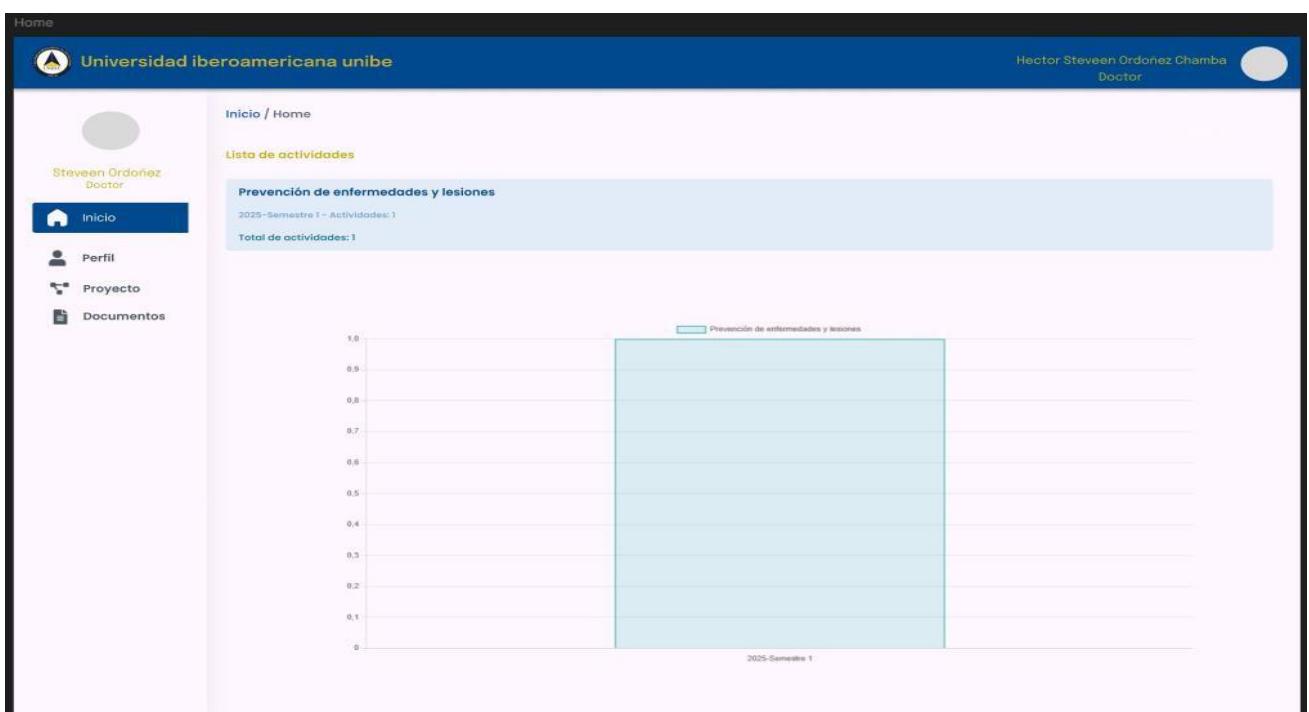


Figura 46. Diseño de interfaz de usuario (Personal Médico Inicio).

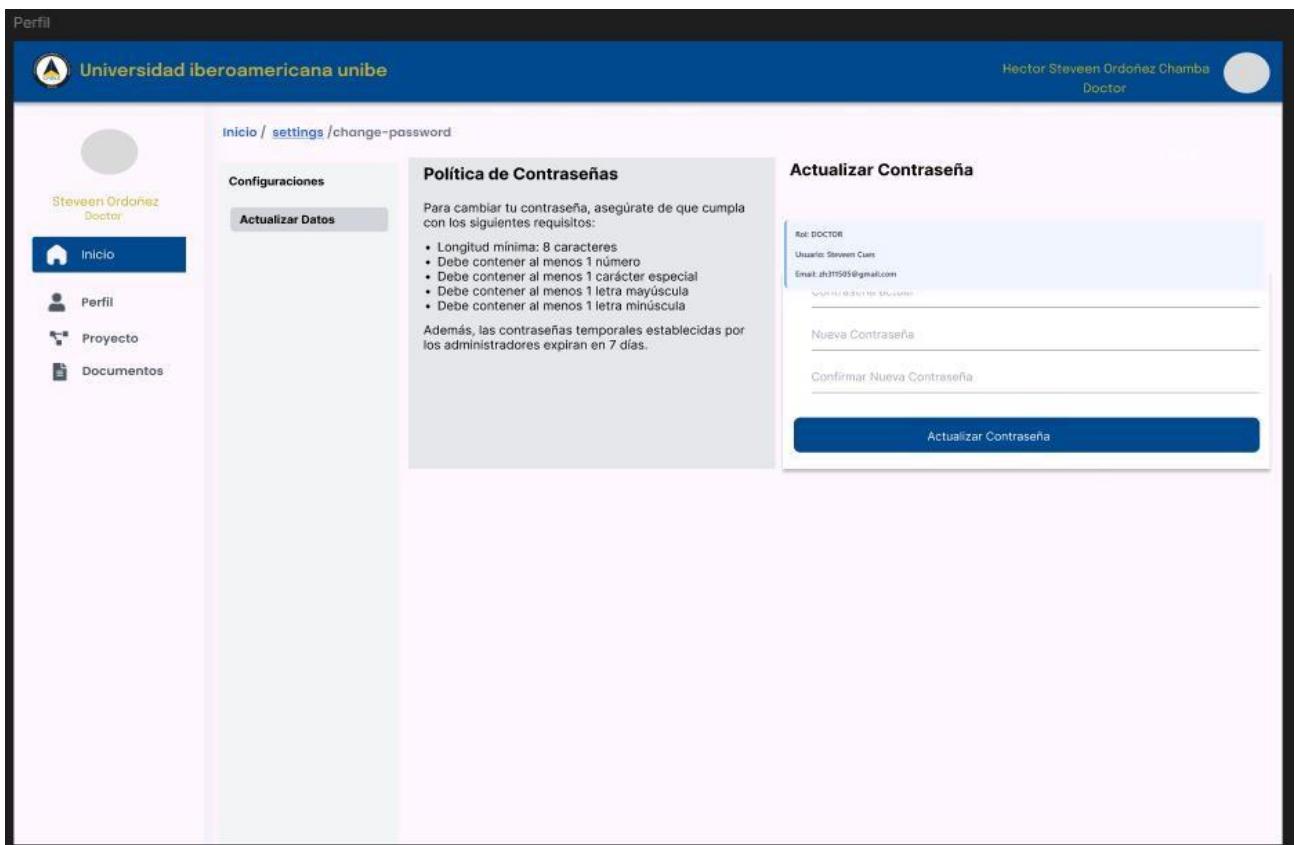


Figura 47. Diseño de interfaz de usuario (Personal Médico Perfil).

