

UNIVERSIDAD PÚBLICA DE EL ALTO

CARRERA INGENIERÍA DE SISTEMAS



PROYECTO DE GRADO

**“SISTEMA INTEGRAL PARA LA GESTIÓN DE RELACIONES
NACIONALES E INTERNACIONALES”**

**CASO: UNIDAD DE RELACIONES INTERNACIONALES UNIVERSIDAD
PÚBLICA DE EL ALTO**

Para Optar al Título de Licenciatura en Ingeniería de Sistemas

MENCIÓN: INFORMÁTICA Y COMUNICACIONES

Postulante : Miguel Huayhua Condori

Tutor Metodológico : Lic. Ing. Helen Fanny Suntura Escobar

Tutor Especialista : Lic. Ing. Yolanda Escobar Mancilla

Tutor Revisor : Lic. Santos Zenon Quispe Apaza

**EL ALTO – BOLIVIA
2024**

DECLARACIÓN JURADA DE AUTENTICIDAD Y RESPONSABILIDAD

Yo, Miguel Huayhua Condori, C.I. 9190304 mediante la presente declaro de manera pública que la propuesta del PROYECTO GRADO titulada “**SISTEMA INTEGRAL PARA LA GESTIÓN DE RELACIONES NACIONALES E INTERNACIONALES**” CASO: **UNIDAD DE RELACIONES INTERNACIONALES UNIVERSIDAD PÚBLICA DE EL ALTO** es original, siendo resultado de mi trabajo personal y no constituye una copia o réplica de trabajos similares elaborados, Autorizo la publicación del resumen de mi propuesta en internet y me comprometo responder a todos los cuestionamientos que se desprende de su lectura. Asimismo, me hago responsable ante la universidad o terceros, de cualquier irregularidad o daño que pudiera ocasionar, por el incumplimiento de lo declarado. De identificarse falsificación, plagio, fraude, o que el **PROYECTO DE GRADO** haya sido publicado anteriormente, asumo las consecuencias y sanciones que de mi acción se deriven, responsabilizándome por todas las cargas legales que se deriven de ello sometiéndome a las normas establecido y vigentes de la Carrera Ingeniería de Sistema de la Universidad Pública de El Alto.

El Alto, noviembre de 2024

Miguel Huayhua Condori

C.I. 9190304 L.P.

miguelhuayhuac@gmail.com

DEDICATORIA

Mi proyecto va dedicado a las personas que confiaron y pusieron su esfuerzo para brindarme las oportunidades de cumplir con este objetivo de vida.

A Dios, que tiene los planes y propósitos para quienes lo buscan y recompensa a quienes se esfuerzan y trabajan por un buen vivir.

A mi familia, que sin más, me dieron la oportunidad de llegar y luchar por mis sueños, con las enseñanzas y recomendaciones a lo largo del tiempo.

A los consejos de mi madre sobre la vida y al esfuerzo de mi padre en el trabajo duro y con dedicación que se que me servirán a tener un mejor control de las acciones que uno va realizando en el transcurso de la vida.

AGRADECIMIENTOS

Primeramente a Dios, que se quien es él quien nos brinda la fuerza para continuar con la lucha de cada objetivo.

A mi tutora metodológica Helen Fanny, quien nos brindaba los consejos necesarios para tener un trabajo bien hecho a presentar.

A mi tutor revisor Santos Zenón, quien fue de ayuda en los momentos necesarios en la presentación de nuestro proyecto.

A mi tutora especialista Yolanda Escobar, quien con la paciencia y confianza fue quien me brindó el primer paso para realizar mi esperado proyecto.

A la Unidad de Relaciones Internacionales de la UPEA, quien me abrió la posibilidad de realizar este proyecto y con el apoyo de los mismos.

Finalizando con la Carrera de Ingeniería de Sistemas, quien nos brindó el espacio y la posibilidad de dar el primer paso con este título que muchos anhelamos.

RESUMEN

Este proyecto aborda los desafíos derivados de la gestión desactualizada de los convenios, becas y oportunidades académicas en la Unidad de Relaciones Internacionales de la Universidad Pública de El Alto (UPEA), evidenciando la necesidad urgente de modernizar los procesos para lograr una gestión más eficiente y accesible que responda a las demandas actuales.

El sistema anterior no dispone de las herramientas necesarias para gestionar los datos de manera eficaz, lo que provoca retrasos en la respuesta a los usuarios y una carga administrativa excesiva, afectando negativamente la experiencia de los mismos.

Para superar estas deficiencias, se propone el desarrollo de un sistema integral utilizando tecnologías como Next.js, NextAuth, Prisma y Express.js, que permitirá centralizar la información y automatizar el registro y seguimiento de convenios, becas y oportunidades académicas, mejorando así la accesibilidad y eficiencia del proceso.

Como recomendaciones clave, se sugiere realizar pruebas exhaustivas de caja blanca, Stress, y accesibilidad para garantizar que el sistema cumpla con altos estándares de calidad y funcionalidad, además de asegurar que sea escalable y adaptable a las necesidades futuras de la universidad, fortaleciendo así el desarrollo académico y administrativo de la UPEA.

Palabras clave: Next.js, Convenios, Oportunidades, Open Up, Relaciones Internacionales.

ABSTRACT

This project addresses the challenges arising from the outdated management of agreements, scholarships, and academic opportunities at the International Relations Unit of the Public University of El Alto (UPEA), highlighting the urgent need to modernize processes to achieve more efficient and accessible management that meets current demands.

The previous system lacks the necessary tools to manage data effectively, leading to delays in user responses and excessive administrative burden, negatively impacting the user experience.

To overcome these deficiencies, the development of a comprehensive web system is proposed, using technologies such as Next.js, NextAuth, Prisma, and Express.js. This system will centralize information and automate the registration and tracking of agreements, scholarships, and academic opportunities, thus improving the accessibility and efficiency of the process.

Key recommendations include conducting thorough white-box, stress, acceptance, and accessibility tests to ensure that the system meets high standards of quality and functionality, while also ensuring it is scalable and adaptable to the university's future needs, thus strengthening the academic and administrative development of UPEA.

Keywords: Next.js, Agreements, Opportunities, Open Up, International Relations.

ÍNDICE GENERAL

CAPÍTULO I: MARCO PRELIMINAR	1
MARCO PRELIMINAR	2
1.1. INTRODUCCIÓN	2
1.2. ANTECEDENTES	3
1.2.1. Antecedentes Institucionales	3
1.2.2. Antecedentes afines al proyecto de grado	4
1.3. PLANTEAMIENTO DEL PROBLEMA	7
1.3.1. Problema Principal	8
1.3.2. Problemas Secundarios	8
1.3.3. Formulación del Problema	9
1.4. OBJETIVOS	9
1.4.1. Objetivo General	9
1.4.2. Objetivos Específicos	9
1.5. JUSTIFICACIÓN	10
1.5.1. Justificación Técnica	10
1.5.2. Justificación Económica	10
1.5.3. Justificación Social	11
1.6. METODOLOGÍA	11
1.6.1. Metodología de Desarrollo	11
1.6.2. Métricas de Calidad	12
1.6.3. Costo	12
1.6.4. Seguridad	12
1.6.5. Pruebas y Software	13
1.7. HERRAMIENTAS	14
1.8. LÍMITES Y ALCANCES	14
1.8.1. Alcances	14
1.8.2. Límites	15
1.9. APORTES	16
CAPÍTULO II: MARCO TEÓRICO	17
2. MARCO TEÓRICO	18

2.1. INTRODUCCIÓN	18
2.2. CONVENIOS	18
2.2.1. Convenio Marco	18
2.2.2. Convenio Específico	19
2.3. BECAS	20
2.3.1. Tipo de Becas	21
2.4. PASANTÍAS	23
2.5. EVENTOS ACADÉMICOS	23
2.5.1. Congreso	24
2.5.2. Convención	24
2.5.3. Seminario	24
2.5.4. Foro	24
2.6. SISTEMA	24
2.7. SISTEMA INTEGRAL DE INFORMACIÓN	25
2.8. TYPESCRIPT	25
2.9. ARQUITECTURA REST	26
2.10. REACT	26
2.11. NEXTJS	27
2.11.1. Principales Características de NextJs	27
2.12. NEXTAUTH	28
2.13. JWT (JSON WEB TOKEN)	30
2.14. EXPRESSJS	30
2.15. PRISMA	31
2.16. BCrypt	31
2.17. POSTGRESQL	31
2.18. SASS	32
2.19. AXIOS	32
2.20. RSUITE	32
2.21. MATERIAL UI	32
2.22. FRAMEWORK WEB	33
2.23. METODOLOGÍA OPEN UP	33
2.23.1. Fase de Inicio	34

2.23.2. Fase de Elaboración	34
2.23.3. Fase de Construcción	35
2.23.4. Fase de Transición	35
2.24. Norma ISO/IEC 25000	36
2.25. COCOMO II	39
2.25.1. Ecuaciones de Estimación	39
2.26. Pruebas de Stress	41
2.27. Pruebas de Caja Blanca	41
2.28. Pruebas de Caja Negra	42
2.29. Pruebas de Accesibilidad	42
CAPÍTULO III: MARCO APLICATIVO	44
3. MARCO APLICATIVO	45
3.1. INTRODUCCIÓN	45
3.2. METODOLOGÍA APLICADA (OPEN UP)	45
3.2.1. Entregables	45
3.2.2. Fase de Inicio	46
3.2.3. Fase de Elaboración	61
3.2.4. Fase de Construcción	77
3.2.5. Fase de Transición	80
CAPÍTULO IV: CALIDAD, COSTOS, SEGURIDAD Y PRUEBAS	93
4. INTRODUCCIÓN	94
4.1. GESTIÓN DE CALIDAD ISO/IEC 25000	94
4.1.1. Objetivos del Uso de Gestión de Calidad	94
4.1.2. Aplicación de la ISO 25010 (Calidad del Producto Software)	94
4.2. SEGURIDAD DEL SOFTWARE (ISO/IEC - 27000)	107
4.2.1. Integridad y Confidencialidad de la Información (ISO 27001)	107
4.3. PRUEBAS DEL SOFTWARE	113
4.3.1. Prueba de Caja Blanca	113
4.3.2. Prueba de Caja Negra	118
4.3.3. Pruebas de Stress	124
4.3.4. Pruebas de Accesibilidad	127
4.4. ANÁLISIS DE COSTOS	131

4.4.1. INTRODUCCIÓN	131
4.4.2. USO DE COCOMO II	131
CAPÍTULO V: CONCLUSIONES Y RECOMENDACIONES	141
5. CONCLUSIONES Y RECOMENDACIONES	142
5.1. CONCLUSIONES	142
5.2. RECOMENDACIONES	143
BIBLIOGRAFÍA	144
ANEXOS	149

INDICE DE FIGURAS

Figura 1	Organigrama de la Unidad de Relaciones Internacionales	4
Figura 2	Tipos de Convenios	20
Figura 3	Tipos de Becas	21
Figura 4	Fases de la Metodología OpenUP	34
Figura 5	Pasos para la Fase de Elaboración	35
Figura 6	Ejemplo de Caja Blanca	42
Figura 7	Ejemplo de Caja Negra	42
Figura 8	Boceto de Inicio de la Aplicación	51
Figura 9	Modal de Inicio de Sesión	52
Figura 10	Lista de Vídeos y Modal Reproductor	53
Figura 11	Boceto Convenios Dashboard	53
Figura 12	Boceto de Edición de Convenios	54
Figura 13	Boceto para el Visitante de Convenios	54
Figura 14	Boceto para el Visitante de Becas	55
Figura 15	Boceto Becas Dashboard	55
Figura 16	Boceto Edición de Becas	56
Figura 17	Boceto para el Visitante de Pasantías	56
Figura 18	Boceto Pasantías Dashboard	57
Figura 19	Boceto de Edición de Pasantías	57
Figura 20	Boceto de Eventos	58
Figura 21	Boceto Eventos Dashboard	58
Figura 22	Boceto de Edición de Eventos	59
Figura 23	Boceto de Noticias	59
Figura 24	Boceto Noticias Dashboard	60
Figura 25	Boceto Edición de Noticias	60
Figura 26	Diagrama de Paquetes del Sistema	61
Figura 27	Diagrama de Subsistemas del Proyecto	62
Figura 28	Esquema de Actores del Sistema	63
Figura 29	Caso de Uso Visitante	64
Figura 30	Caso de Uso del Administrador	65

Figura 31	Caso de Uso Usuario	66
Figura 32	Caso de Uso Participante de Beca	68
Figura 33	Caso de Uso de Inicio de Sesión	69
Figura 34	Caso de uso Gestión de Convenios	71
Figura 35	Caso de Uso Gestión de Becas	72
Figura 36	Modelo UML	74
Figura 37	Modelo Relacional de la Base de Datos	75
Figura 38	Esquema Funcional del Software	77
Figura 39	Esquema de la Estructura del Proyecto	78
Figura 40	Estructura del proyecto en ExpressJS	79
Figura 41	Ruta de Funciones de los Convenios	79
Figura 42	<i>Pantalla Principal e Inicio de Sesión del Sistema</i>	80
Figura 43	Vista de Gráficos del Sistema	81
Figura 44	Listado de Vídeos del Sistema	81
Figura 45	Listado y Edición de Convenios desde Dashboard	82
Figura 46	Agregado de Nuevos Convenios	82
Figura 47	Listado de Convenios desde Visitante	83
Figura 48	Visualización de Convenios	83
Figura 49	Listado y Edición de Pasantías desde Dashboard	84
Figura 50	Agregado de Nuevas Pasantías	84
Figura 51	Listado de Pasantías desde Visitante	85
Figura 52	Visualización de Pasantías	85
Figura 53	Listado y Edición de Becas desde Dashboard	86
Figura 54	Creación de Nueva Beca	86
Figura 55	Seguimiento de Participantes en Becas	87
Figura 56	Listado de Becas desde Visitantes	87
Figura 57	Visualización de Becas	88
Figura 58	Listado y Edición de Eventos desde Dashboard	88
Figura 59	Creación de Nuevo Evento	89
Figura 60	Listado de Eventos desde Visitantes	89
Figura 61	Visualización de Eventos	90
Figura 62	<i>Listado y Edición de Noticias desde Dashboard</i>	90

Figura 63	Creación de Nueva Noticia	91
Figura 64	Listado de Noticias para los Visitantes	91
Figura 65	Administración de Token JWT desde NextAuth	108
Figura 66	Token de Sesión del Sistema Gestionado en NextJs	109
Figura 67	Generación de Saltos para la Contraseña de Usuarios	110
Figura 68	Control de Usuarios para el Administrador	110
Figura 69	Modificación del Perfil de Usuario	111
Figura 70	Token CSRF Gestionado por NextAuth	111
Figura 71	Validación de Datos para el Módulo de Convenios	112
Figura 72	Modal de Inicio de Sesión del Usuario	112
Figura 73	Diagrama Ciclomática de Complejidad	114
Figura 74	Grafo Ciclomático de Complejidad	115
Figura 75	Ingreso de credenciales desde el login	119
Figura 76	Pantalla Principal de Bienvenida del Sistema	119
Figura 77	Prueba de Caja Negra - Creación de Convenio	120
Figura 78	Alerta del Sistema para el Usuario	121
Figura 79	<i>Alerta de Nuevo Convenio Registrado</i>	122
Figura 80	Datos Insertados en el Módulo de Convenios	122
Figura 81	Publicación Activa del Convenio Creado	123
Figura 82	Solicitudes por Segundo Durante un Minuto	126
Figura 83	Tiempo de Respuesta Durante un Minuto	126
Figura 84	Resultado para la Ruta Principal con 100 Usuarios	127
Figura 85	Análisis de Accesibilidad para la Página de Inicio	127
Figura 86	Análisis de Accesibilidad de Contenido	128
Figura 87	Aria-label, cabecera de ejemplo para becas	128
Figura 88	Imagen de Portada de Convenio	129
Figura 89	Modal de Muestra para las Portadas de Publicaciones	129
Figura 90	Responsividad del Sistema para Celulares y Tablet	130
Figura 91	Responsividad del Sistema para Computadoras	130
Figura 92	Acceso Dashboard desde Movil	131
Figura 93	<i>Lineas de Código del Sistema</i>	133

INDICE DE TABLAS

Tabla 1	Constantes Aplicados a los Tipos de Modelo COCOMO II	40
Tabla 2	Entregables del Proyecto	45
Tabla 3	Planificación del Proyecto	47
Tabla 4	Requisitos Funcionales del Proyecto	48
Tabla 5	Requerimientos no Funcionales del Proyecto	50
Tabla 6	<i>Operaciones del Visitante en el Sistema</i>	64
Tabla 7	Operaciones del Administrador en el Sistema	66
Tabla 8	Operaciones Realizadas por el Usuario	67
Tabla 9	Operaciones Realizadas por el Nuevo Participante	68
Tabla 10	Inicio de Sesión- Usuario y Administrador	70
Tabla 11	Gestión de Publicaciones y Convenios	72
Tabla 12	Gestión de Becas	73
Tabla 13	Parámetros de Medición el Sistema	95
Tabla 14	Puntos de Función sin Ajustar	95
Tabla 15	Parámetros de Medición de los Puntos de Función	96
Tabla 16	Valores de Escala de Eficiencia	99
Tabla 17	Preguntas de ponderación de eficiencia	100
Tabla 18	Escala de Usabilidad del Sistema	102
Tabla 19	Preguntas de Ponderación de Usabilidad	102
Tabla 20	Entornos Soportados del Sistema	105
Tabla 21	Resultados de Calidad Aplicando la ISO 25010	106
Tabla 22	Caminos del Sistema desde el Inicio de Sesión	117
Tabla 23	Caminos del Sistema Basado en el Tipo de Usuario	118
Tabla 24	Tipos de Datos de Inicio de Sesión	119
Tabla 25	Entradas de Registro de Convenio	121
Tabla 26	Parámetros Utilizados para la Prueba de Stress	125
Tabla 27	Parámetros Utilizados para la Prueba de Stress	125
Tabla 28	Tipo de Modelo COCOMO II	132
Tabla 29	Constantes Aplicados a los Tipos de Modelo COCOMO II	132
Tabla 30	Atributos Seleccionados como Valor de Ajuste	134

Tabla 31 Estimación Aproximada Usando COCOMO II 140

INDICE DE ECUACIONES

Ecuación 1	Funcionalidad del Sistema	37
Ecuación 2	Eficiencia del Sistema	38
Ecuación 3	Usabilidad del Sistema	38
Ecuación 4	Mantenibilidad del Sistema	38
Ecuación 5	Portabilidad del Sistema	39
Ecuación 6	Esfuerzo Dedicado en el Proyecto	40
Ecuación 7	Tiempo de Esfuerzo en Meses	41
Ecuación 8	Puntos de Función	98
Ecuación 9	Puntos de Función Máximo	98
Ecuación 10	Funcionalidad del Sistema	99
Ecuación 11	Eficiencia del Sistema	101
Ecuación 12	Usabilidad del Sistema	103
Ecuación 13	Índice de Madurez del Software	104
Ecuación 14	Grado de Portabilidad del Sistema	105
Ecuación 15	Complejidad Ciclomática	116
Ecuación 16	Esfuerzo Aplicado al Software	136
Ecuación 17	Tiempo Requerido para Desarrollar el Sistema	137
Ecuación 18	Número de personas Necesarias para el Proyecto	138
Ecuación 19	Costo Total del Proyecto	139

LISTADO DE ABREVIATURAS

WEB: "World Wide Web" (Red Informática Mundial)

UML: Lenguaje Unificado de Modelado (Unified Modeling Language)

COCOMO: Modelo Constructivo de Costos (Constructive Cost Model)

ISO: Organización Internacional de Normalización (International Organization for Standardization)

IEC: Comisión Electrotécnica Internacional (International Electrotechnical Commission)

CRUD: Crear, Leer, Actualizar, Eliminar (Create, Read, Update, Delete)

SGSI: Sistema de Gestión de Seguridad de la Información

SASS: Hoja de estilos sintácticamente impresionante (Syntactically Awesome Style Sheets)

VRAC: Valor de Referencia de Actividad Científica

CIDI: Centro Internacional de Desarrollo de la Información

TIC: Tecnologías de la Información y la Comunicación

SII: Sistema Integral de Información

MVC: Modelo Vista Controlador (Model-View-Controller)

REST: Transferencia de Estado Representacional (Representational State Transfer)

ES2016: ECMAScript 2016

JS: JavaScript

JSON: JavaScript Object Notation (Notación de Objetos de JavaScript)

JWT: JSON Web Token (Token Web JSON)

API: Interfaz de Programación de Aplicaciones (Application Programming Interface)

QA: Aseguramiento de Calidad (Quality Assurance)

UI: Interfaz de Usuario (User Interface)

UX: Experiencia de Usuario (User Experience)

XML: Lenguaje de Marcado Extensible (eXtensible Markup Language)

HTML: Lenguaje de Marcado de Hipertexto (HyperText Markup Language)

CSS: Hojas de Estilo en Cascada (Cascading Style Sheets)

W3C: World Wide Web Consortium (Consortio World Wide Web)

WCAG: Pautas de Accesibilidad para el Contenido Web (Web Content Accessibility Guidelines)

HTTP: Protocolo de Transferencia de Hipertexto (HyperText Transfer Protocol)

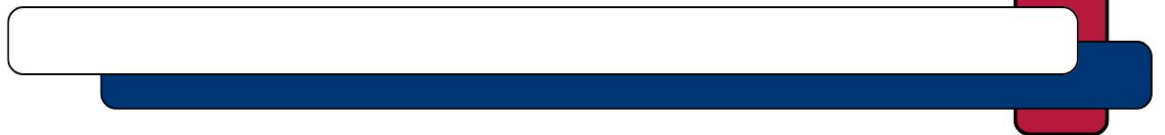
HTTPS: Protocolo Seguro de Transferencia de Hipertexto (HyperText Transfer Protocol Secure)

SQL: Lenguaje de Consulta Estructurado (Structured Query Language)

TCP/IP: Protocolo de Control de Transmisión/Protocolo de Internet (Transmission Control Protocol/Internet Protocol)

CAPÍTULO I: MARCO PRELIMINAR

**INGENIERÍA
DE SISTEMAS**
UNIVERSIDAD PÚBLICA DE EL ALTO



MARCO PRELIMINAR

1.1. INTRODUCCIÓN

La globalización ha incrementado la importancia de las relaciones internacionales en el ámbito académico, especialmente en universidades como la UPEA, donde la Unidad de Relaciones Internacionales gestiona convenios y colaboraciones con diversas instituciones; el sistema actual presenta limitaciones significativas para manejar la creciente cantidad de información que se genera, lo que dificulta la atención eficiente a la comunidad universitaria.

La principal problemática es la acumulación desorganizada de información administrativa y externa, lo que afecta la generación y publicación de datos claros y accesibles, y esto impide que estudiantes y docentes participen activamente en programas internacionales, además de reducir la visibilidad de las oportunidades ofrecidas por la Unidad.

Este proyecto busca desarrollar un sistema web integral que permita organizar y difundir la información de manera eficiente, facilitando la gestión interna y mejorando el acceso a convenios, becas y oportunidades; la plataforma moderna e interactiva aumentará la visibilidad de la Unidad y fomentará la participación de la comunidad universitaria.

La implementación del sistema se realizará mediante metodologías ágiles y herramientas tecnológicas avanzadas, como el desarrollo web y la gestión de bases de datos el cual esto garantizará que el sistema cumpla con las necesidades de la

Unidad y responda a las demandas de la comunidad universitaria de manera eficiente.

1.2. ANTECEDENTES

1.2.1. Antecedentes Institucionales

La unidad pertenece y trabaja a partir de rectorado de la Universidad Pública de El Alto, en donde es aquel que se encarga en recibir todos los convenios para después pasar por las respectivas secretarías tanto general y jurídica, llegando finalmente la información a la Unidad de Relaciones Internacionales que lleva funcionando desde el 2016 y su labor es en mantener las relaciones realizadas con demás universidad y empresas para el beneficio de la universidad.

1.2.1.1. Visión

Profundizar los procesos de coordinación, gestión, formalización y seguimiento de programas estratégicos y los acuerdos de cooperación con universidades nacionales y del exterior, así como las instituciones académicas, instituciones públicas, privadas, embajadas, consulados y organizaciones bilaterales y multilaterales que coadyuven en el proceso de internalización de la Educación Superior.

1.2.1.2. Misión

La Dirección de Relaciones Internacionales DRNI de la UPEA, es la encargada de la internacionalización del conocimiento académico, científico, tecnológico, de Interacción Social e Investigación en todos los campos del conocimiento, para

fortalecer y expandir los vínculos internacionales con universidades nacionales y extranjeras; de beneficio para docentes y estudiantes.

1.2.1.3. Objetivo

Promover el fortalecimiento de las relaciones de la Universidad Pública de El Alto (UPEA), con instituciones Nacionales, Internacionales relacionadas con la Educación Superior.

1.2.1.4. Organigrama

Figura 1

Organigrama de la Unidad de Relaciones Internacionales



Nota. Organigrama de la Unidad de Relaciones Internacionales desde Rectorado, Unidad de relaciones internacionales, 2024, fuente: Unidad de Relaciones Internacionales

1.2.2. Antecedentes afines al proyecto de grado

1.2.2.1. Antecedentes Internacionales

- En Fuentes (2021), en su proyecto “**Plan de mejoramiento en el sistema de información y atención al usuario institucional de una institución**

prestadora de servicios de salud Córdoba”, posee como objetivo plantear el uso de un modelo tecnológico para la medición de la satisfacción del cliente, que permita llevar a cabo un análisis confiable de la información suministrada, de tal forma que puedan identificarse de forma rápida y clara los aspectos críticos del servicio ofrecido.

- Según Estrada (2022), propone **“Diseño de un Sistema web para el control de Currículum Vitae Electrónico de personal docente basado en una arquitectura orientada a servicios (API REST)”**, que tiene por objetivo desarrollar e implementar un sistema de información basado en un entorno web para el registro y control de información de productividad académica del personal docente, para el proceso del desarrollo se utilizó la metodología lweb.
- En Vera (2022), **“Desarrollo de un sistema web de gestión de nómina para la automatización de procesos del área de recursos humanos en la Empresa Reciclajes y Procesos Ecológicos S.A.”**, tiene como finalidad el de reducir la carga administrativa y automatizar los procesos debido al trabajo manual que existe en la empresa, su desarrollo implicó el uso de la metodología XP (Extreme Programming).

1.2.2.2. Antecedentes Nacionales

- Según Mamani (2018), propone **“Sistema web de administración de recursos humanos”**, el proyecto tiene por objeto el desarrollar un sistema Web de administración de Recursos Humanos para el caso FRUTALERO S.R.L., el cual se encarga de optimizar el manejo de la información del plantel administrativo y

operativo, tomando como metodología base SCRUM el cual se encargó de gestionar el proceso de creación del proyecto.

- En Mamani (2020), con el proyecto “**Sistema de información para el seguro social universitario**”, tiene como objetivo principal desarrollar un sistema web de registro y acceso de servicios, de los estudiantes al seguro universitario, para la Dirección de seguro de salud de la Universidad Pública de El Alto, a fin de lograr una eficiente administración en el manejo y gestión de la información. Se realizó con la metodología UWE.

1.2.2.3. Antecedentes Locales

- Según Mamani (2020), “**Sistema web de control de seguimiento de servicios y gestión de clientes para la empresa Consultora Contadores Públicos & Auditores AYS S.R.L.**”, Universidad Mayor de San Andrés, proyecto que utiliza herramientas propias de las tecnologías de la información para presentar de forma, clara, sencilla y facilitar el diálogo razonable con el cliente para generar mayor rentabilidad, hizo el uso de la combinación de la metodología Kanban y UWE para el desarrollo del sistema web.
- En Pachari (2020), “**Portal Web para la gestión de información administrativa y académica**”, Universidad Pública de El Alto, es un sistema con el objetivo de apoyar la gestión de la información administrativa y académica, coadyuvando con la organización, disponibilidad y accesibilidad de la información de la Unidad Educativa Villa Alemania Fe y Alegría. En su

proceso de creación se hizo el uso de la metodología UWE que se adecuó al tipo de desarrollo del proyecto.

- Según Aruquipa (2020), “**Sistema web integrado de administración y control de bibliotecas**”, Universidad Pública de El Alto, un proyecto orientado a brindar el servicio de apoyo bibliográfico de consulta a la comunidad universitaria y población en general para así adquirir material necesario para su uso. Hizo el uso de la metodología UWE que le permitió especificar de mejor manera su aplicación web en todo el proceso para su creación.

1.3. PLANTEAMIENTO DEL PROBLEMA

La Unidad de Relaciones Internacionales de la UPEA enfrenta importantes desafíos en la administración de su información, derivados de la acumulación desorganizada de datos relacionados con convenios, becas y oportunidades internacionales gestionadas con universidades y empresas. Este desorden dificulta la generación y publicación de información clara y accesible para la comunidad universitaria, lo que limita el acceso de estudiantes y docentes a programas como pasantías e intercambios, afectando su participación activa. Además, la falta de detalle en las publicaciones genera incertidumbre y dudas entre los interesados, mientras que los procesos de registro y control son ineficientes y carecen de un adecuado seguimiento, lo que compromete la precisión y utilidad de los datos almacenados. Sumado a esto, la escasa visibilidad de la Unidad en medios de comunicación restringe su alcance, dificultando la promoción de las oportunidades que ofrece y reduciendo el interés en sus actividades. Esto evidencia la necesidad de

implementar mejoras en la gestión de la información y en las estrategias de difusión.

1.3.1. Problema Principal

La Unidad de Relaciones Internacionales enfrenta problemas de organización y visibilidad de la información que maneja, lo que limita su conocimiento y acceso por parte de los estudiantes interesados en oportunidades internacionales. No actualizar el sistema ha generado ineficiencias en los procesos administrativos, lo que hace urgente la necesidad de renovar y digitalizar su gestión para mejorar la accesibilidad, visibilidad y eficiencia en la atención de las demandas cambiantes.

1.3.2. Problemas Secundarios

- El escaso conocimiento en algunas becas disponibles reduce los participantes , dificultando cumplir los requisitos para su desarrollo.
- A pesar de la existencia de información relevante, la falta de detalles en las publicaciones actuales dificulta la resolución eficiente de las dudas de los interesados.
- Los procesos actuales de registro y control de participantes y convenios carecen de un adecuado seguimiento, afectando la disponibilidad y el orden de la información.
- La Unidad responsable no es suficientemente conocida y carece de la presencia esperada en medios de difusión, limitando su alcance y efectividad en atraer interesados.

1.3.3. Formulación del Problema

¿De qué manera se pueden mejorar los procesos administrativos de la Unidad de Relaciones Internacionales de la UPEA para maximizar el aprovechamiento de oportunidades nacionales e internacionales?

1.4. OBJETIVOS

1.4.1. Objetivo General

Desarrollar un sistema integral de gestión de información para la Unidad de Relaciones Internacionales de la Universidad Pública de El Alto, con el fin de optimizar los procesos administrativos y facilitar la difusión de convenios y sus beneficios derivados.

1.4.2. Objetivos Específicos

- Crear un módulo en el sistema para facilitar la participación de universitarios en las publicaciones de tipo beca derivados desde los convenios.
- Desarrollar un espacio centralizado que brinde acceso claro y detallado a convenios, becas y eventos actualizados, resolviendo eficazmente las dudas de los interesados.
- Digitalizar el registro y seguimiento de participantes y convenios en general, eliminando los procesos manuales y mejorando la eficiencia.
- Desarrollar una interfaz accesible que proporcione información clara y actualizada sobre la Unidad.

1.5. JUSTIFICACIÓN

Este proyecto se presenta como una iniciativa estratégica y esencial para la Universidad Pública de El Alto (UPEA) con un enfoque claro en el beneficio directo de la comunidad universitaria, con el objetivo primordial de mejorar el acceso a información relevante para los estudiantes, docentes y personal administrativo, específicamente relacionada con nuevas oportunidades académicas, tales como becas, ofertas y descuentos ofrecidos por instituciones afiliadas a la universidad.

1.5.1. Justificación Técnica

La implementación de un sistema integral constituirá una mejora significativa en la eficiencia operativa de la Unidad de Relaciones Internacionales de la UPEA. Esto se traducirá en una reducción sustancial de la carga de trabajo y permitirá una gestión más eficaz de todas las operaciones, en cuanto a la elaboración del sistema desde el desarrollo e implementación se necesitará en hardware un procesador de por lo menos de 2 núcleos, memoria RAM de 4GB y un almacenamiento de al menos 50GB, en software podemos destacar que el uso necesario de un sistema operativo radicaría en Ubuntu, Debian o CentOS.

1.5.2. Justificación Económica

La automatización de procesos y la eliminación de tareas manuales en el proyecto conllevarán a una reducción sostenida de los costos operativos a lo largo del tiempo y todo gracias a la mayor eficiencia en la gestión de recursos humanos, liberando al personal de tareas rutinarias y permitiéndoles enfocarse en becas estratégicas, lo que, a su vez, maximizará su productividad y el rendimiento de la

unidad.

Además, la mejora en la gestión de relaciones internacionales no solo beneficiará a la unidad, sino que también generará oportunidades de establecer alianzas institucionales estratégicas con otras universidades e instituciones.

1.5.3. Justificación Social

La mejora en la gestión de relaciones nacionales e internacionales mediante este proyecto tendrá un impacto social significativo en la comunidad universitaria y la sociedad en general, además que, en la optimización de los procesos y la centralización de la información contribuirán directamente a la mejora de la calidad educativa ofrecida por la UPEA, enriqueciendo la experiencia académica de los estudiantes con el acceso a recursos como becas y ofertas educativas externas, este proyecto proporcionará oportunidades para que un mayor número de personas acceda a la educación superior, promoviendo el conocimiento.

1.6. METODOLOGÍA

1.6.1. Metodología de Desarrollo

“OpenUP es una metodología de desarrollo de software, basada en RUP (Rational Unified Process), que contiene el conjunto mínimo de prácticas que ayudan a un equipo de desarrollo de software a realizar un producto de calidad, de una forma eficiente” (Ríos, 2013, p. 2), la metodología OpenUP presenta el siguiente ciclo de vida:

- Inicio

- Elaboración
- Construcción
- Transición

1.6.2. Métricas de Calidad

Existen normas que están establecidas para medir las métricas de calidad, estamos hablando de la norma ISO/IEC 25000 el cual también llamadas Square (Requisitos y Evaluación de Calidad de Productos de Software) están conformadas por las normas ISO/IEC 9126 e ISO/IEC 14598, surgen para crear modelos, métricas, procesos y herramientas de evaluación de calidad del software como productos por medio de la especificación de los requisitos (Roa, 2015).

1.6.3. Costo

El modelo por usar es el COCOMO II, en donde “toma en cuenta enfoques más modernos para el desarrollo de software, tales como el desarrollo rápido que usa lenguajes dinámicos, el desarrollo mediante composición de componentes y el uso de la programación de base de datos” (Ian, 2011, p. 637).