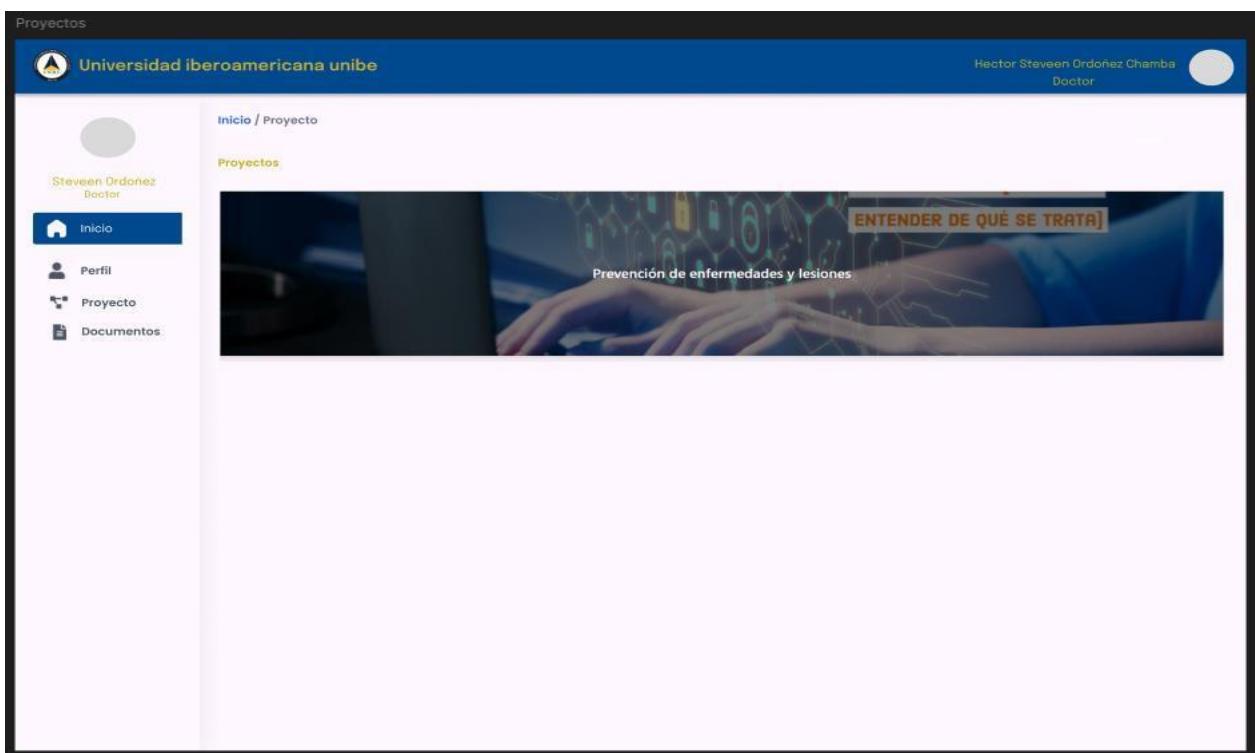


Figura 48. Diseño de interfaz de usuario (Personal Médico Proyectos).

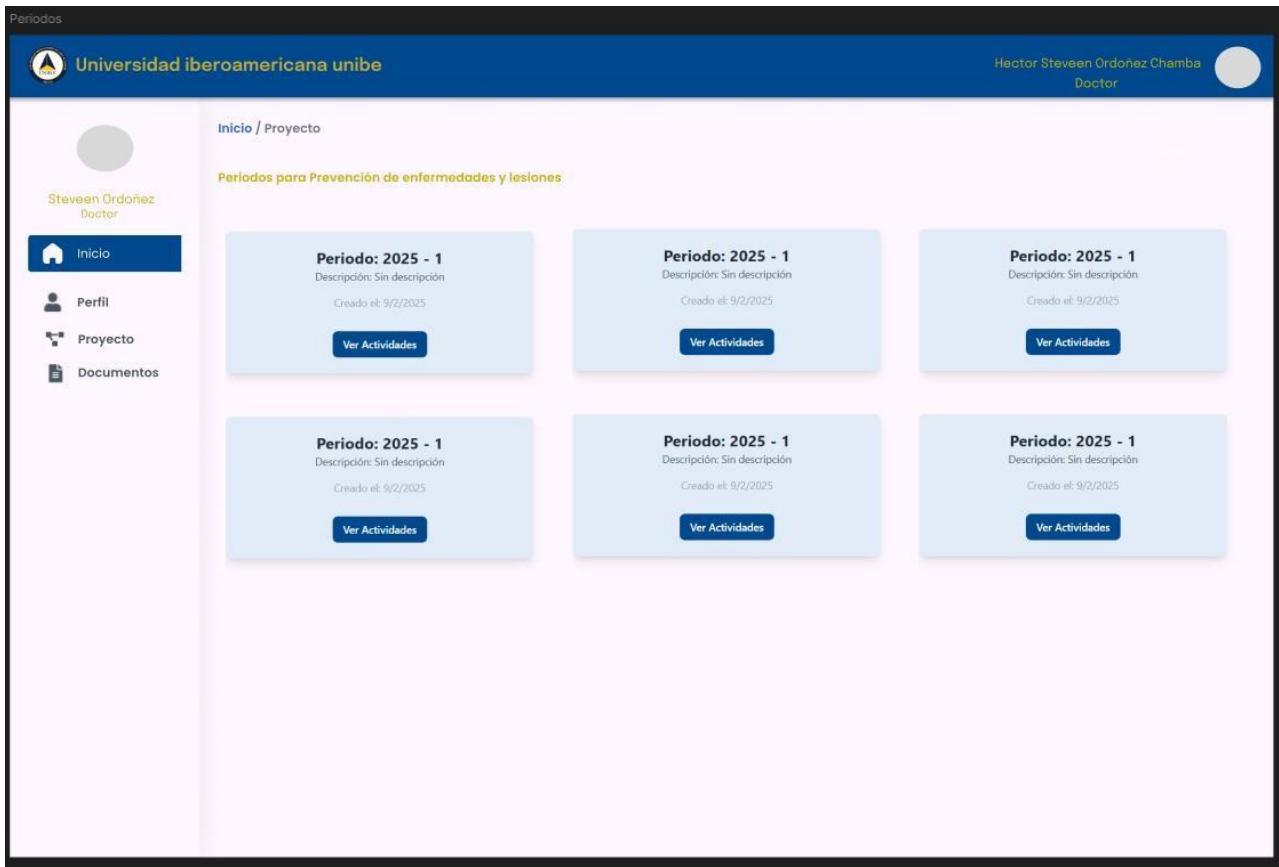


Figura 49. Diseño de interfaz de usuario (*Personal Médico Periodos*).