1.6.4. Seguridad

Existe una norma que nos dará los pasos necesarios para el proceso de desarrollo del software llamada NB/ISO/IEC – 27001, IBNORCA (2013), menciona que “ha sido preparada para proporcionar los requisitos para establecer, implementar, mantener y mejorar de manera continua un sistema de gestión de la seguridad de la información”.

1.6.5. Pruebas y Software

1.6.5.1. Caja Negra

En Ian (2011, p. 224), nos dice “el sistema se trata como una caja negra cuyo comportamiento sólo puede determinarse por el estudio de entradas y salidas relacionadas”, por lo cual el enfoque estará en ver si el sistema cumple con los requerimientos solicitados.

1.6.5.2. Caja Blanca

“Las pruebas de caja blanca o pruebas estructurales son un enfoque sistemático a las pruebas donde se usa el conocimiento del código fuente del programa para diseñar pruebas de defecto” (Ian, 2011, p. 585), donde su objetivo principal es enfocarse en la estructura interna del software.

1.6.5.3. Pruebas de Stress

Según Myers (2012), menciona que “las pruebas de Stress someten el programa a cargas o tensiones pesadas. Esto no debe confundirse con las pruebas de volumen; un Stress intenso es un volumen máximo de datos, actividad que se produce en un corto periodo de tiempo.”

1.6.5.4. Pruebas de Accesibilidad

En Agesic (2022, p. 4), nos menciona que “cuando nos enfrentamos a un sitio web o aplicación y debemos evaluar su accesibilidad, es importante tener una metodología de evaluación probada que permita descubrir las posibles barreras o errores de accesibilidad que deben ser corregidos”, los pasos a seguir según la

metodología debemos son los siguientes:

- Determinar nivel de conformidad a evaluar
- Elegir las páginas de muestra
- Realizar una evaluación manual

1.7. HERRAMIENTAS

- NextJs: En NextJs (2023), nos dice que “Next.js es un Framework de React flexible que nos da bloques de construcción para crear aplicaciones web rápidas”.
- PostgreSQL: “PostgreSQL es un sistema de gestión de bases de datos orientado a objetos y relacional (ORDBMS) que hace énfasis en la extensibilidad y conformidad con los estándares” (Domínguez, 2019, p. 11).
- Typescript: En el libro de Rozentals (2017), menciona que “TypeScript is both a language and a set of tools to generate JavaScript. It was designed by Anders Hejlsberg at Microsoft (the designer of C#), and is an open source project to help developers write enterprise-scale JavaScript.” [TypeScript es a la vez un lenguaje y un conjunto de herramientas para generar JavaScript. Fue diseñado por Anders Hejlsberg en Microsoft (el diseñador de C#), y es un proyecto de código abierto para ayudar los desarrolladores escribir JavaScript a escala empresarial].

1.8. LÍMITES Y ALCANCES

1.8.1. Alcances

El proyecto provee ciertos alcances que vienen representados en los siguientes

módulos:

- El módulo de inicio de sesión permitirá que el sistema pueda tener accesos controlados en todo momento.
- La implementación del módulo de información que contendrá los convenios, personal, las publicaciones, noticias y ofertas que serán brindados por esta unidad.
- La interfaz que vendrá en un módulo, en donde nos mostrará toda la información que se tiene sobre la unidad y el trabajo que realizan.
- El uso de módulos estadísticos informativos a cerca del alcance e impacto que generan las publicaciones con las personas.
- La creación del módulo de digitalización de archivos, que permitirá automatizar la documentación manual que se viene elaborando.

1.8.2. Límites

El proyecto se centrará exclusivamente en la Universidad Pública de El Alto (UPEA) y su Unidad de Relaciones Internacionales, sin extenderse a otras instituciones o universidades en La Paz o en Bolivia en general.

Además, es importante tener en cuenta que los recursos financieros disponibles pueden limitar la capacidad del proyecto para implementar características adicionales o llevar a cabo expansiones significativas del proyecto, toda la infraestructura tecnológica requerida será establecida en la UPEA, y, por ende, esta requerirá acceso a internet para que pueda ser visualizada por las personas interesadas.

Por último, la efectividad del proyecto dependerá en gran medida de la capacitación adecuada del personal encargado de su operación por lo que la falta de capacitación o de recursos para esta formación podría tener un impacto en la adopción efectiva del sistema.

1.9. APORTES

El presente proyecto, más allá de beneficiar a la Unidad encargada con la administración de información que circula dentro de la misma, ayudará a mantener informados con nuevas convocatorias provenientes de becas y convenios a todos los universitarios en general que entren a revisar la plataforma una vez publicada cada convocatoria, permitiendo así, tener un número mayor de interesados con dichas publicaciones.

CAPÍTULO II: MARCO TEÓRICO

**INGENIERÍA
DE SISTEMAS**
UNIVERSIDAD PÚBLICA DE EL ALTO



2. MARCO TEÓRICO

2.1. INTRODUCCIÓN

El capítulo nos brindará la teoría y conceptos acerca de las herramientas de desarrollo utilizadas en el presente proyecto, nos adentraremos con la información extraída de autores de libros digitales disponibles en documentos e información de la web para que se puedan aclarar las dudas en ciertos puntos que se mencionan en nuestro proyecto.

2.2. CONVENIOS

Los convenios son “manifestaciones de voluntad en las que las partes establecen compromisos generales o específicos de cooperación mutua, para desarrollar en forma planificada becas de interés y beneficio recíproco.” (UTPL, 2015, p. 15), por lo que podemos describirlo un cumplimiento entre dos o más partes donde quedan con obligaciones según su acuerdo y conveniencia. Existen convenios específicos para ciertas áreas, pero en nuestra mencionaremos aquellas que tienen el ámbito académico y cooperación institucional.

2.2.1. Convenio Marco

En VRAC (2024), menciona que un convenio marco tiene la “voluntad de cooperar, desde directrices amplias, con el fin de lograr colaboración recíproca orientada a la promoción conjunta de proyectos en los campos académico, científico, investigativo, de extensión, proyección social entre otros.”, al ser un marco aplicado a

cubrir puntos importantes entre el cumplimiento de alianzas entre universidades e instituciones locales o del exterior podemos dividirlo entre estas categorías.

2.2.1.1. Convenios de Investigación

Permite la colaboración entre instituciones para desarrollar proyectos científicos, accesos de recursos compartidos y la publicación de resultados para generar avances.

2.2.1.2. Convenios de Movilidad Académica

Facilitan el intercambio estudiantil, docentes o investigadores para obtener la experiencia académica y cultural y así promover la internacionalización.

2.2.1.3. Convenios de Formación y Capacitación

Se establecen programas conjuntos, como ser diplomados, maestrías o talleres para fortalecer las competencias de estudiantes universitarios o personas afines a la universidad.

2.2.1.4. Convenios de Aplicación Profesional

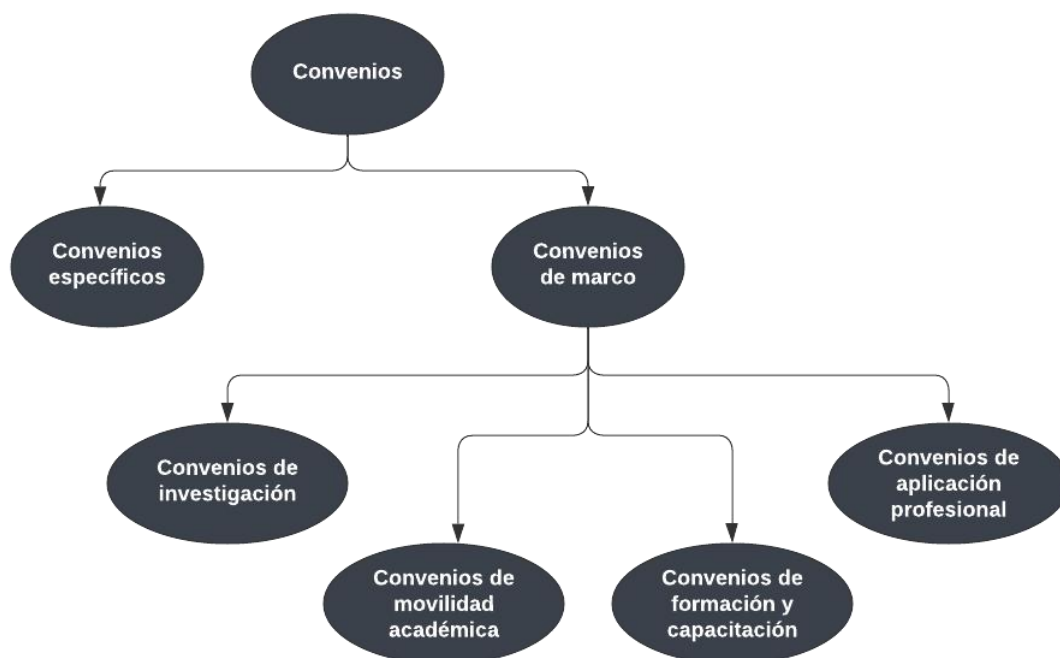
Establecen alianzas para que estudiantes y egresados participen en actividades prácticas dentro de instituciones o empresas, así permitirles aplicar sus conocimientos teóricos en un entorno real para así desarrollar soluciones concretas y fortalecer su experiencia profesional mediante pasantías o proyectos colaborativos.

2.2.2. Convenio Específico

“Es aquel que abre posibilidades de cooperación específicas entre las

instituciones u organismos participantes, en este se determinan las áreas de conocimiento, actividades concretas, obligaciones y compromisos financieros de las partes, así como las dependencias académicas que se encargarán de su realización” (VRAC, 2024).

Figura 2
Tipos de Convenios



2.3. BECAS

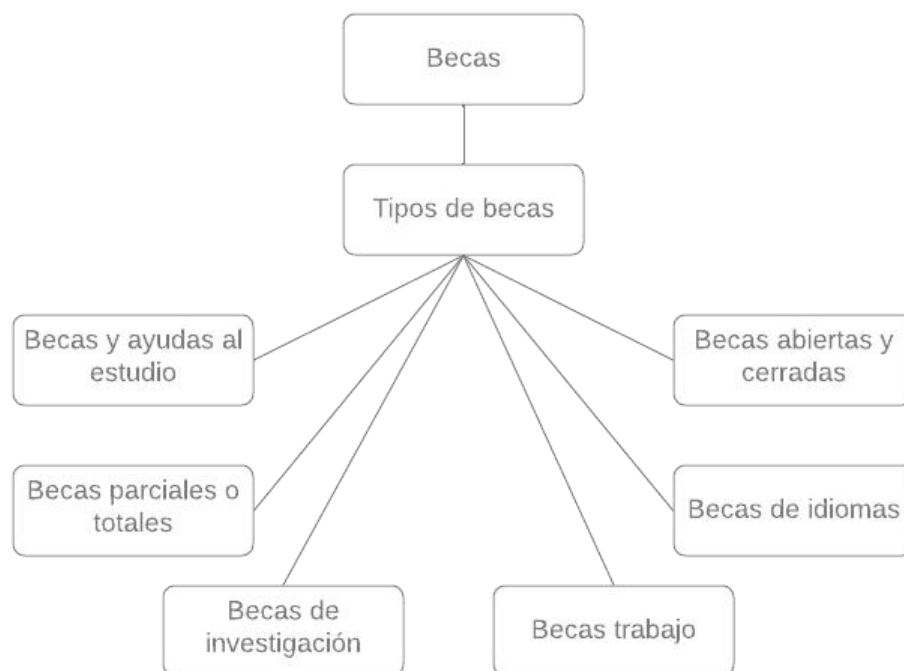
Una beca “es una subvención para cursar estudios superiores, académicos o técnicos, universitarios de postgrado y/o para realizar investigaciones conducentes a la obtención de un título universitario, estudios técnicos y/o para el desarrollo profesional” (CIDI, 2013, p. 2), en donde sirven de ayuda económica que se brindan a instituciones y que los estudiantes puedan pagar parcialmente o de manera total un

programa de formación.

2.3.1. Tipo de Becas

Las becas pueden ser otorgadas a personas que cumplan con ciertas características a base del criterio de la institución, así también, ver los méritos y el monto económico del postulante, por lo que pueden deberse diversos factores que varían según la beca y la institución ofertante las cuales se muestran a continuación.

Figura 3
Tipos de Becas



2.3.1.1. Becas y Ayudas al Estudio

Según Castro (2015, p. 1), “pueden incluir deducciones en las tasas académicas, exenciones totales del importe de la matrícula, pequeñas bolsas de dinero para cubrir los gastos de libros y material didáctica, transporte, alojamiento y manutención,

seguros, etc”.

2.3.1.2. Becas Parciales o Totales

Según Castro (2015, p. 1), nos dice que “se dividen en parciales o totales, pudiendo cubrir únicamente los gastos de los estudios (en el caso de becas de estudio) o también gastos de hospedaje, estadía, pasaje, seguro médico y hasta clases de idiomas”.

2.3.1.3. Becas de Investigación

“Se conceden a estudiantes interesados en realizar proyectos o trabajos de investigación sobre cualquier tema educativo, científico, técnico o artístico. La forma más habitual de clasificar estas becas es por el nivel de estudios requerido para acceder a ellas” (Castro, 2015, p. 1).

2.3.1.4. Becas Abiertas

En Castro (2015, p. 1), menciona que “este tipo de becas universitarias es el candidato quien propone el tema a desarrollar, debiendo presentar una memoria detallada del proyecto de investigación.

2.3.1.5. Becas Cerradas

“Son becas para la realización de proyectos de investigación concretos, desarrollados por equipos de centros de investigación, universidades, centros de desarrollo tecnológico, etc” (Castro, 2015, p. 1).

2.3.1.6. Becas de Trabajo

Según Castro (2015, p. 2), permiten realizar prácticas profesionales o empleo para jóvenes. Se otorgan a estudiantes de los últimos cursos y licenciados universitarios que desean completar su formación, adquirir experiencia a cambios de una retribución económica, generalmente modesta. Existe una modalidad en que el estudiante puede obtener el reconocimiento de su tiempo de trabajo en el exterior con créditos académicos.

2.3.1.7. Becas para aprender Idiomas

Según Castro (2015, p. 2), habla que “se otorgan a estudiantes que desean aprender un idioma fuera de su país de origen”.

2.4. PASANTÍAS

Según Cabezas (1998, p. 1), nos describen una pasantía como “la actividad curricular que los estudiantes de los dos últimos ciclos realizan al incorporarse a una servicio, un proyecto de investigación, o un programa de extensión”, la pasantía puede entenderse como un periodo para que los estudiantes que están finalizando sus estudios académicos y que puedan realizar sus prácticas profesionales.

2.5. EVENTOS ACADÉMICOS

En Castelblanco (2011), nos dice que “los eventos académicos son el conjunto de actividades de carácter profesional, artístico o técnico que implica algún tipo de enseñanza o capacitación”, además nos divide en ciertos tipos tipos de eventos académicos.

2.5.1. Congreso

Tipo de evento donde se hace una confrontación entre un grupo de personas de la misma especialidad y/o interés. De esta manera haciendo un intercambio de experiencias, opiniones, toma de decisiones, resolver problemas, averiguar hechos, identificas situaciones, planear o inspirar algo (Castelblanco, 2011).

2.5.2. Convención

“Evento donde se exponen diferentes situaciones de un tema general y/o específico, y además se llega a una decisión o propuesta” (Castelblanco, 2011).

2.5.3. Seminario

Castelblanco (2011), nos dice que tiene “naturaleza técnica y académica cuyo objetivo es realizar un estudio profundo de determinadas materias con un tratamiento que requiere una interactividad entre los especialistas. En este evento académico se expone lo más reciente de temas generales con conferencias previamente seleccionados”.

2.5.4. Foro

“Evento donde un grupo en su totalidad discute un tema, hecho o problema, conducido por un coordinador. Generalmente constituye la parte final de simposios o mesas redondas” (Castelblanco, 2011).

2.6. SISTEMA

Un sistema puede tener diferentes conceptos basados en distintos autores de

diferentes ramas, pero existe una que nos detalla mejor lo que un sistema representa en donde “se identifican los sistema como conjuntos de elementos que guardan estrechas relaciones entre sí, que mantiene al sistema directo o indirectamente unido de modo más o menos estable y cuyo comportamiento global persigue, normalmente, algún tipo de objetivo” (Cathalifaud, 1998, p. 3).

2.7. SISTEMA INTEGRAL DE INFORMACIÓN

De acuerdo a Gaytan (2016, p. 1), menciona que un sistema integral de información (SII), es un sistema de información que hace uso intensivo y extensivo de las Tecnologías de la Información y las Comunicaciones (TIC) para integrar o centralizar la gestión de la información dentro de una organización. Un SII soporta todos los procesos de negocio y de soporte de la organización, por lo que es un sistema que incluye en conjunto todas las soluciones de información de una organización en uno solo.

2.8. TYPESCRIPT

TypeScript es un lenguaje de programación moderno que permite crear aplicaciones web robustas en JavaScript. TypeScript no requiere de ningún tipo de plugin, puesto que lo que hace es generar código JavaScript que se ejecuta en cualquier navegador, plataforma o sistema operativo (Valverde, 2016, p. 4), creado por el ingeniero de Microsoft llamado Anders Hejlsberg en 2010 y lanzado para el público en 2010. TypeScript es un medio en el cual nos brinda el tipado de objetos por parte de los desarrolladores, lo cual lo hace bastante intuitivo y fácil de entender a la hora de introducir código. Finalmente es pasado al lenguaje de JavaScript o

ES2016, podemos decir que TypeScript es el compilador que nos traduce instrucciones de un lenguaje a otro y que además hace que el código sea mucho más manejable para las futuras personas que vayan acceder a ella.