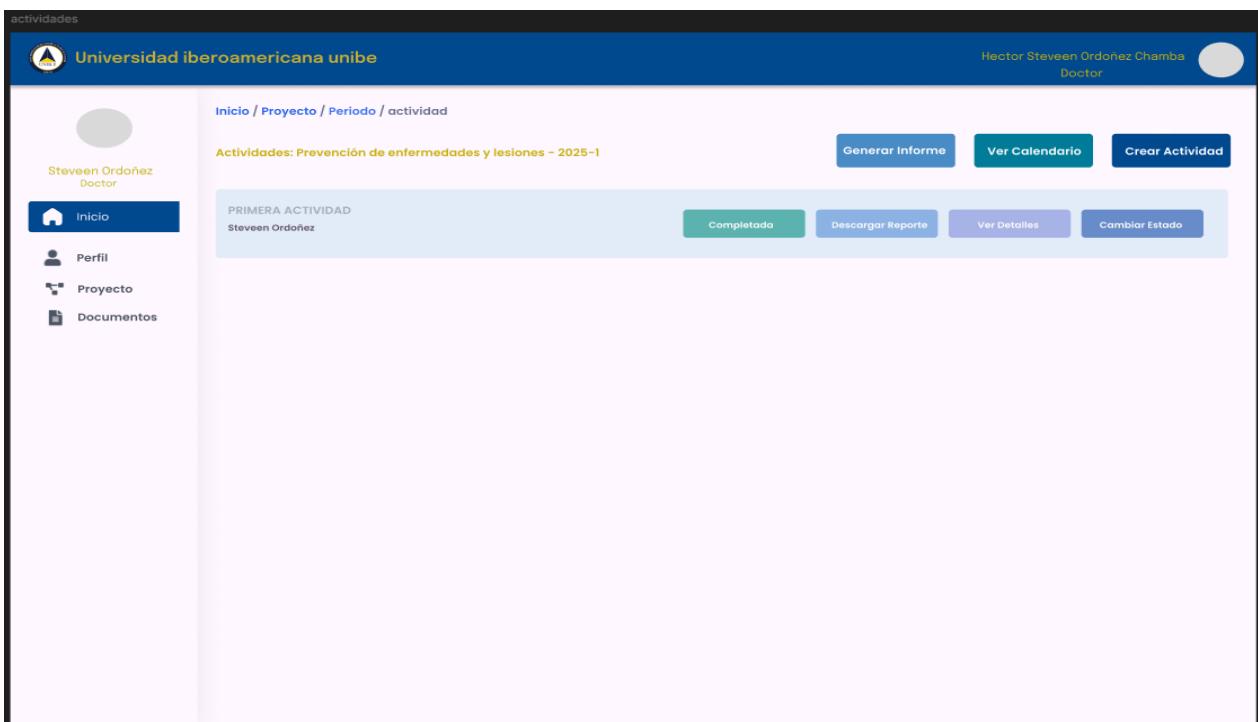


Figura 50. Diseño de interfaz de usuario (*Personal Médico Actividades*).

crear actividades

Universidad iberoamericana unibe

Hector Steveen Ordonez Chamba
Doctor

Inicio / Proyecto / Periodo / actividad / crear-actividad

Steveen Ordonez
Doctor

Inicio

Perfil

Proyecto

Documentos

	BENESTAR UNIVERSITARIO	CÓDIGO: PÁGINA: 1 DE 9
INFORME DE ACTIVIDADES REALIZADAS		
VERSIÓN: 001		
FECHA DE ACTUALIZACIÓN:		

BENESTAR UNIVERSITARIO

1. Datos Generales

Nombre del Proyecto: Prevención de enfermedades y lesiones
Institución Ejecutora: unibe
Responsable del Proyecto: Steveen Ordonez
Cargo: profesor
Unidad: orientel
Periodo del informe: 2023 - 1

2. Objetivo General

3. Actividades Realizadas

4. Número de Participantes

5. Presupuesto utilizado

6. Anexos

[Anterior](#) [Siguiente](#)

Crear Actividad

Nombre de la actividad

Responsable del Proyecto

Cargo

dd/mm/aaaa

dd/mm/aaaa

dd/mm/aaaa

Institución ejecutora

Unidad

Objetivo general

Número de participantes

Presupuesto utilizado

Actividades realizadas

Agregar Tarea

Adjuntar Documentos

Presupuesto

Seleccionar archivo Ningún archivo seleccionado

Lista de participantes

Seleccionar archivo Ningún archivo seleccionado

Fotos

Seleccionar archivo Ningún archivo seleccionado

Memorando de gestión

Seleccionar archivo Ningún archivo seleccionado

Artes gráficos de difusión

Seleccionar archivo Ningún archivo seleccionado

Crear Actividad

Figura 51. Diseño de interfaz de usuario (Personal Médico Crear Actividades).

Detalles de la actividad

Universidad iberoamericana unibe

Hector Steveen Ordonez Chamba
Doctor

Inicio / Proyecto / Periodo / actividad

Steveen Ordonez
Doctor

Inicio

Perfil

Proyecto

Documentos

Detalles de la actividad

	BENESTAR UNIVERSITARIO	CÓDIGO: PÁGINA: 1 DE 2
INFORME DE ACTIVIDADES REALIZADAS		
VERSIÓN: 001		
FECHA DE ACTUALIZACIÓN: 2023-02-09		

Primera Actividad

BENESTAR UNIVERSITARIO

1. Datos Generales

Nombre del Proyecto: Prevención de enfermedades y lesiones
Institución Ejecutora: unibe
Responsable del Proyecto: Steveen Ordonez
Cargo: profesor
Unidad: orientel
Periodo del informe: 2023 - 1

2. Objetivo General

objetivo

3. Actividades Realizadas

Nombre: SEGUNDA
Descripción: asa

Nombre: primera
Descripción: asa

4. Número de Participantes

344

5. Presupuesto utilizado

3434

6. Anexos

Presupuesto
Este archivo no es una imagen o está en formato no soportado.
[Descargar Presupuesto](#)

Artes gráficos de difusión
Este archivo no es una imagen o está en formato no soportado.

Figura 52. Diseño de interfaz de usuario (Personal Médico Visualizar Actividades).

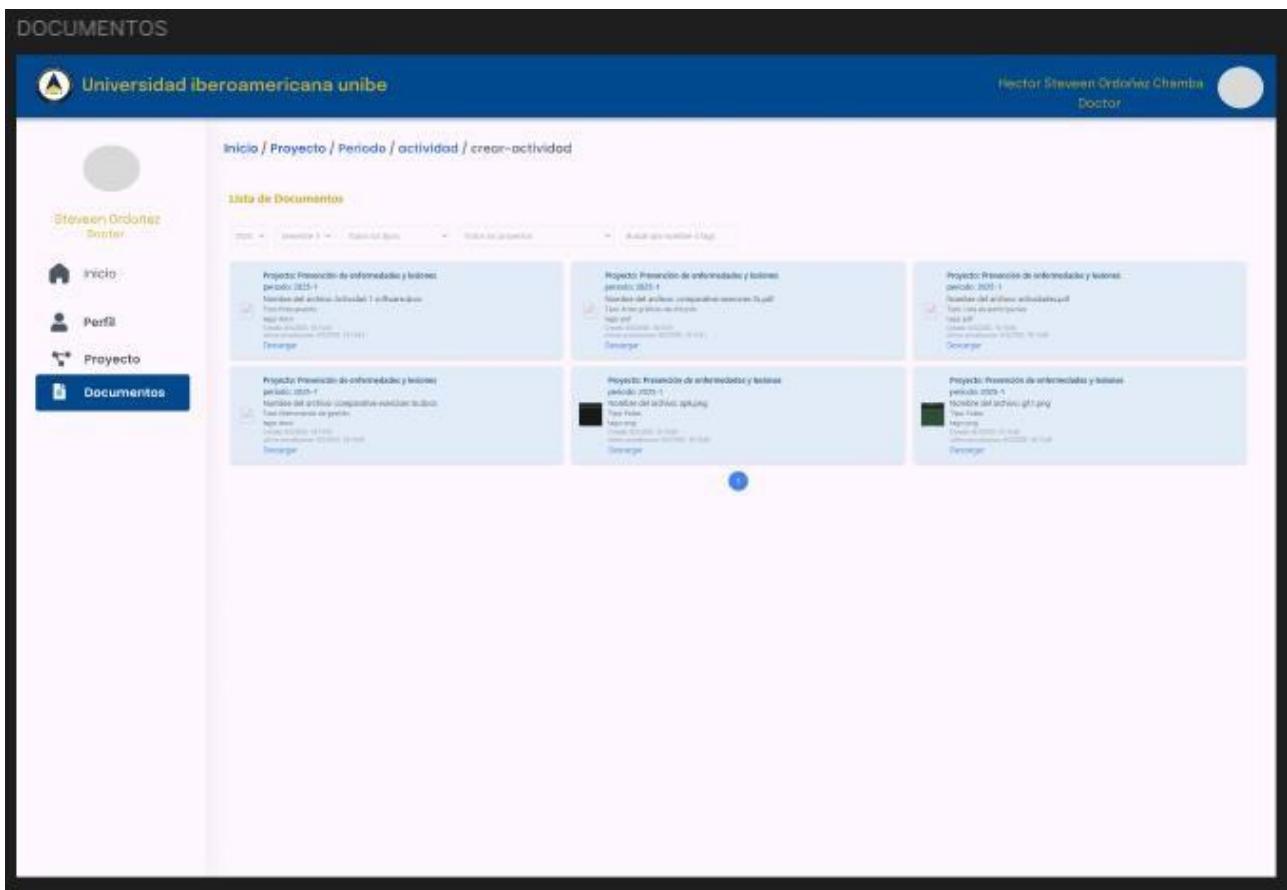


Figura 53. Diseño de interfaz de usuario (*Personal Médico Documentos*).

Este modelo de interfaz está enfocado en optimizar el rendimiento y la disposición de los datos en el sector de salud de la Dirección de Bienestar Universitario de la Institución de Educación Superior. Cada una de las pantallas ha sido creada para asegurar una interacción intuitiva y simplificar el manejo de documentos en la institución.

Diseño de arquitectura

En el proyecto se empleó la arquitectura Modelo-Vista-Controlador (MVC), que separa la presentación, la lógica corporativa y la gestión de datos. Este diseño permite un crecimiento por módulos, lo que facilita la administración y la ampliación del sistema.

El backend se creó usando AWS Amplify, una solución serverless que permite una integración eficiente con otro servicio de AWS, simplificando procesos como la autenticación, el almacenamiento de datos y la administración de API. Se empleó React en el frontend, lo que posibilitó una interfaz dinámica y de respuesta. Para la administración de datos, se utilizó DynamoDB, una base de datos NoSQL con gran capacidad de escalabilidad y alto desempeño.

Asimismo, el sistema fue creado con un modelo monolítico, concentrando todos sus elementos dentro de una sola aplicación. Esto facilita la gestión del software y posibilita una implementación más organizada.

Se implementó la estructura en Amazon Web Services (AWS), empleando servicios como AWS Cognito para la validación de usuarios y S3 para la protección de documentos. Esta resolución proporciona escalabilidad, resguardo y amplia disponibilidad.

Escalamiento en la nube

Nuestra arquitectura en la nube, basada en AWS Amplify, permite una escalabilidad automática y eficiente sin necesidad de gestionar infraestructura física. Gracias a los servicios gestionados de AWS, la capacidad de cómputo se ajusta dinámicamente según la demanda, optimizando tanto el rendimiento como los costos operativos.

Esta flexibilidad nos permite escalar la aplicación de manera eficiente a medida que crecen los volúmenes de tráfico y cambian los requisitos de latencia. Además, podemos aprovechar servicios adicionales de AWS para ajustarlos a nuevos requerimientos además que AWS nos permite incrementar o reducir los recursos computacionales de manera automática, asegurando un equilibrio óptimo entre desempeño y costos.

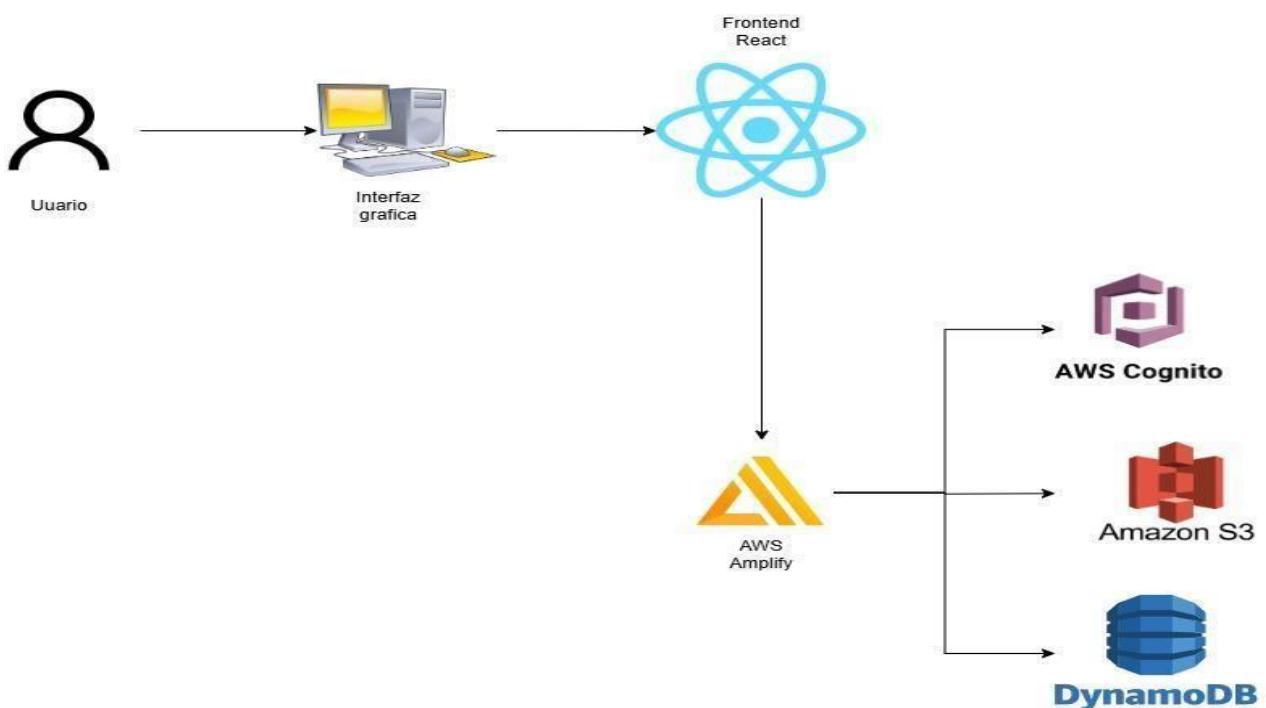


Figura 54. Arquitectura del sistema.

Desarrollo

Tecnologías utilizadas

En esta sección se detallan las tecnologías empleadas en el desarrollo del sistema web de bienestar universitario para la gestión documental del área de medicina en la institución de educación superior.

Se desarrolló el backend del sistema utilizando AWS Amplify, una solución basada en la nube que facilita la gestión del backend de forma escalable y sin requerir de servidores. Amplify facilita el enlace con otros servicios de Amazon Web Services, tales como AWS Cognito para la comprobación de usuarios, AWS Lambda para la realización de funciones remotas y Amazon S3 para el almacenamiento de archivos. Se eligió DynamoDB para administrar la base de datos, un sistema NoSQL que brinda una gran disponibilidad y eficacia. El desarrollo principal se llevó a cabo con React, una librería de JavaScript que permite la construcción de interfaces de usuario tanto dinámicas como extensibles. Gracias a su DOM virtual, React promueve la generación de componentes reutilizables y optimiza el desempeño de la aplicación.

Se optó por Visual Studio Code para la edición y desarrollo del código fuente, un editor liviano y altamente personalizado que proporciona soporte para extensiones, simplificando la labor con tecnologías como JavaScript, AWS y React.

El sistema se desarrolló para que pueda ser usado desde diversos dispositivos y navegadores, asegurando una experiencia de usuario segura y excelente. El sistema ofrece escalabilidad, protección y una administración sencilla gracias a la infraestructura en la nube.