2.9. ARQUITECTURA REST

Más conocido como (Representational State Transfer), nos dicen que “es un modelo de arquitectura de software que consta de un conjunto coordinado de restricciones arquitectónicas aplicadas a los componentes, conectores y datos, dentro de un sistema hipermedia distribuido” (Rios, 2014), centrarse más en las funciones que realizan cada uno de los componentes, las restricciones y la manera en la que se interpretan los datos.

2.10. REACT

En Molero (2021), hablan de que React es una biblioteca de JavaScript creada para crear interfaces de usuarios rápida e interactivas para aplicaciones web y móviles. Es una biblioteca front de código abierto, basada en componentes, responsable solo de la capa de visualización de la aplicación. En la arquitectura Model View Controller (MVC), la capa de vista es responsable de cómo se ve y se siente la aplicación, fue creada por los desarrolladores de Facebook en 2011, cuando los desarrolladores de la plataforma vieron que los anuncios incrementaran decidieron que React sería un buen aliado para hacer que el rendimiento y las características de las aplicaciones vayan mejorando.

2.11. NEXTJS

“NextJs es un framework JavaScript ligero y de código abierto creado sobre React, que permite desarrollar aplicaciones y sitios web muy rápidos y fáciles de usar.” (Aplyca, 2022). Creado por Guillermo Rauch y lanzado en 2016 y teniendo alrededor de 3000 colaboradores en el mundo.

2.11.1. Principales Características de NextJs

NextJs tiene características que hace que sea bastante llamativa a la hora de elegir un framework que permita un buen desarrollo de aplicaciones.

2.11.1.1. Recarga en Caliente (Hot Code Reloading)

Cuando se desarrolla en el entorno de NextJs y este detecta cambios automáticamente se recarga la página actual.

2.11.1.2. Enrutamiento Automático

NextJs trabaja con un enrutamiento automático mediante el uso de carpetas dentro la carpeta principal denominada por defecto “app”, No se necesita configuración extra cuando se necesite crear rutas de acceso.

2.11.1.3. Renderizado del Lado del Servidor

React de por sí es un librería que solo es utilizado para renderizar en el lado del cliente, en NextJs se puede tomar como opción el utilizarlo tanto del lado del cliente como en el servidor, por lo que se pueden crear los componentes deseados antes de ser creados desde el HTML enviado al cliente.

2.11.1.4. Separación Automática de Código

Al contrario de generar toda la compilación en un solo archivo JavaScript, NextJs hace que toda la aplicación construida se divida en bundles separados de código compilado. Al entrar a una ruta, sólo se carga el código JavaScript necesario para la página en específico.

2.11.1.5. Componentes Dinámicos

Los componentes dinámicos sirven para que NextJs pueda cargarlos de manera dinámica haciendo que los recursos que se consumen en el primer renderizado no sean por consecuencia de estos componentes, por lo que ayuda a que la página cargue mucho más rápido.

2.11.1.6. Compatibilidad con TypeScript

NextJs cuenta con el soporte de TypeScript, el cual permite que el código sea mucho más accesible y comprensible para los futuros desarrolladores que manejen los proyectos creados con el Framework.

2.12. NEXTAUTH

NextAuth.js es una solución de autenticación para Next.js. Esta solución ha evolucionado más allá de su objetivo inicial de proporcionar autenticación exclusivamente para Next.js y actualmente está en medio de un rebranding estratégico de NextAuth.js. Esto significa que está en proceso de convertirse en una biblioteca de autenticación que puede soportar una variedad de frameworks JavaScript (Vessuri, 2024). Fue creado y lanzado en 2016 por Iain Collins NextAuth

cuenta con una variedad de proveedores de autenticación el cual hace que sea viable a la hora de crear proyectos además de conexión con muchas bases de datos y ORMs. Tiene soportes principales como ser:

- Apple
- Auth0
- Azure Active Directory
- Discord
- Dropbox
- Email
- Facebook
- GitHub
- GitLab
- Google
- Instagram
- LinkedIn
- Mailchimp
- Medium
- Netlify
- Pinterest

- Reddit
- Spotify
- Twitch
- Twitter
- Wikimedia
- Zoom

2.13. JWT (JSON WEB TOKEN)

JSON Web Token (JWT) es un estándar abierto (RFC7519) que define un modo compacto y autocontenido para transmitir información segura entre distintas partes utilizando objetos JSON. Una de las características principales de JWT es que esta información puede ser verificada porque es firmada digitalmente. JWT puede ser firmado usando algoritmos de clave-hash (hmac) o claves publico/privada usando RSA (ADIF, s.f.), fue propuesto para un estándar abierto por IETF mediante el formato JSON. El cliente almacena el token JWT en el almacenamiento de navegador o en una cookie para después ser utilizado como una medida de seguridad para la comprobación de sesiones dentro de los sistemas.

2.14. EXPRESSJS

Express es un framework, escrito en JavaScript y alojado dentro del entorno de ejecución NodeJS. Express está diseñado para simplificar el desarrollo de aplicaciones web y APIs en Node.js al proporcionar una capa de abstracción sobre el servidor HTTP incorporado de Node.js, lo que facilita la creación de servidores web y

la gestión de rutas, solicitudes, respuestas (García Grandes, s.f). Fue creado por T.J. Holowaychuk lanzando la primera versión en 2010.

2.15. PRISMA

Prisma es un kit de herramientas (toolkit) de base de datos de código abierto. Reemplaza los ORM tradicionales y facilita el acceso a la base de datos con un generador de consultas o query builder automáticamente generado y type-safe que se adapta a nuestro esquema de base de datos (Adhemar, 2020), prisma es un lenguaje de definición de esquema (SDL) que nos permite hacer migraciones y generar código a partir del esquema creado.

2.16. BCrypt

Bcrypt es una herramienta que es utilizada para hashear contraseñas fue creado por Niels Provos y David Mazieres en su primera publicación en 1999 según Skysnag (2023), “Bcrypt es una función de hash de contraseñas y derivación de claves basada en el cifrado Blowfish”.

2.17. POSTGRESQL

En (Domínguez, 2019, p. 11), menciona que es un sistema de gestión de bases de datos orientado a objetos y relacional (ORDBMS) que hace énfasis en la extensibilidad y conformidad con los estándares. Está liberado bajo la licencia free/open source PostgreSQL, similar a la licencia MIT. PostgreSQL está desarrollado por el Grupo de Desarrollo Global de PostgreSQL, el cual consiste en un grupo de voluntarios empleados y supervisados por empresas como Red Hat y

EnterpriseDB.

2.18. SASS

Sass es un procesador CSS. Un preprocesador CSS es una herramienta que nos permite generar, de manera automática, hojas de estilo, añadiéndoles características que no tiene CSS, y que son propias de los lenguajes de programación, como puede ser variables, funciones, selectores anidados, herencia, etcétera (Pérez, 2019), fue creado en 2006 por Natalie Weizenbaum y diseñado por Hampton Catlin.

2.19. AXIOS

Axios fue desarrollado por Jim VandeHei y lanzada el 2017, “axios es una librería JavaScript que puede ejecutarse en el navegador y que nos permite hacer sencillas las operaciones como cliente HTTP, por lo que podremos configurar y realizar solicitudes a un servidor y recibiremos respuestas fáciles de procesar” (arsys, s.f.).

2.20. RSUITE

Es una librería de código abierto, creado por Simon Guo que está respaldada por un grupo de desarrolladores en GitHub, según GeeksforGeeks (2024), “React Suite es un librería front-end popular con un conjunto de componentes React que están diseñados para la plataforma intermedia y los productos back-end”.

2.21. MATERIAL UI

Es una librería que fue lanzada por Google y que está basada en Material

Design con el objetivo de crear diseños profesionales para aplicaciones web, según González (2024), “Material UI es una biblioteca de componentes de interfaz de usuario de React que implementa directrices de diseño de Material Design”.

2.22. FRAMEWORK WEB

El concepto de framework se emplea en muchos ámbitos del desarrollo de sistemas software, no solo en el ámbito de aplicaciones Web. Podemos encontrar frameworks para el desarrollo de aplicaciones médicas, de visión por computador, para el desarrollo de juegos, y para cualquier ámbito que pueda ocurrirsenos (Gutiérrez, s.f.), también se le denomina un “marco de trabajo”, en donde nos genera un aplicación configurable al cual podemos añadirle piezas para crear aplicaciones completas.

2.23. METODOLOGÍA OPEN UP

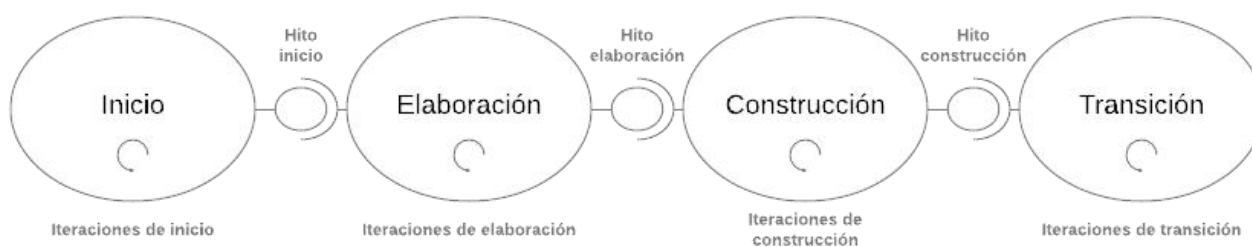
OpenUP es una metodología de Proceso Unificado que aplica enfoque iterativos e incrementales dentro de un ciclo de vida estructurado, utiliza una filosofía ágil que se enfoca en la naturaleza de colaboración para el desarrollo de software, basada en RUP (Rational Unified Process), que contiene el conjunto mínimo de prácticas que ayudan a un equipo de desarrollo de software a realizar un productos de alta calidad, de una forma eficiente (Hernández Barrios, 2014), OpenUp tiene las siguientes características:

- Desarrollo incremental
- Uso de casos de uso y escenarios

- Manejo de riesgos
- Diseño basado en la arquitectura

OpenUP cuenta con ciclo de vida que comprende desde: Inicio, elaboración, construcción, transición.

Figura 4
Fases de la Metodología OpenUP



Nota. Fases de la metodología OpenUP, Jromero, 2009, fuente: Open Up

2.23.1. Fase de Inicio

“En esta fase las necesidades de cada participante del proyecto son tomadas en cuenta y plasmadas en objetivos del proyecto.” (Ríos, 2013, p. 3), en esta fase se definen, el ámbito, límites, requerimientos importantes y un boceto de planificación.

2.23.2. Fase de Elaboración

En esta fase se realizan tareas de análisis del dominio y definición de la arquitectura del sistema. Se debe elaborar un plan de proyecto, estableciendo unos requisitos y una arquitectura estables. Por otro lado, el proceso de desarrollo, las herramientas, la infraestructura utilizar y el entorno de desarrollo también se especifican en detalle en esta fase. Al final de la fase se debe tener un definición

clara y precisa de los casos de uso, los actores, la arquitectura del sistema y un prototipo ejecutable de la misma (Ríos, 2013, p. 3).

2.23.3. Fase de Construcción

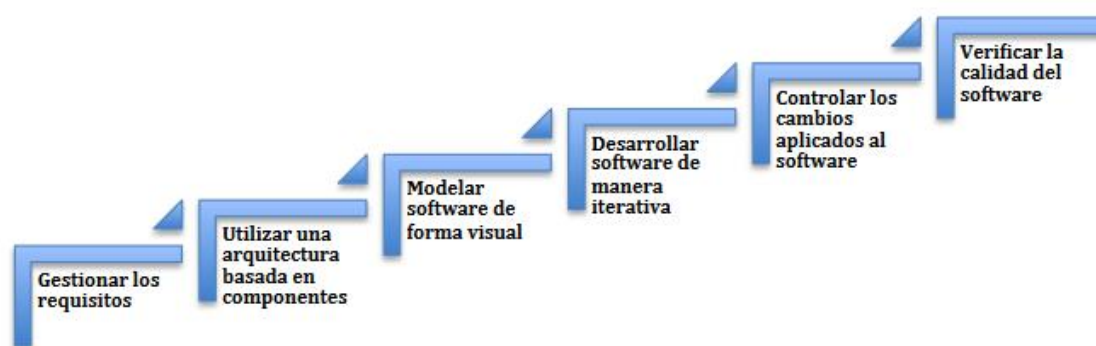
Todos los componentes y funcionalidades del sistema que falten por implementar son realizados, probados e integrados en esta fase. Los resultados obtenidos en forma de incrementos ejecutables deben ser desarrollados de la forma más rápida posible sin dejar de lado la calidad de lo desarrollado (Ríos, 2013, p. 3).

2.23.4. Fase de Transición

Esta fase corresponde a la introducción del producto en la comunidad de usuarios, cuando el producto está lo suficientemente maduro. La fase de la transición consta de las subfases de pruebas de versiones beta, pilotaje y capacitación de los usuarios finales y de los encargados del mantenimiento del sistema. En función de la respuesta obtenida por los usuarios puede ser necesario realizar cambios en las entregas finales o implementar alguna funcionalidad más. (Ríos, 2013, p. 3).

Figura 5

Pasos para la Fase de Elaboración



Nota. Pasos establecidos para cada fase de la metodología Open Up, Jromero, 2009, fuente: Open Up

2.24. Norma ISO/IEC 25000

La norma ISO/IEC 25000 tiene su origen en dos normas ISO/IEC 9126 e ISO/IEC 14598 y se elaboró para dar cobertura a las pymes que se dedican a desarrollar aplicaciones propias y software a medida solicitados por clientes o para uso propio de la empresa. (GrupoACMS, 2022). El cual nos da a entender que va más enfocado a la certificación que avala la calidad de productos software de empresas para comercializarlos o para que se use de manera interna.

ISO/IEC 25000 se compone de 6 familias.

ISO/IEC 25001 encargada de las definiciones de términos usados en la ISO 25000 junto con la SQuaRE.

ISO/IEC 25011 toma el grupo de normas de calidad ya sean externas e internas.

ISO/IEC 25021 nos informa métricas, reglas y definiciones con están sujetas con la definición de la calidad del software.

ISO/IEC 25031 contiene los requisitos de calidad que se emplean en la aparición de un proyecto de software.

ISO/IEC 25041 nos dice sobre los requisitos y las recomendaciones que van para el proceso de evaluación de la calidad de un software.

ISO/IEC 25050 - 25099 nos muestran las variantes de la extensión SQuaRE.

Para evaluar la calidad del software la realizamos en 3 fases las cuales son: evaluación de propiedades del software, desempeño del comportamiento del software en producción, mediciones de la efectividad del software.

Se debe tomar en cuenta 5 puntos importantes a la hora de determinar la calidad del software:

- Funcionalidad
- Eficiencia
- Usabilidad
- Compatibilidad
- Mantenibilidad

Las siguientes ecuaciones nos ayudarán a ponderar un resultado en porcentaje de cada inciso:

Ecuación 1 Funcionalidad del Sistema

$$F = \frac{PF}{PF_{max}} * 100\% \quad (1)$$

En donde:

F = funcionalidad del sistema

PF = Puntos de función obtenidos a partir de los parámetros de medición calculados

PFmax = Puntos de función máximos de la tabla de de parámetros de medición.

Ecuación 2 Eficiencia del Sistema

$$EF = \frac{\sum e_i}{n^2} * 100\% \quad (2)$$

En donde:

EF = Eficiencia del sistema

$\sum e_i$ = sumatoria de los valores de ponderación de eficiencia del sistema

n = Número de preguntas de eficiencia

Ecuación 3 Usabilidad del Sistema

$$U = \left(\frac{\sum x_i}{n} * 100 \right) / 5 \quad (3)$$

En donde:

U = Usabilidad del sistema

$\sum x_i$ = Sumatoria de los parámetros de usabilidad

n = Número de parámetros utilizados en el sistema

Ecuación 4 Mantenibilidad del Sistema

$$Md = \frac{Mt - (Fa + Fc + Fd)}{Mt} * 100\% \quad (4)$$

En donde:

Md = Nivel de madurez del sistema

Mt = Número de módulos totales para la última versión del sistema

Fd = Número de módulos que se eliminan en la actualización

Fc = Número de módulos que se modifican en cada versión

Fa = Numero de módulos nuevos que se añaden a cada versión

Ecuación 5 Portabilidad del Sistema

$$P = \frac{Es}{Mn} * 100\% \quad (5)$$

En donde:

P= Portabilidad del sistema

Es = cantidad de entornos soportados

Mn = cantidad de modificaciones necesarias

2.25. COCOMO II

Modelo COCOMO II, modelo de estimación que se encuentra en la jerarquía de modelos de estimación de software con el nombre COCOMO, por Constructive Cost Model (Modelo Constructivo de Coste). El modelo COCOMO original se ha convertido en uno de los modelos de estimación de coste del software más utilizados y estudiados en la industria. (Miss, 2020, p. 2).

2.25.1. Ecuaciones de Estimación

Las ecuaciones de estimación nos ayudan a medir el esfuerzo generado para la creación de un software, basado en la cantidad de código y el tipo de modelo seleccionado. Las cuales son:

Tabla 1*Constantes Aplicados a los Tipos de Modelo COCOMO II*

MODELO USADO	CONSTANTES			
	A	B	C	D
Orgánico	2.4	1.05	2.5	0.38
Semi acoplado	3	1.12	2.5	0.35
Empotrado	3.6	1.2	2.5	0.32

Ecuación 6 Esfuerzo Dedicado en el Proyecto

$$E = A * KLDC^B * FAE \quad (6)$$

En donde:

E= Esfuerzo de persona por mes

KLDC = Lineas de código del proyecto calculado en miles.