Producto de software desarrollado

La aplicación está alojada en un repositorio de GitHub, donde se encuentra todo el código fuente. El repositorio es público, por lo que cualquier persona puede acceder.

Enlace al repositorio:

<https://github.com/unibe-projects/starter-unibe-docs.git>

Pruebas de Carga y Estrés

Para evaluar el rendimiento y la estabilidad del sistema bajo condiciones de carga y estrés, se realizaron pruebas utilizando la herramienta **Artillery**. Esta sección presenta los resultados obtenidos durante las pruebas de rendimiento de la API GraphQL, enfocándose especialmente en la creación de actividades y documentos, y analizando el impacto al escalar hasta 120 usuarios concurrentes.

Se llevaron a cabo pruebas simulando diversos niveles de carga y documentando métricas cruciales de rendimiento (KPI). Los tres casos estudiados mediante un método numérico serán: 40, 80 y 120 usuarios simultáneamente, cada uno con una duración de 60 segundos.

Metodología

Se realizaron evaluaciones de desempeño con Artillery.io, un instrumento particular para evaluar el desempeño de aplicaciones que emplean APIs. Se generó un documento de configuración en YAML con el objetivo de simular la interacción de usuarios virtuales con la API GraphQL del sistema, replicando escenarios de carga genuina.

Parámetros de prueba:

- Duración de la prueba: 60 segundos.
- Tasa de solicitudes: 40, 80 y 120 solicitudes por segundo (progresión aritmética para evaluar el escalamiento).
- Método de evaluación: Desarrollo de iniciativas a través de la API GraphQL.

Métricas evaluadas:

1. Parámetros de rendimiento:
 - Códigos de respuesta **HTTP**.
 - Tiempos de respuesta promedio y percentiles (**p95, p99**).
 - Tasa de solicitudes exitosas vs. fallidas (% de éxito).
2. Uso de recursos computacionales:
 - **Uso del procesador (CPU)** en porcentaje.
 - **Consumo de memoria (RAM)** en porcentaje.

A continuación, se presentan los resultados obtenidos en cada uno de los tres escenarios evaluados:

Escenario 1: 40 Usuarios Concurrentes

La prueba inicial simuló 40 usuarios concurrentes interactuando con la API. Los resultados se muestran en la Tabla 21.

Tabla 30. Pruebas de carga y estrés primer escenario 40 usuarios.

Métrica	Valor
Solicitudes Totales	2400
Solicitudes Exitosas (Código 200)	2400
Tasa de Solicitudes Exitosas	100%
Solicitudes Fallidas	0
Tiempo de Respuesta	
Mínimo	92 ms
Máximo	749 ms
Promedio	120.7 ms
Mediana	113.3 ms
p95 (Percentil 95)	159.2 ms
p99 (Percentil 99)	301.9 ms
Tasa de Respuesta (200)	2400
Usuarios Virtuales Completados	2400
Usuarios Virtuales Creados	2400
Crear Actividad	1170
Crear Documento	1230
Duración de la Sesión de los Usuarios	
Mínimo	198.8 s

Máximo	1526.9 s
Promedio	257.8 s
Mediana	232.8 s
p95 (Percentil 95)	361.5 s
p99 (Percentil 99)	804.5 s

Indicadores de Rendimiento:

- Códigos de respuesta HTTP:
 - Solicitudes Exitosas (Código 200): 2400
 - Solicitudes Fallidas: 0
 - Tasa de Solicitudes Exitosas: 100%
- Tiempos de Respuesta:
 - Promedio 120.7 ms
 - Mediana: 113.3 ms
 - P95 (Percentil 95): 159.2 ms
 - P99 (Percentil 99): 301.9 ms

Uso de Recursos Computacionales:

- CPU: Uso del CPU del 15%
- RAM: Uso del CPU 25%

Escenario 2: 80 Usuarios Concurrentes

La prueba con 80 usuarios concurrentes se ejecutó durante 80 segundos. Los resultados se muestran en la Tabla 22.

Tabla 31. Métricas de Rendimiento para 80 Usuarios Concurrentes

Métrica	Valor
Solicitudes Totales	4800
Solicitudes Exitosas (Código 200)	4800
Tasa de Solicitudes Exitosas	100%
Solicitudes Fallidas	0
Tiempo de Respuesta	
Mínimo	90 ms
Máximo	1623 ms
Promedio	116.7 ms
Mediana	111.1 ms
p95 (Percentil 95)	144 ms
p99 (Percentil 99)	219.2 ms
Tasa de Respuesta (200)	4800
Usuarios Virtuales Completados	4800
Usuarios Virtuales Creados	4800
Creación de Actividad	2417
Creación de Documento	2383
Duración de la Sesión de los Usuarios	
Mínimo	198.1 s
Máximo	1864.7 s
Promedio	252.1 s

Mediana	228.2 s
p95 (Percentil 95)	333.7 s
p99 (Percentil 99)	788.5 s

Indicadores de Rendimiento:

- Códigos de respuesta HTTP:
 - Solicitudes Exitosas (Código 200): 4800
 - Solicitudes Fallidas: 0
 - Tasa de Solicitudes Exitosas: 100%
- Tiempos de Respuesta:
 - Promedio: 116.7 ms
 - Mediana: 111.1 ms
 - p95 (Percentil 95): 144 ms
 - p99 (Percentil 99): 219.2 ms

Uso de Recursos Computacionales:

- CPU: Uso del CPU 30%
- RAM: Uso del CPU 40%

Nota: La prueba de 80 usuarios mostró métricas ligeramente fluctuantes durante la prueba. Los valores dados representan el rango de valores observados.

Escenario 3: 120 Usuarios Concurrentes

La evaluación con 120 participantes al mismo tiempo se llevó a cabo por un periodo de 120 segundos. Los hallazgos se presentan en la tabla.

Tabla 32. Métricas de Rendimiento para 120 Usuarios Concurrentes

Métrica	Valor
Solicitudes Totales	7200
Solicitudes Exitosas (Código 200)	7200
Tasa de Solicitudes Exitosas	100%
Solicitudes Fallidas	0
Tiempo de Respuesta	
Mínimo	92
Máximo	686
Promedio	119.5
Mediana	117.9
p95 (Percentil 95)	162.4
p99 (Percentil 99)	424.2
Tasa de Respuesta (200)	7200
Usuarios Virtuales Completados	7200
Usuarios Virtuales Creados	7200
Creación de Actividad	3154
Creación de Documento	2805
Duración de la Sesión de los Usuarios	
Mínimo	200.6
Máximo	1074.6
Promedio	265

Mediana	257.3
p95 (Percentil 95)	327.1
p99 (Percentil 99)	889.1

Indicadores de Rendimiento:

- Códigos de Respuesta HTTP:
 - Solicitudes Exitosas (Código 200): 7200
 - Solicitudes Fallidas: 0
 - Tasa de Solicitudes Exitosas: 100%
- Tiempos de Respuesta:
 - Promedio: 119.5
 - Mediana: 117.9
 - p95 (Percentil 95): 162.4
 - p99 (Percentil 99): 424.2

Uso de Recursos Computacionales:

- CPU: Uso del CPU 55%
- RAM: Uso del CPU 65%

Optimización del Rendimiento y Uso de Recursos Computacionales

Para una correcta evaluación del rendimiento de la API bajo pruebas de carga y estrés, es fundamental analizar no solo los tiempos de respuesta y la tasa de éxito de las solicitudes, sino también el impacto en los recursos computacionales. En este caso, se evaluaron dos indicadores clave adicionales: **el uso del procesador (CPU) y el consumo de memoria (RAM)**. Estos indicadores proporcionan una visión más completa del comportamiento de la API bajo diferentes niveles de carga, permitiendo identificar posibles cuellos de botella relacionados con los recursos.

Resultados y Análisis

Escenario 1: 40 Usuarios Concurrentes

- **CPU:** 15%
- **RAM:** 25%
- **Análisis:** Con una carga baja, el uso de recursos es bastante moderado. La API está manejando eficientemente las solicitudes, y los tiempos de respuesta son estables. El consumo de CPU y memoria RAM es reducido, lo que sugiere que el sistema opera dentro de sus límites sin requerir mejoras adicionales en esta etapa.

Escenario 2: 80 Usuarios Concurrentes

- **CPU:** 30%
- **RAM:** 40%
- **Análisis:** El aumento en el número de usuarios puede provocar un aumento en el uso de CPU y RAM. Aunque siguen dentro de límites aceptables, el incremento en la carga indica que el sistema necesita más recursos para gestionar las solicitudes, lo que podría provocar algún tipo de demora si no se aplican medidas adicionales. En esta área, podríamos mejorar componentes como la base de datos o la aplicación de caché para incrementar la eficiencia.

Escenario 3: 120 Usuarios Concurrentes

- **CPU:** 55%
- **RAM:** 65%
- **Análisis:** Conforme el número de usuarios se incrementa a 120, el uso de CPU y RAM se incrementa de forma notable. Aunque los tiempos de respuesta se mantienen en límites aceptables, el incremento en el tiempo máximo de respuesta y la utilización excesiva de recursos indican que la API podría estar próxima a su máxima capacidad. En este escenario, el sistema podría estar enfrentando dificultades para manejar la elevada demanda, lo que podría generar atascos, en particular en la gestión de peticiones simultáneas.

Optimización Propuesta

La investigación de los recursos informáticos revela que, aunque la API maneja correctamente cargas moderadas de usuarios al mismo tiempo, se topa con obstáculos al incrementar significativamente el número de usuarios. Para optimizar el desempeño y evitar el desplome del sistema en circunstancias de alta demanda, es esencial implementar las siguientes acciones:

1. **Mejora de la Base de Datos:** Evaluar las consultas y aplicar índices para incrementar la rapidez en el acceso y la modificación de datos.
2. **Implementación de Caché:** Implementar tácticas de caché que reduzcan el impacto en la base de datos y mejoren los tiempos de respuesta.
3. **Análisis de Código:** Realizar un estudio de la API para identificar potenciales aspectos críticos en la gestión de peticiones y optimizar el código.

En general, las pruebas de carga revelaron que la API es capaz de manejar cargas moderadas de usuarios concurrentes (hasta 120 usuarios).

Disponibilidad del sistema

El sistema se implementó en una instancia EC2 para ser evaluado por el equipo de Bienestar Universitario de la facultad de Medicina desde el lunes 3 de febrero de 2025 hasta el lunes 10 de febrero de 2025, totalizando así 7 días.

En este periodo, no se detectaron inconvenientes que afectaran el desempeño del sistema, lo que indica que se mantuvo operativo sin interrupciones.

Cálculo de Disponibilidad:

La disponibilidad del sistema se mide con la siguiente fórmula:

$$\text{Disponibilidad}(\%) = \left(\frac{\text{Tiempo Total} - \text{Tiempo de Inactividad}}{\text{Tiempo Total}} \right) \times 100$$

Dado que el sistema estuvo en operación 24 horas al día durante 7 días, el Tiempo Total es:

$$7 \text{ días} \times 24 \text{ horas/día} = 168 \text{ horas}$$

Como no se reportaron fallos, el Tiempo de Inactividad es 0 horas, por lo que la disponibilidad es:

$$(168-0 / 168) \times 100 = 100\%$$

Toda la evidencia del proceso de pruebas de carga y estrés se encuentra documentada en el Anexo 7.

Manual de Usuario

Por último, se ha creado un documento completo con el fin de asistir al usuario en la utilización correcta de la plataforma en línea. Este documento abarca un manual para los usuarios, en el que se detallan cómo moverse por la aplicación y las características que han sido incorporadas de acuerdo con los requisitos tanto funcionales como no funcionales definidos (Consultar Anexo 5). Además, se ha desarrollado una documentación técnica destinada a los programadores, ofreciendo datos esenciales sobre la estructura, la configuración y el cuidado del sistema (Consultar Anexo 4).

CAPÍTULO V

CONCLUSIONES Y RECOMENDACIONES

En el presente capítulo, se presentan las principales conclusiones obtenidas a partir del desarrollo del sistema web de bienestar universitario para la gestión documental en el área de medicina, así como recomendaciones para futuras mejoras e implementaciones.

Conclusiones

Considerando los objetivos definidos y la metodología utilizada, hemos llegado a las siguientes conclusiones:

- Con base a la encuesta aplicada a toda la población del sector médico y a un miembro del departamento administrativo para obtener los requerimientos tanto funcionales como no funcionales del sistema, se evaluaron sus respuestas para guiar las decisiones sobre los procedimientos de esta área. Esto facilitó el reconocimiento de las exigencias del equipo y la estipulación de las etapas a seguir en la creación del sistema web.
- Después de reunir los requisitos tanto funcionales como no funcionales del sistema web, se eligió la metodología más apropiada para el desarrollo. Se decidió utilizar la metodología en cascada, dado que se adapta de manera más efectiva a las necesidades y particularidades del desarrollo.
- Luego de reunir los requisitos y elegir la metodología más adecuada, se llevó a cabo un diseño en Figma, que se mostró al personal del área de medicina y administración. En esta fase, se confirmaron tanto las características visuales como los procedimientos realizados, lo que facilitó el desarrollo del sistema y aseguró que el sistema cumpla con las necesidades del usuario.
- Siguiendo la estructura de la metodología en cascada, se procedió al desarrollo del sistema aprovechando tecnologías actuales como React y AWS Amplify, además de sus servicios S3, Cognito, DynamoDB y EC2. Igualmente, se utilizó GitHub como plataforma para el control de versiones. Esta combinación aseguró que el sistema se ajustara a las especificaciones lo que significó la disminución de esfuerzo en los procesos realizados por el personal del área de medicina y administración.
- Después del desarrollo del sistema web, se realizó su verificación por el área de medicina y administración. Cabe destacar que se realizaron pruebas de carga y estrés, control de errores, documentación tanto como para usuarios como para desarrolladores y la disponibilidad del sistema durante la validación del sistema lo

que conllevo a la validación del sistema ya que cumplió con los requerimientos, diseños y procedimientos establecidas.

Recomendaciones

Para garantizar la sostenibilidad y mejora continua del sistema, se proponen las siguientes recomendaciones:

- Examinar la implicación de los alumnos en la recopilación de requisitos, puesto que esto no solo permitiría que el sistema maneje los archivos y tareas del sector médico, sino que también promovería el compromiso de la comunidad estudiantil al hacer más accesibles las actividades en las que participan.
- Incorporar alertas automáticas acerca de modificaciones o actualizaciones en las acciones realizadas por los usuarios esto permitirá conocer la información más reciente y acceder a los documentos subidos en el sistema de forma más rápida. Además, esto fomentará la claridad y fortalecerá la confianza en el sistema.
- Incluir un módulo que exhiba información acerca del número de archivos descargados, modificados y suprimidos, junto con las explicaciones que impulsan estas acciones, generará mayor seguridad en el sistema y garantizará la transparencia de los datos para los usuarios.
- Mejorar la accesibilidad del sistema en teléfonos móviles esto permitirá que los usuarios puedan controlar y acceder a la información desde cualquier plataforma esto hará que el sistema opere de forma eficaz y sin interrupciones en diversos dispositivos, asegurando una experiencia de usuario excelente en todas las plataformas.
- Incluir una sección que permita a los usuarios proporcionar sus opiniones sobre las actualizaciones según sus requerimientos esto no solo ayudará a realizar mejoras constantes en el sistema, sino que también hará posible detectar fallos que antes no se habían notado, lo que a su vez enriquecerá la experiencia general de los usuarios.