FAE = Atributos de esfuerzo

A = Coeficiente otorgado por COCOMO II

B = Coeficiente otorgado por COCOMO II

La siguiente ecuación nos sirve para determinar el tiempo de desarrollo que se va a implementar en el sistema.

Ecuación 7 Tiempo de Esfuerzo en Meses

$$T_{dev} = C * E^D, \text{ en meses} \quad (7)$$

En donde:

E= Esfuerzo de persona por mes

Tdev= Tiempo de desarrollado aplicado al proyecto.

C = Coeficiente otorgado por COCOMO II

D = Coeficiente otorgado por COCOMO II

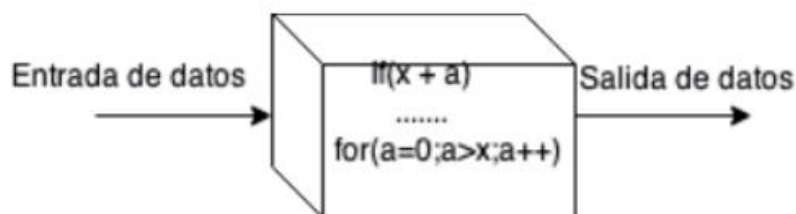
2.26. Pruebas de Stress

En López (2010, p. 1), menciona que esta prueba se utiliza normalmente para romper la aplicación. Se va doblando en número de usuarios que se agregan a la aplicación y se ejecuta un prueba de carga hasta que se rompe. Este tipo de prueba se realiza para determinar la solidez de la aplicación en los momentos de carga extrema.

2.27. Pruebas de Caja Blanca

En Maldonado (2021, p. 14), dice de que es una técnica de diseño de caso de prueba que se usa la estructura de control para obtener los casos de prueba. Dentro de esta estructura de control podemos encontrar la estructura de un componente de software como pueden ser sentencias de decisiones, caminos distintos del código, la estructura de una página web. etc.

Figura 6
Ejemplo de Caja Blanca



Nota. Ejemplo de cómo funciona una prueba de Caja Blanca en un software, Maldonado, 2021, fuente: Pruebas de Software

2.28. Pruebas de Caja Negra

Son las que utilizan el análisis de especificación, tanto funcional como no funcional, sin tener en cuenta la estructura interna del programa para diseñar los casos de pruebas. A diferencia de las pruebas de caja blanca, estas suelen realizarse durante las últimas etapas de las pruebas (Maldonado, 2021, p. 15).

Figura 7
Ejemplo de Caja Negra



Nota. Ejemplo de cómo funciona una prueba de Caja Negra en un Software, Maldonado, 2021, fuente: Pruebas de Software

2.29. Pruebas de Accesibilidad

En Agesic (2022, p. 3), menciona que “la accesibilidad se debe tener en cuenta durante todo el ciclo de vida de los diferentes productos digitales, tanto en su

creación, como en sus nuevas versiones”.

Las etapas que se deben tomar en cuenta son las siguientes:

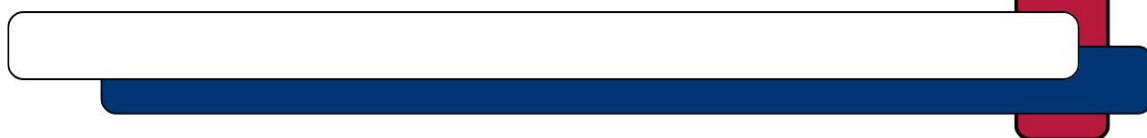
- Planificación: se define las herramientas y la metodología a usar y así asegurar el cumplimiento de los requisitos.
- Diseño y desarrollo: se corrobora que la solución cumpla con los requisitos de accesibilidad
- Testing: el producto debe cumplir con los requisitos.
- Gestión de contenido: al crear o editar contenido, debe existir una verificación, previo al guardado de la publicación.

La siguiente metodología brinda instrucciones sobre los controles de accesibilidad que deben realizarse. Los pasos para seguir son los siguientes:

- Paso 1: Determinar nivel de conformidad a evaluar.
- Paso 2: Elegir las páginas de muestra.
- Paso 3: Evaluar con herramientas automáticas.
- Paso 4: Realizar evaluación manual.

CAPÍTULO III: MARCO APLICATIVO

**INGENIERÍA
DE SISTEMAS**
UNIVERSIDAD PÚBLICA DE EL ALTO



3. MARCO APLICATIVO

3.1. INTRODUCCIÓN

Este capítulo realizaremos la aplicación de la metodología aplicada y requisitos necesarios que esta nos pide para que vayan acorde a lo mencionado en el capítulo 2, utilizaremos desde el prototipo, casos de uso, modelos de bases de datos y mostrando finalmente los resultados finales como ser secciones acabadas del proyecto.

3.2. METODOLOGÍA APLICADA (OPEN UP)

El proyecto presenta fases iterativas en las cuales son las 4 ya mencionadas, empezaremos con el orden establecido que se propone en Open UP, y llegar al objetivo final del sistema.

3.2.1. Entregables

Los entregables son los resultados que esperamos usando la metodología con OpenUP y que nos brindan una visualización general sobre la manera en la que el proyecto será ejecutado y los hitos esperados.

Tabla 2
Entregables del Proyecto

Fase	Actividad	Documento
Inicio	Comienzo del proyecto	Visualización del sistema

	Planificación del proyecto	Plan de elaboración del proyecto
	Identificación de requerimientos del proyecto	Documento de especificación de requerimientos del software
	Boceto del programa	Documento de boceto del sistema
Elaboración	Desarrollo de la arquitectura	Documento de arquitectura del sistema
	Casos de uso principales	Documento de casos de uso principales del proyecto
	Desarrollo de modelos de datos	Creación de modelos de la base de datos
Construcción	Desarrollo del sistema	Codificación de la solución en base a lo estable en la fase de la elaboración
Transición	Despliegue de la solución	Documento - Manual técnico
		Documento - Manual de usuario

3.2.2. Fase de Inicio

La fase de inicio definiremos los requerimientos, la planificación del proyecto y el boceto del sistema.

3.2.2.1. PLANIFICACIÓN DEL PROYECTO

La planificación presentada a continuación es un detalle general de los tiempos empleados para cada etapa de elaboración del proyecto.

Tabla 3
Planificación del Proyecto

Fase	Iteraciones	Tarea	Inicio	Final
FASE DE INICIO	1ra	Obtención de requerimientos	25-03-2024	26-03-2024
		Definición de requerimientos	27-03-2024	30-03-2024
		Creación del boceto	05-04-2024	10-04-2024
FASE DE ELABORACIÓN	1ra	Análisis de arquitectura del software	11-04-2024	11-04-2024
		Documentación de la arquitectura del software	12-04-2024	18-04-2024
	2da	Análisis de los casos de uso del software	19-04-2024	29-04-2024
		Documentación de los casos de uso del software	30-04-2024	15-05-2024
	3ra	Análisis de datos del sistema	16-05-2024	30-05-2024
FASE DE DESARROLLO	1ra	Desarrollo del sistema	01-06-2024	30-09-2024
FASE DE TRANSICIÓN	1ra	Estudio del manual técnico	01-10-2024	05-10-2024

	Creación del manual técnico	06-10-2024	10-10-2024
2da	Estudio del manual de usuarios	11-10-2024	15-10-2024
	Creación del manual de usuarios	16-10-2024	24-10-2024
3ra	Despliegue a producción	24-10-2024	03-11-2024

3.2.2.2. REQUERIMIENTOS DEFINIDOS

Como fase inicial empezamos con la obtención de requerimientos funcionales y no funcionales del proyecto

Tabla 4
Requisitos Funcionales del Proyecto

Nro.	Descripción
RF-01	Listado de convenios nacionales e internacionales ya sean institucionales o para cada carrera por separado. Se deben realizar las operaciones CRUD desde el administrador.
RF-02	Listado de becas y operaciones CRUD desde el administrador y el listado de becas para visitantes desde una pestaña separada.
RF-03	Listado de eventos en una pestaña separada para los visitantes, y operaciones CRUD desde administrador.
RF-04	El sistema debe tener una visualización de galería similar a una galería tradicional y contar con descripción y título y con operaciones

CRUD desde el administrador.	
RF-05	El sistema de contar con un listado de pasantías para cada carrera o ser institucional y tener las operaciones CRUD desde el lado del administrador.
RF-06	Convenios, pasantías, eventos, becas, noticias, idiomas y galería debe contar con los filtros de listado que se espera que usen los visitantes para buscar información más precisa de lo publicado.
RF-07	El sector de becas debe contar con 3 categorías, las cuales son, becas, noticias e idiomas.
RF-08	El sistema debe brindar información adicional sobre la unidad de relaciones internacionales
RF-09	El sistema debe brindar el registro de carreras e instituciones afiliadas con cada convenio.
RF-10	El sistema debe contar con herramientas que permitan modificar documentos.
RF-11	El sistema debe tener una administración de personal vinculado a un solo usuario.
RF-12	El sistema debe tener la opción de editar la información acerca de la unidad.
RF-13	El sistema debe de contar con la función de registro de participantes para las becas que son idiomas y becas para poder estar al tanto y realizar búsquedas más rápidas de participantes.
RF-14	El sistema debe tener un control de acceso para cada usuario entrante registrando la persona con su usuario
RF-15	El sistema debe tener la capacidad mostrar vídeos relacionados con las publicaciones que se realizan a partir de cualquier medio

RF-16	El sistema debe contar con los acceso directos hacia nuestras redes sociales
RF - 17	El sistema tiene que contar con la capacidad de añadir vídeos y ser alojados en el servidor.
RF - 18	El sistema debe tener la capacidad de exportar la información en archivos fáciles de recuperar.

Tabla 5
Requerimientos no Funcionales del Proyecto

Nro.	Descripción
RNF-01	El sistema debe estar la web y funcionar desde internet
RNF-02	El sistema debe poder ser accesible desde un dispositivo movil
RNF-03	El sistema debe poder albergar archivos pdf y word
RNF-04	El sistema debe presentar de mantenimiento en cierto periodo de tiempo o en cuanto la unidad lo requiera
RNF-05	Páginas externas deben poder relacionarse si así lo requieran con las publicaciones generadas a través de filtros mediante un link
RNF-06	El sistema debe ser claro con la información que muestra.
RNF-07	El sistema debe contar con fuentes visibles a la vista.
RNF-08	El sistema debe tener respuestas rápidas por parte del servidor
RNF-09	El sistema debe ser visible a la luz con colores claros.
RNF-10	El sistema debe contar con colores acordes a la institucion y la propia universidad

3.2.2.3. Boceto del Sistema

El boceto del sistema nos permitirá tener una visualización previa de la arquitectura del sistema en general y de manera simple basado en los requerimientos otorgados por los futuros usuarios, donde dividiremos el sistema en módulos para mostrar cómo estas tiene dicha gestión en subtítulos diferentes para su mayor comprensión. A continuación detallamos bocetos simples de la pantalla inicial, el inicio de sesión y el apartado de contenido multimedia como son los vídeos.

3.2.2.3.1. Boceto Inicial del Sistema

Detallamos aspectos como el inicio de sesión, la pantalla principal del sistema y el lugar de publicación de vídeos, cabe mencionar que los bocetos son ideas iniciales por lo que es un inicio para luego definir más detalle en los módulos que se irán realizando en la fase de construcción.