REFERENCIAS BIBLIOGRÁFICAS

- Acosta E., H. F., Borrero C., J. A., de Sistemas, C. A. V. S. D. de I., de Sistemas, D. de I., & de Sistemas, D. de I. (s.f.). *Sistema de gestión de información con herramientas de analítica de datos*. Edu.co. Recuperado el 11 de febrero de 2025, de <https://manglar.uninorte.edu.co/bitstream/handle/10584/11211/Proyecto%20final.pdf?sequence=1&isAllowed=y>
- Alvarado, L., & Valentino, A. (2024). *Automatización de la administración de procesos y su impacto en la plataforma tecnológica para la gestión de comunidades académicas y recreativas de la Universidad Técnica de Babahoyo*. Babahoyo: UTB-FAFI.
- Codina, L., Marcos, M. C., & Pedraza, R. (2023). *Sistemas web: Componentes y servicios interconectados*. Editorial Universitaria.
- Condori-Ojeda, P. (2020). *Universo, población y muestra*. Curso Taller. Recuperado de <https://www.aacademica.org/cporfirio/18.pdf>
- Delgado Olivera, L. de la C., & Díaz Alonso, L. M. (2021). *Modelos de desarrollo de software*. Revista Cubana de Ciencias Informáticas, 15(1), 37–51. Recuperado de http://scielo.sld.cu/scielo.php?pid=s2227-18992021000100037&script=sci_arttext
- Escuela Profesional de Ingeniería de Sistemas. (s.f.). *Universidad Privada de Tacna*. Recuperado de <https://repositorio.upt.edu.pe/bitstream/handle/20.500.12969/2509/Canahua-Choque-Luis.pdf>
- Generator, M. (s.f.). *Use of Web applications for the management of veterinary clinics and their impact on the improvement of administrative processes*. Gdeon.org. Recuperado el 9 de febrero de 2025, de <https://journals.gdeon.org/index.php/esj/article/view/174/250>
- Gómez, A., & López, M. del C. (s.f.). *Un modelo de estimación de proyectos de software*. Recuperado de <https://surl.li/zigrja>

Hernández, R., Fernández, C., & Baptista, P. (2020). *Metodología de la investigación* (6^a ed.). McGraw Hill. Recuperado de <https://www.esup.edu.pe/wp-content/uploads/2020/12/2.%20Hernandez,%20Fernandez%20y%20Baptista-Metodolog%C3%A3da%20Investigacion%20Cientifica%20ta%20ed.pdf>

Importancia del análisis de datos en la toma de decisiones. (s.f.). Edu.pe. Recuperado el 10 de febrero de 2025, de https://www.esan.edu.pe/conexion-esan/importancia-del-analisis-de-datos-en-la-toma-de-decisiones?utm_source=chatgpt.com

Martín, A. E., Chávez, S. B., Flores, S. R., Zogbe, A. S., Rodríguez, N., & Murazzo, M. A. (s.f.). *Desarrollo de una app móvil de nueva generación*. Universidad Nacional de La Plata. Recuperado de <https://sedici.unlp.edu.ar/bitstream/handle/10915/163837/Ponencia.pdf-PDFA.pdf>

Murgueytio, F. M., Galarza, P. J., & Barrientos, A. (2022). *Proceso de automatización de pruebas de aplicaciones web desarrolladas con React, Angular, Ant y Laravel*. Memorias de la Conferencia Iberoamericana en Sistemas, Cibernética e Informática, 192–197.

Oa, D. E. H. (s.f.). Editorial Ediciones Futuro. Redalyc.org. Recuperado el 11 de febrero de 2025, de <https://www.redalyc.org/journal/3783/378366538003/378366538003.pdf>

Rodríguez Mendoza, J. A. (2023). *Desarrollo del sistema web para la automatización del registro de ficha socioeconómica en la Unidad de Bienestar Universitario en la Universidad Nacional de Barranca 2020* [Tesis de licenciatura, Universidad Nacional de Barranca]. Recuperado de <https://repositorio.unjfsc.edu.pe/bitstream/handle/20.500.14067/7514/TESIS%20-%20JOBSEPH%20ALEXIS%20RODRIGUEZ%20MENDOZA-CORREGIDO.pdf>

Sistemas Computacionales, I. E. N. I. (s.f.). Universidad Técnica de Cotopaxi Extensión La Maná. Edu.ec. Recuperado el 11 de febrero de 2025, de <https://repositorio.utc.edu.ec/server/api/core/bitstreams/df822594-95c7-478b-8b44-82571e74e4cb/content>

Universidad Iberoamericana del Ecuador. (2012). *Reglamento de Bienestar Universitario*. Recuperado de <https://unibe.edu.ec/wp-content/uploads/2022/08/reglamento-bienestar-estudiantil-medico.pdf>

Vista de Transformación Digital en la Gestión Documental de una Institución de Educación Superior Tecnológica Pública: Implementación de un Sistema Web con Metodología XP. (s.f.). *Edu.pe*. Recuperado el 9 de febrero de 2025, de <https://revistas.une.edu.pe/index.php/QantuYachay/article/view/54/54>

Vista de Metodologías Ágiles y su Contribución al Trabajo en Equipo: Una Revisión Narrativa del Rol de Scrum en la Cohesión y Productividad Organizacional. (s.f.). *Innovasciencesbusiness.org*. Recuperado el 11 de febrero de 2025, de <https://www.innovasciencesbusiness.org/index.php/ISB/article/view/195/193>

Vega, E. (2023). *Guía metodológica para la elaboración de trabajos de investigación, ESGE, 2023-2025*. Academia.edu. Recuperado el 15 de febrero de 2025, de https://www.academia.edu/96457915/Gu%C3%A1da_metodol%C3%B3gica_para_la_elaboraci%C3%B3n_de_trabajos_de_investigaci%C3%B3n_ESGE_2023_2025

Zambrano Plúa, I. E., Quindemil Torrijo, E. M., & Rumbaut León, F. (2021). *Gestión documental en universidades: Una mirada desde Latinoamérica*. *ReHuSo: Revista de Ciencias Humanísticas y Sociales*, 6(Especial), 108–119. Recuperado de <https://www.redalyc.org/journal/6731/673171216010/673171216010.pdf>

ANEXOS

Anexo 1. Cuestionario para recopilación de requerimientos

Tabla 33. Operacionalización de Variables (Levantamiento de requerimientos)

Variable	Definición Conceptual	Definición Operacional	Dimensiones	Indicadores	Ítem
Requerimientos funcionales del sistema para la gestión documental del área de medicina	Actividades específicas que el sistema debe cumplir para gestionar eficientemente la documentación del área de medicina.	Conjunto de funcionalidades específicas del sistema que permiten la gestión eficiente de la documentación.	Gestión de usuarios	Autentificación de usuario	1,2,3,4
				Gestión documental	Seguridad de documentos 5,6,7,8,9
				Registrar documentos	10,11,12
				Modificar documentos	13
				Eliminar documentos	14
			Gestión de archivos	Búsqueda de archivos	15
				Descargar archivos	16
				Calidad de Contraseñas	17
				Controles de acceso	18
				Encriptación	19
Requerimientos no funcionales del sistema para la gestión documental del área de medicina.	Restricciones en los servicios del sistema, abarcando restricciones vinculadas al tiempo de operación, la fase de elaboración y el respeto a normas.	Conjunto de restricciones y características que determinan el rendimiento y calidad del sistema.	Seguridad	Cantidad de usuarios concurrentes	20
				Facilidad de uso	21,22,23

Cuestionario para la Identificación de Requerimientos Funcionales y No Funcionales del Sistema web para la gestión documental del área de medicina e instituciones de educación superior

Objetivo del Instrumento:

Identificar los requerimientos funcionales y no funcionales del sistema web para la gestión documental del área de medicina e instituciones de educación superior.