Figura 8

Boceto de Inicio de la Aplicación

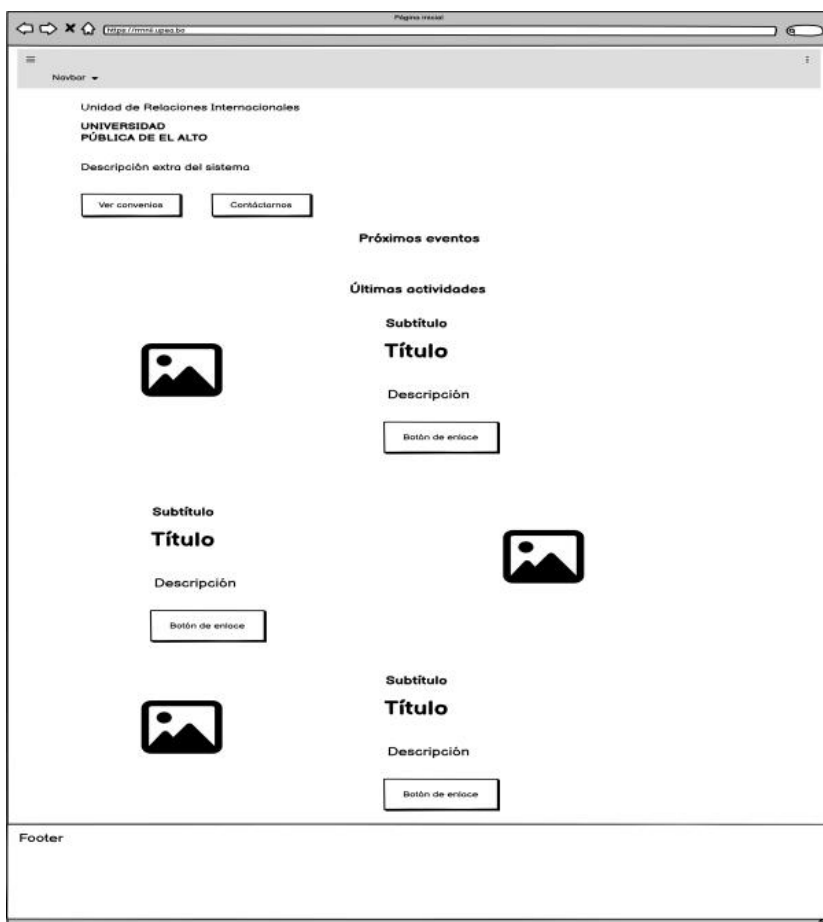


Figura 9
Modal de Inicio de Sesión

MODAL DE INICIO DE SESIÓN

USUARIO

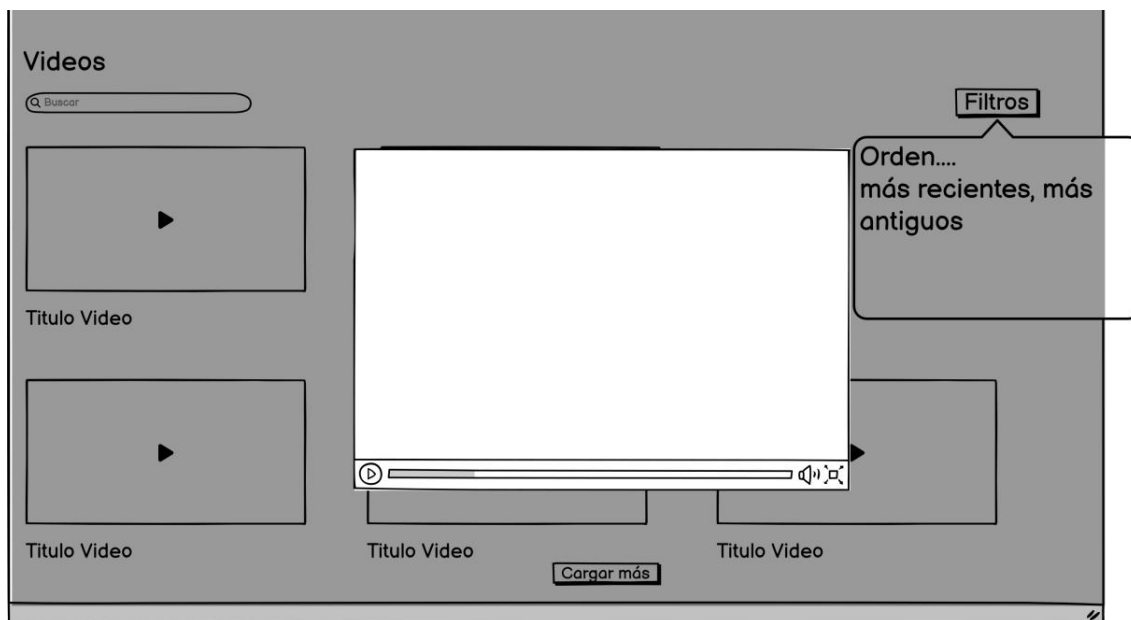
CONTRASEÑA

CAPTCHA DEL SISTEMA

INTRODUCIR CAPTCHA

INICIAR SESIÓN

Figura 10
Lista de Vídeos y Modal Reproductor



3.2.2.3.2. Boceto del Módulo de Convenios

Figura 11
Boceto Convenios Dashboard

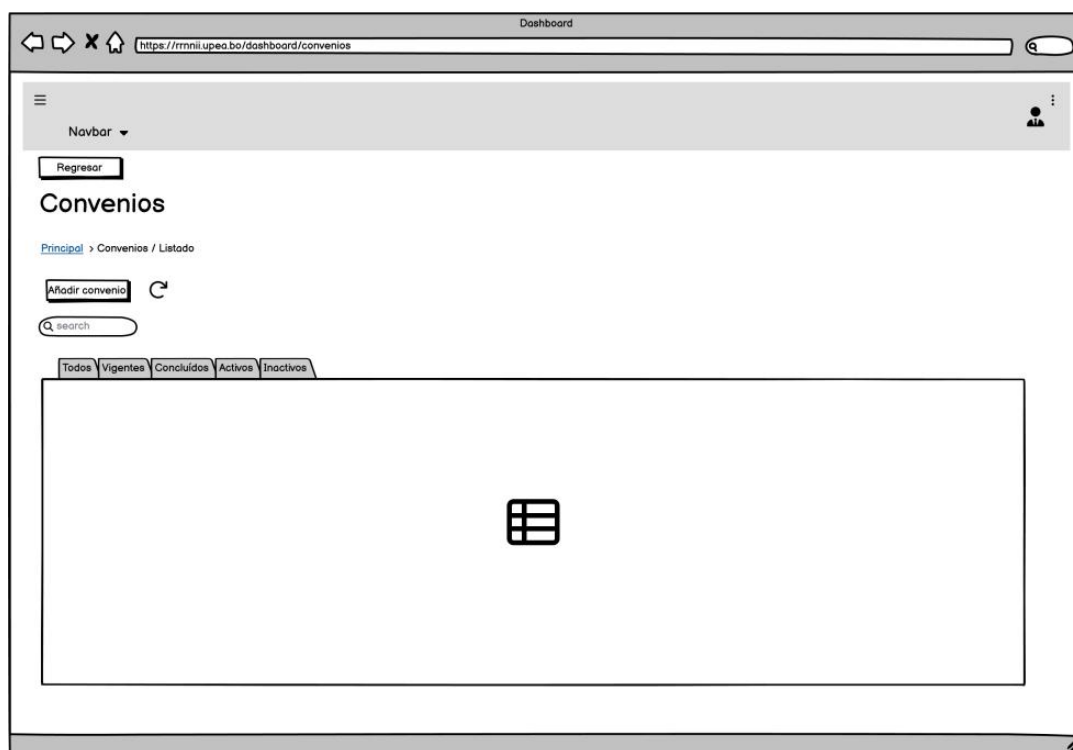


Figura 12
Boceto de Edición de Convenios

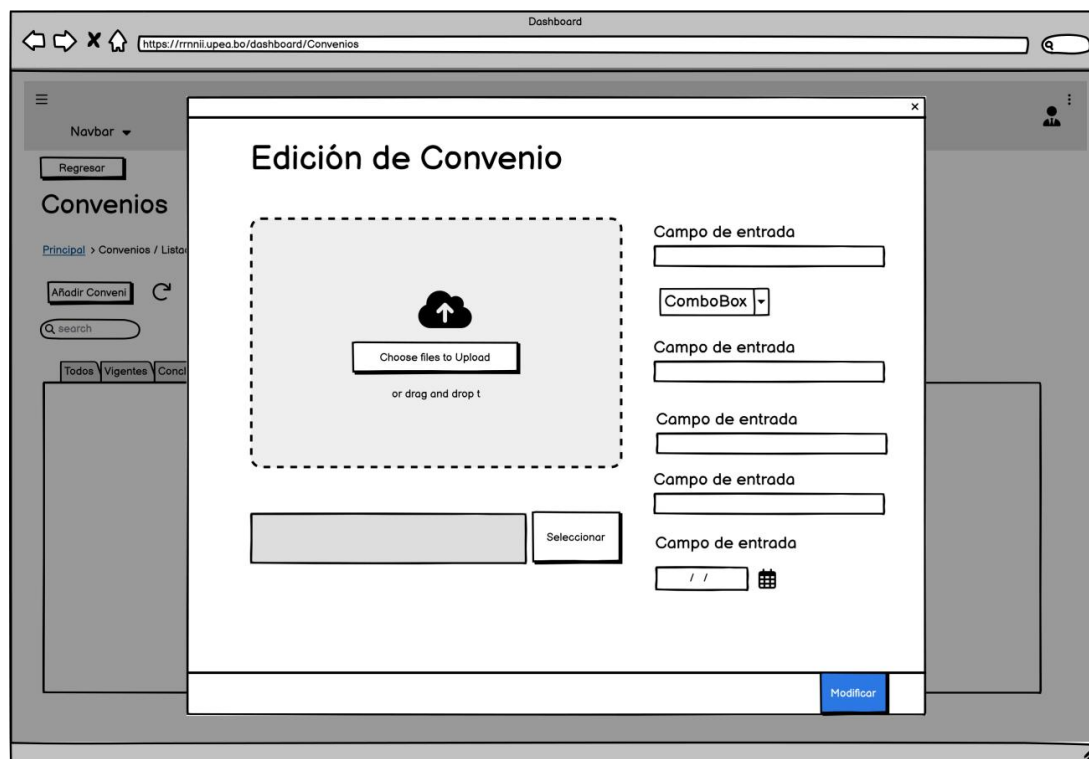
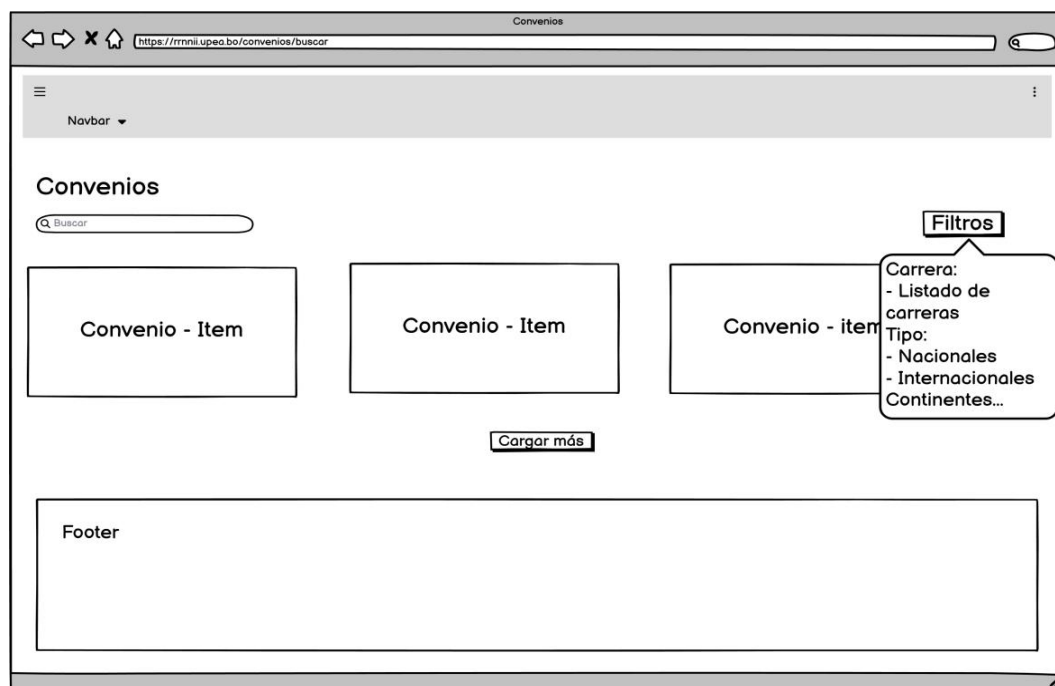


Figura 13
Boceto para el Visitante de Convenios



3.2.2.3.3. Boceto del Módulo de Becas

Figura 14

Boceto para el Visitante de Becas

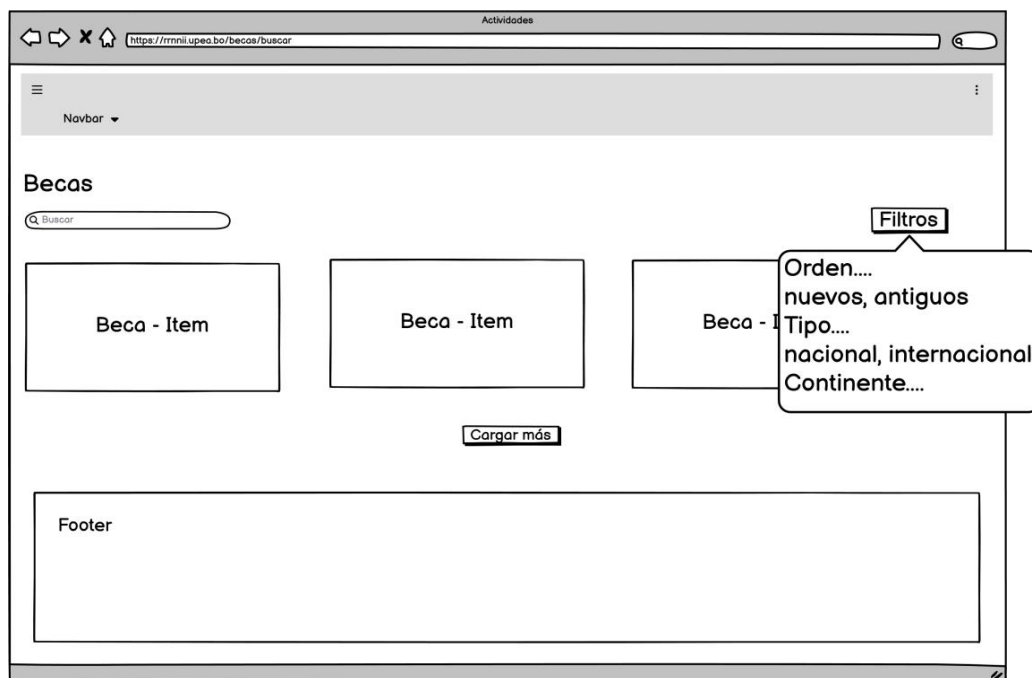
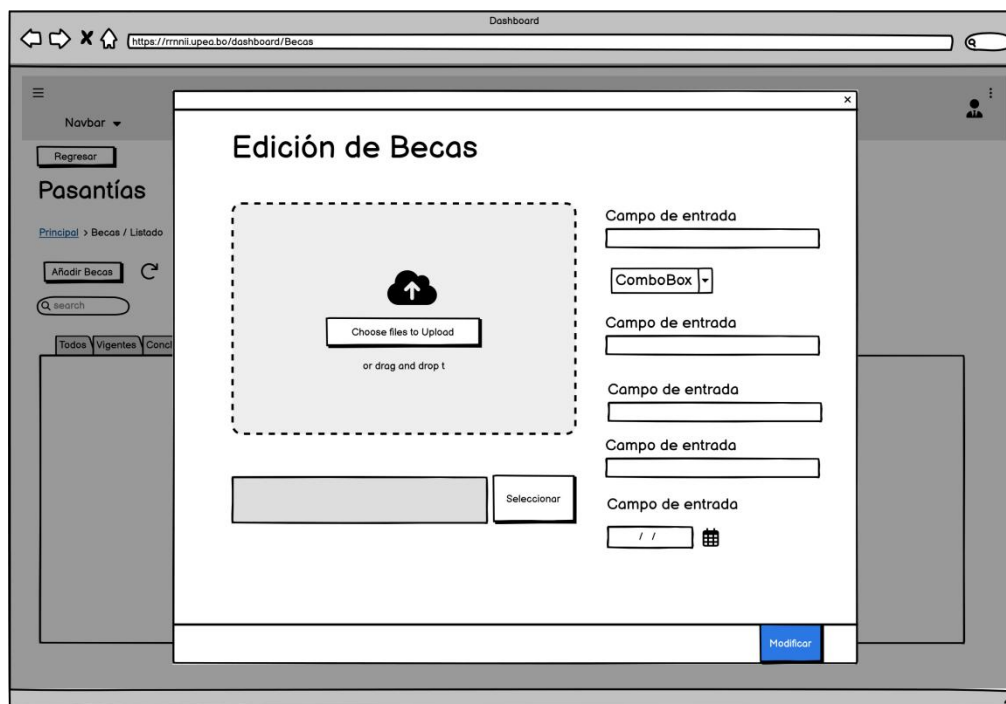


Figura 15

Boceto Becas Dashboard



Figura 16
Boceto Edición de Becas



3.2.2.3.4. Boceto del Módulo de pasantías

Figura 17
Boceto para el Visitante de Pasantías

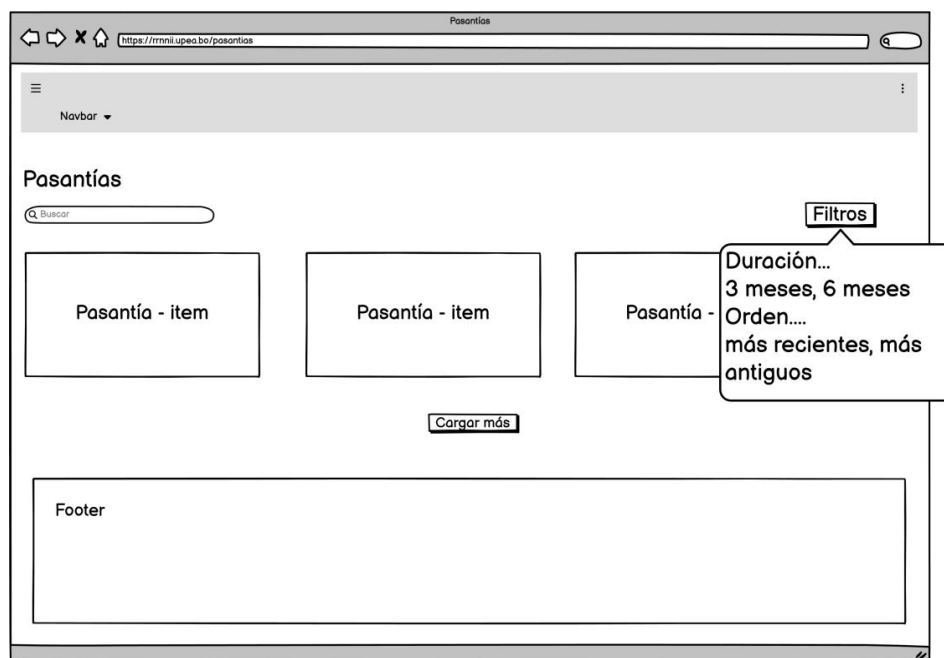


Figura 18
Boceto Pasantías Dashboard

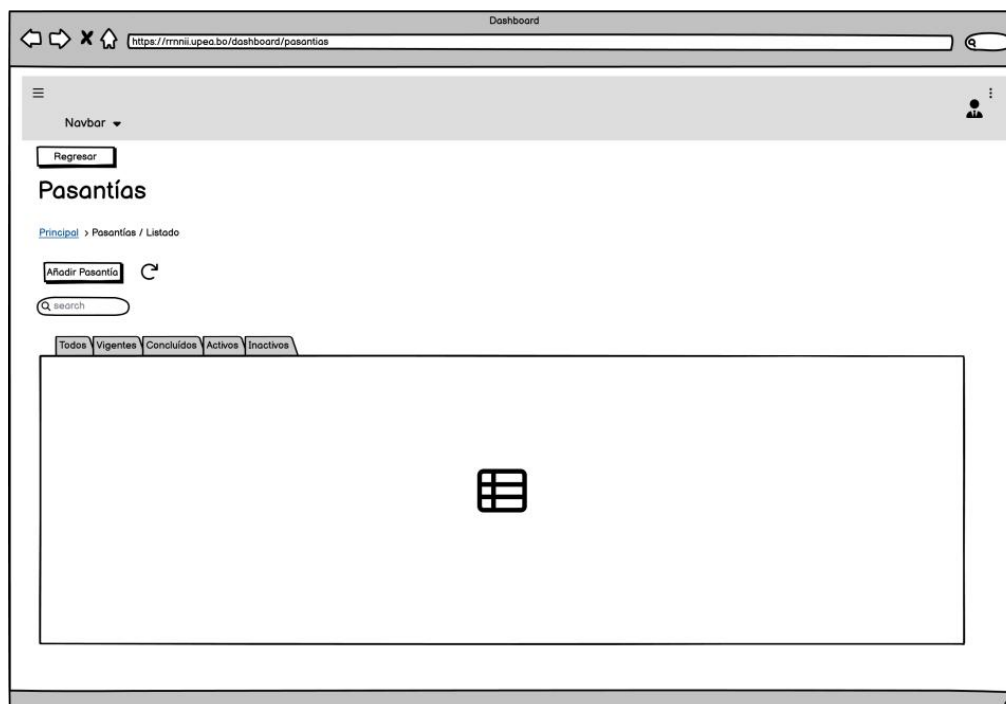
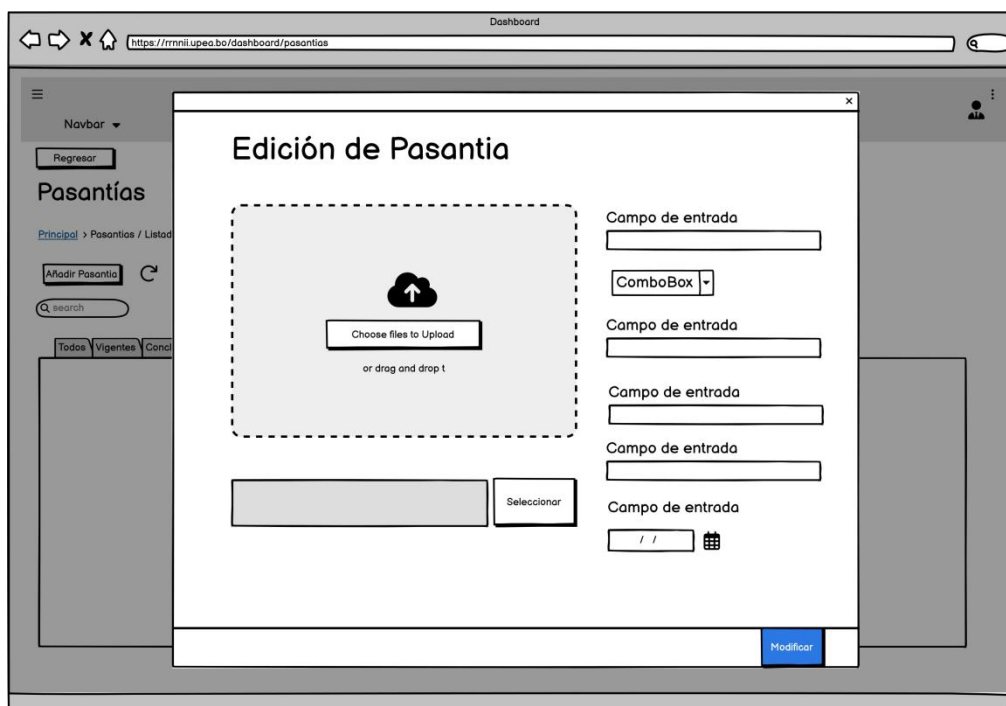