Instrucciones:

Por favor, lea atentamente cada ítem presentado y seleccione la opción que considere correspondiente. Agradezco sinceramente el tiempo que dediqué a completar este cuestionario.

Instrumento 1.1 Operacionalización de Variables (Levantamiento de requerimiento/requisitos)

#	PREGUNTAS
1	¿El sistema debe permitir la autenticación de usuarios mediante correo electrónico y contraseña? • Si • No
2	¿El sistema debe soportar la recuperación de contraseñas mediante correo electrónico? • Si • No

3	<p>¿El sistema debe permitir a los usuarios actualizar su correo electrónico o contraseña desde su perfil?</p> <ul style="list-style-type: none"> • Si • No
4	<p>¿El sistema debe mantener la sesión del usuario activa después de cerrar la aplicación (sesión persistente)?</p> <ul style="list-style-type: none"> • Si • No
5	<p>¿El sistema debe permitir asignar permisos específicos a cada usuario para controlar el acceso a documentos?</p> <ul style="list-style-type: none"> • Si • No
6	<p>¿El sistema debe manejar diferentes niveles de acceso a los documentos (por ejemplo, lectura, edición, eliminación)?</p> <ul style="list-style-type: none"> • Si • No
7	<p>¿El sistema debe permitir el control de versiones de los documentos para mantener un historial de cambios?</p> <ul style="list-style-type: none"> • Si • No
8	<p>¿El sistema debe permitir el seguimiento del estado de los documentos (borrador, revisión, aprobado, archivado)?</p> <ul style="list-style-type: none"> • Si • No
9	<p>¿El sistema debe permitir que los documentos sean compartidos con usuarios específicos o grupos?</p> <ul style="list-style-type: none"> • Si • No
10	<p>¿El sistema debe permitir el registro de nuevos documentos en la base de datos?</p>

	<ul style="list-style-type: none"> • Si • No
11	<p>¿El sistema debe permitir el registro de metadatos al momento de la carga de los archivos?</p> <ul style="list-style-type: none"> • Si • No
12	<p>¿El sistema debe permitir la categorización de documentos por tipo y fecha?</p> <ul style="list-style-type: none"> • Si • No
13	<p>¿El sistema debe permitir la modificación de documentos existentes?</p> <ul style="list-style-type: none"> • Si • No
14	<p>¿El sistema debe permitir la eliminación de documentos?</p> <ul style="list-style-type: none"> • Si • No
15	<p>¿El sistema debe permitir la búsqueda de documentos mediante palabras clave?</p> <ul style="list-style-type: none"> • Si • No
16	<p>¿El sistema debe permitir la descarga de archivos?</p> <ul style="list-style-type: none"> • Si • No
17	<p>¿El sistema debe requerir contraseñas fuertes para la autenticación de usuarios?</p> <ul style="list-style-type: none"> • Si • No
18	<p>¿El sistema debe tener niveles de permisos para usuarios (por ejemplo, administrador, usuario regular)?</p> <ul style="list-style-type: none"> • Si • No

19	<p>¿El sistema debe cifrar los datos sensibles almacenados?</p> <ul style="list-style-type: none"> • Si • No
20	<p>¿El sistema debe ser capaz de manejar al menos 100 usuarios concurrentes sin degradar su rendimiento?</p> <ul style="list-style-type: none"> • Si • No
21	<p>¿El sistema debe ser accesible desde dispositivos móviles?</p> <ul style="list-style-type: none"> • Si • No
22	<p>¿El sistema debe ser fácil de usar y entender para usuarios con conocimientos básicos de informática?</p> <ul style="list-style-type: none"> • Si • No
23	<p>¿El sistema debe proporcionar mensajes de error claros y detallados en caso de fallos?</p> <ul style="list-style-type: none"> • Si • No

Anexo 2. Validación del producto/software.

Tabla 25. Operacionalización de Variables (*Validación del producto de Software*)

Variable	Definición Conceptual	Definición Operacional	Dimensiones	Indicadores	Ítem
Validar el funcionamiento del sistema web para la gestión documental del área de medicina	La prueba de software es el proceso de ejecución de un programa con la intención de encontrar errores. (Myers y Sanders, 2004)	La prueba de software es el proceso de ejecución de un programa con la intención de encontrar errores. (Myers y Sanders, 2004)	Rendimiento	Tiempo de respuesta del sistema	1
			Fiabilidad	Frecuencia de errores	2
			Disponibilidad	Porcentaje de disponibilidad	3
			Mantenibilidad	Documentación del código	4

Instrumento 1.2 Operacionalización de Variables (Validación del producto/software)

#	PREGUNTAS
1	¿El sistema funciona correctamente bajo pruebas de carga y estrés? <ul style="list-style-type: none">• Si• No
2	¿El sistema replica de manera confiable las funcionalidades y navegación del prototipo, asegurando que todos los flujos de interacción y componentes visuales funcionen de forma consistente y sin errores? <ul style="list-style-type: none">• Si• No
3	¿El sistema está disponible al menos el 95% del tiempo? <ul style="list-style-type: none">• Si• No
4	¿El sistema tiene documentación completa para desarrolladores y usuarios? <ul style="list-style-type: none">• Si• No

Anexo 3. Validación del Instrumento por parte de los expertos.

URL:

<https://docs.google.com/document/d/1TFKn2m1cD3Wcqm6Glc0rJyRXml17OEzXovLuwS5vXw/edit?usp=sharing>

URL:

https://docs.google.com/document/d/1XiayOSQSOnUqZ_Qr_bkxEGrOLHPkfYeGDZ3GLrIqRyg/edit?usp=sharing

Anexo 4. Documentación para desarrolladores.

URL:

https://docs.google.com/document/d/1etg43tATb_iEGJ6Ij692m1bgrW7TR7X7/edit?usp=sharing&ouid=111127076113079881699&rtpof=true&sd=true

Anexo 5. Documentación para usuarios.

URL:

https://docs.google.com/document/d/1H2AV3bSE_1JkwahCDFNVNvUX_dnMvwBa/edit?usp=sharing&ouid=111127076113079881699&rtpof=true&sd=true

Anexo 6. Prototipos.

URL:

<https://www.figma.com/design/fT3vzV4E63pbFH88K9J96n/Untitled?node-id=0-1&t=xxolp5kEQjeBIp94-1>

Anexo 7. Pruebas de Carga y Estrés.

URL:

<https://drive.google.com/file/d/15EJxIM7Z6dUK4NuyERdZ0M8KScGITsr/view?usp=sharing>

Anexo 8. Link del Sistema.

URL:

<https://unibe-docs.netlify.app/login>

<https://github.com/unibe-projects/starter-unibe-docs>