Figura 19
Boceto de Edición de Pasantías



3.2.2.3.5. Boceto del Módulo de Eventos

Figura 20

Boceto de Eventos

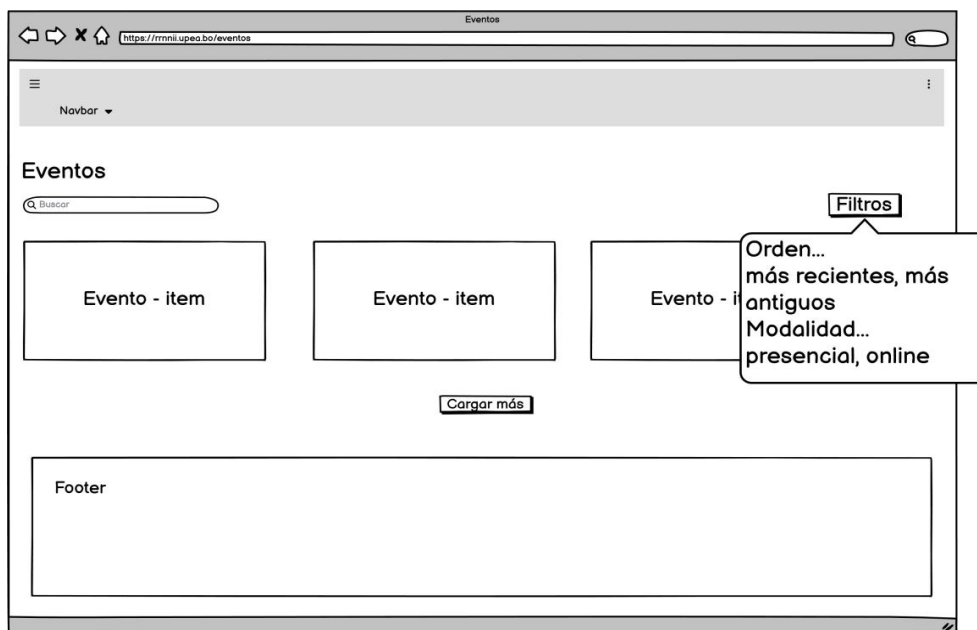


Figura 21

Boceto Eventos Dashboard

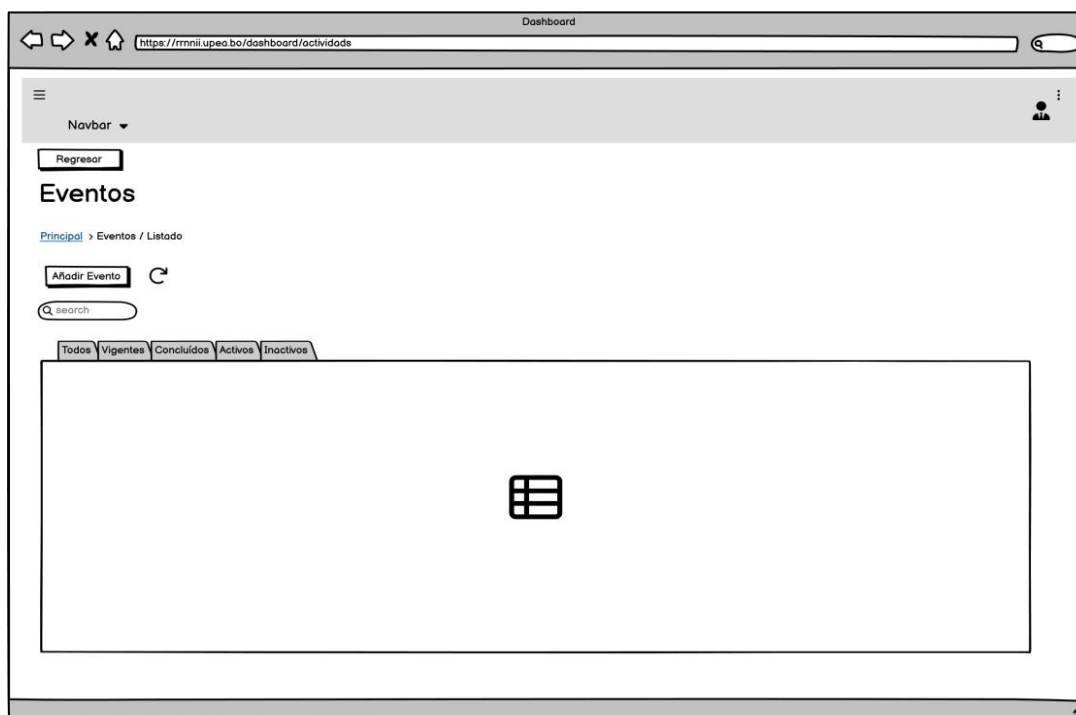


Figura 22
Boceto de Edición de Eventos



3.2.2.3.6. Boceto del Módulo de Noticias

Figura 23
Boceto de Noticias

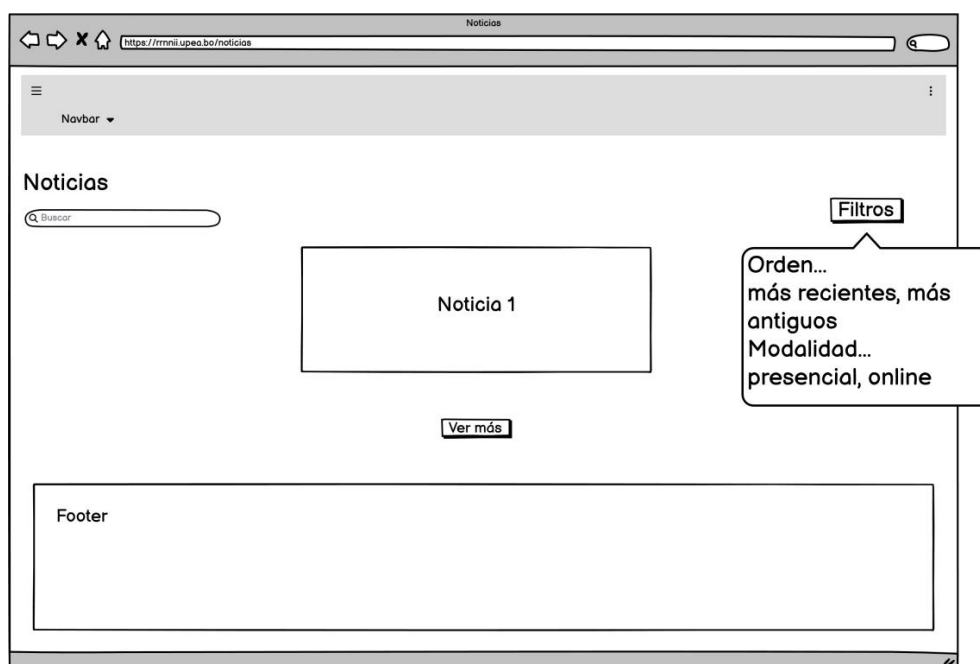


Figura 24
Boceto Noticias Dashboard

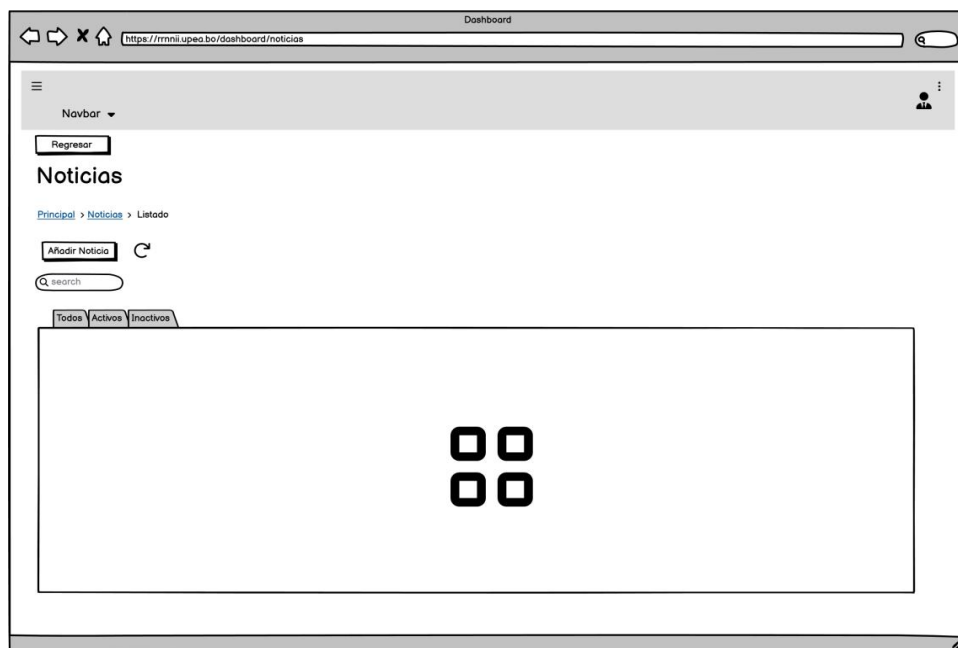
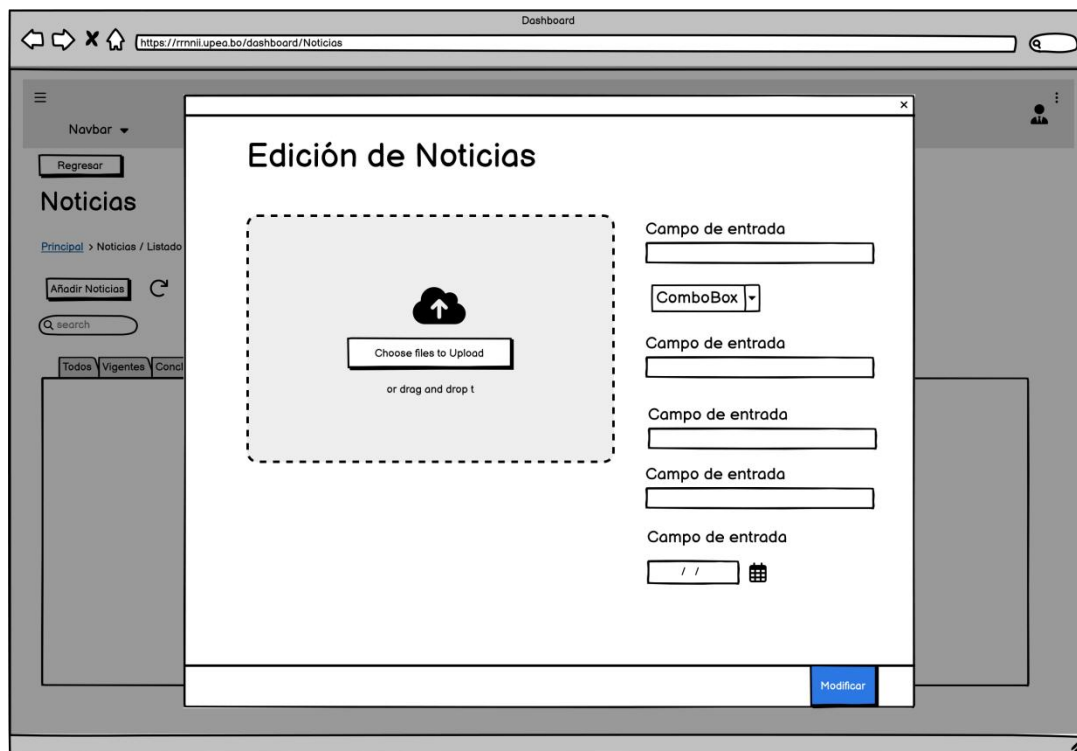


Figura 25
Boceto Edición de Noticias



3.2.3. Fase de Elaboración

En esta fase nos involucraremos en 3 iteraciones principales para la creación de la estructura del sistema, donde la primera tiene como objetivo la arquitectura del software, la segunda iteración la creación de los casos de uso del sistema, la última iteración contiene el modelo relacional.

3.2.3.1. Primera Iteración

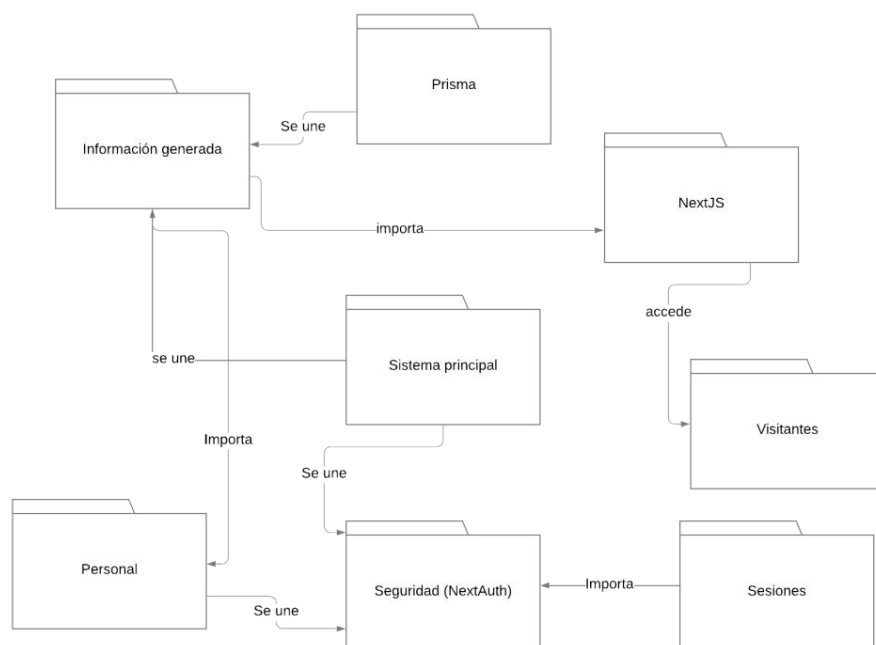
La primera iteración obtenemos 2 estructuras principales para el proyecto como ser: el diagrama de paquetes y el diagrama de subsistemas.

3.2.3.1.1. Diagrama de Paquetes

Nuestro diagrama de paquetes nos representará los aspectos lógicos de nuestro sistema para mostrarnos una vista de alto nivel.

Figura 26

Diagrama de Paquetes del Sistema

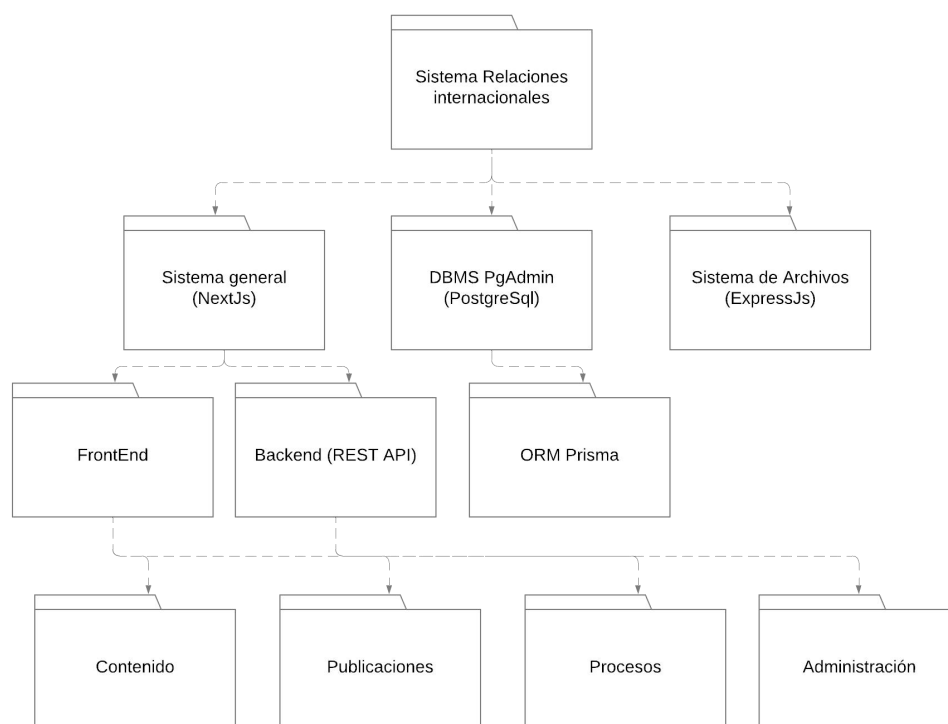


3.2.3.1.2. Diagrama de Subsistemas

El diagrama de subsistemas nos muestra las interacciones o relaciones que existen entre cada subsistema que compone el sistema completo, y el cómo estas trabajan en conjunto con otras para lograr una meta propia que van acorde con el objetivo principal del proyecto.

Figura 27

Diagrama de Subsistemas del Proyecto



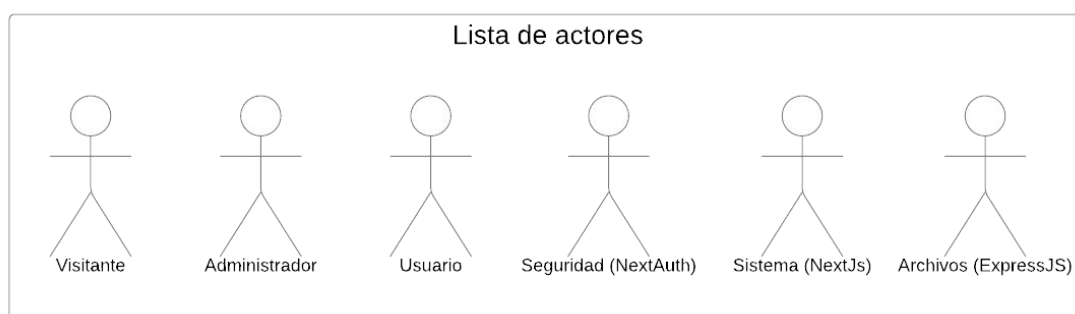
3.2.3.2. Segunda Iteración

La segunda iteración creamos los casos de uso en base a los requerimientos establecidos de la primera fase y el modelo de todas las entidades participantes en un modelo UML.

3.2.3.2.1. Actores

Los actores son quienes interactuarán con el sistema, las cuales son 6: administrador, usuarios, visitantes, seguridad (NextAuth), Sistema (NextJs), Archivos (ExpressJS), cada uno de ellos tiene un papel importante en cuanto al cumplimiento del objetivo del sistema, a continuación se mostrarán los actores en conjunto relacionados con la gestión esperada de cada uno de los módulos ya mencionados anteriormente.

Figura 28
Esquema de Actores del Sistema



3.2.3.2.2. Caso de Uso Visitante

Los visitantes son las personas que están vinculadas con la universidad pueden ser así como los participantes a becas o universitarios, docentes que buscan convenios que pueden beneficiar a los mismos, o visitantes afines a cualquier institución que necesiten tener un mayor conocimiento en el trabajo que realiza la universidad con la Unidad.

Figura 29
Caso de Uso Visitante

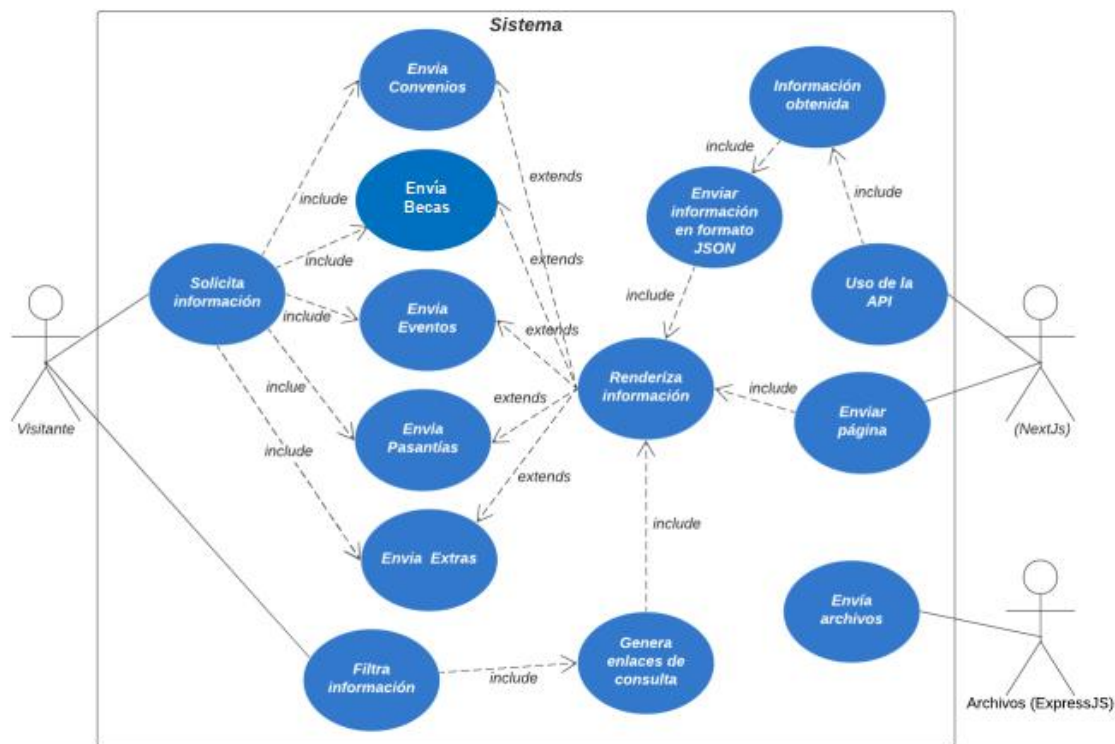


Tabla 6
Operaciones del Visitante en el Sistema

Actor	Acción	Descripción
Visitante	Visita el sistema	Solicita información al sistema acerca de los convenios, becas, eventos, noticias, pasantías e información de la unidad
NextJs	Genera contenido	El sistema es la encargada en el renderizado de la información a través de la solicitud REST a su propia API para generar los archivos estáticos y dinámicos a mostrar

Archivos (ExpressJS)	Envía archivos	Los archivos que se necesitan para mostrar contenido multimedia o la descarga de documentos las realiza el sistema archivos basado en ExpressJS
-------------------------	-------------------	---

3.2.3.2.3. Caso de Uso Administrador

El administrador tiene la capacidad de realizar acciones con un nivel de importancia mayor al que un usuario común y que le permiten tener el control absoluto del sistema desde la publicación de información hasta la vista de acciones realizadas en las tablas por los usuarios o en la propia administración del personal y accesos.

Figura 30
Caso de Uso del Administrador

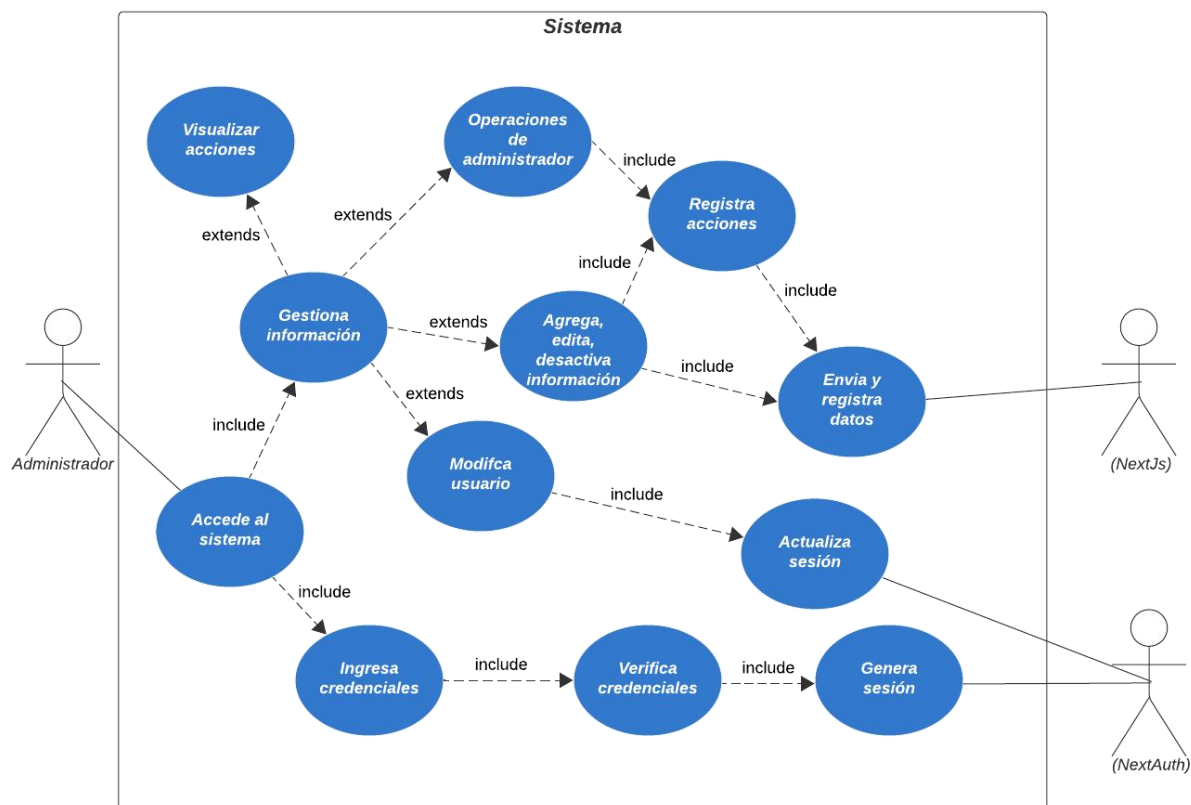


Tabla 7
Operaciones del Administrador en el Sistema

Actor	Acción	Descripción
Administrador	Control de la información	El administrador realiza el control absoluto dentro del sistema, pudiendo realizar todo tipo de acción con la información
NextJs	Administración de los datos	El sistema realiza la tarea de aceptar los cambios que solicita el administrador y es quien las ejecuta
NextAuth	Control de sesiones	Es la encargada de tener las sesiones del usuario actualizadas y permitir el acceso a rutas confidenciales

3.2.3.2.4. Caso de Uso Usuario

Los usuarios son designados al personal y que pertenecen a dos categorías como ser: técnico y secretario(a), tienen permitido diferentes movimientos dentro del sistema, a diferencia del administrador (usuario mayor), se tienen ciertas manipulaciones de información restringidas, como ser: vista de acciones realizadas por los usuarios, administración del personal, acceso a la manipulación completa de la información de la base de datos.

Figura 31
Caso de Uso Usuario

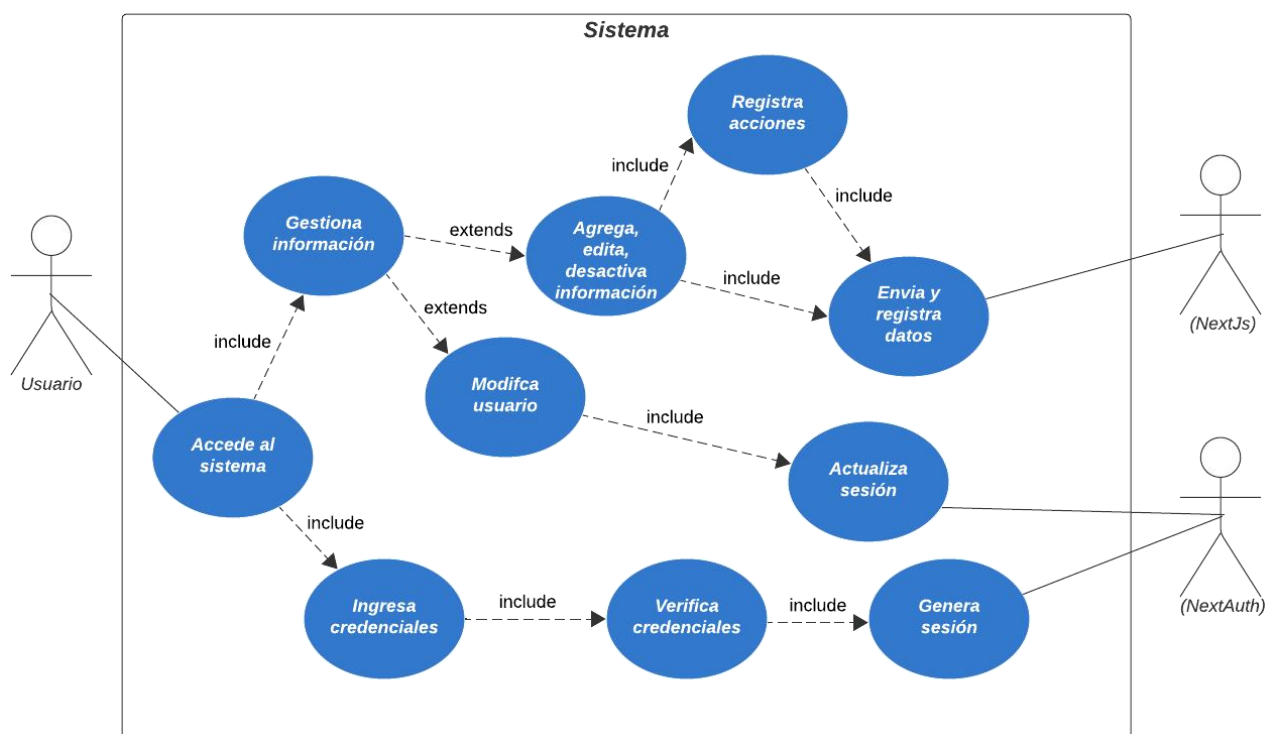


Tabla 8
Operaciones Realizadas por el Usuario

Actor	Acción	Descripción
Usuario	Control a nivel básico de la información	El usuario realiza tareas de gestión de información básicas que son visibles para los visitantes
NextJs	Administración de los datos	El sistema realiza la tarea de aceptar los cambios que solicita el usuario y es quien las ejecuta
NextAuth	Control de sesiones	Es la encargada de tener las sesiones del usuario actualizadas y permitir el acceso a rutas confidenciales a los tipos de usuarios

3.2.3.2.5. Caso de Uso Visitante (Modo Participante)

El participante de beca es un visitante del sistema que no requiere registro

previo. Este usuario puede explorar las becas disponibles y, si encuentra una de interés, ingresar sus datos personales para postularse como candidato. Su interacción con el sistema está limitada exclusivamente a la visualización y llenado del formulario de postulación, sin acceso a funcionalidades administrativas.

Figura 32
Caso de Uso Participante de Beca

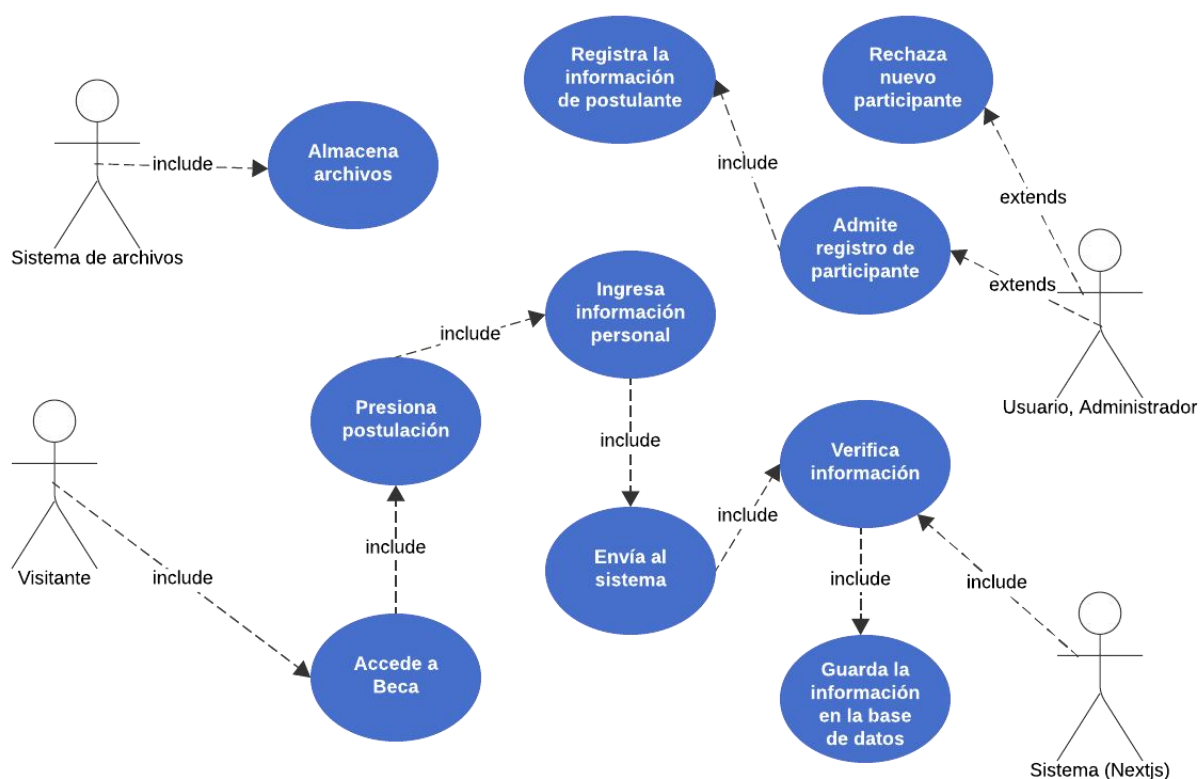


Tabla 9
Operaciones Realizadas por el Nuevo Participante

Actor	Acción	Descripción
Usuario, administrador	Revisión de postulantes	Los usuarios tanto administrador como usuario común pueden revisar las solicitudes de los

 participantes nuevos y guardarlos

Visitante	Interesado en la postulación a becas	El visitantes puede realizar la búsqueda de becas vigentes y solicitar una postulación
NextJs	Procesamiento de postulación a través de formulario	Tiene la tarea de verificar los datos subidos y registrarlos para posteriormente ser visible por parte de los usuarios y el administrador para su revisión
ExpressJS (archivos)	Almacenamiento de archivos	Es el encargado de guardar los archivos que se suben como documentos o imágenes que pueden servir como respaldo

3.2.3.2.6. Caso de Uso Inicio de Sesión

El caso de uso Inicio de Sesión permite que los actores del sistema, como el Administrador y el Usuario con Rol Simple, se autenticuen mediante NextAuth para acceder a funcionalidades según su rol. El sistema garantiza la seguridad de las credenciales mediante protocolos robustos, validando los permisos antes de otorgar acceso a las diferentes secciones de la plataforma.

Figura 33

Caso de Uso de Inicio de Sesión

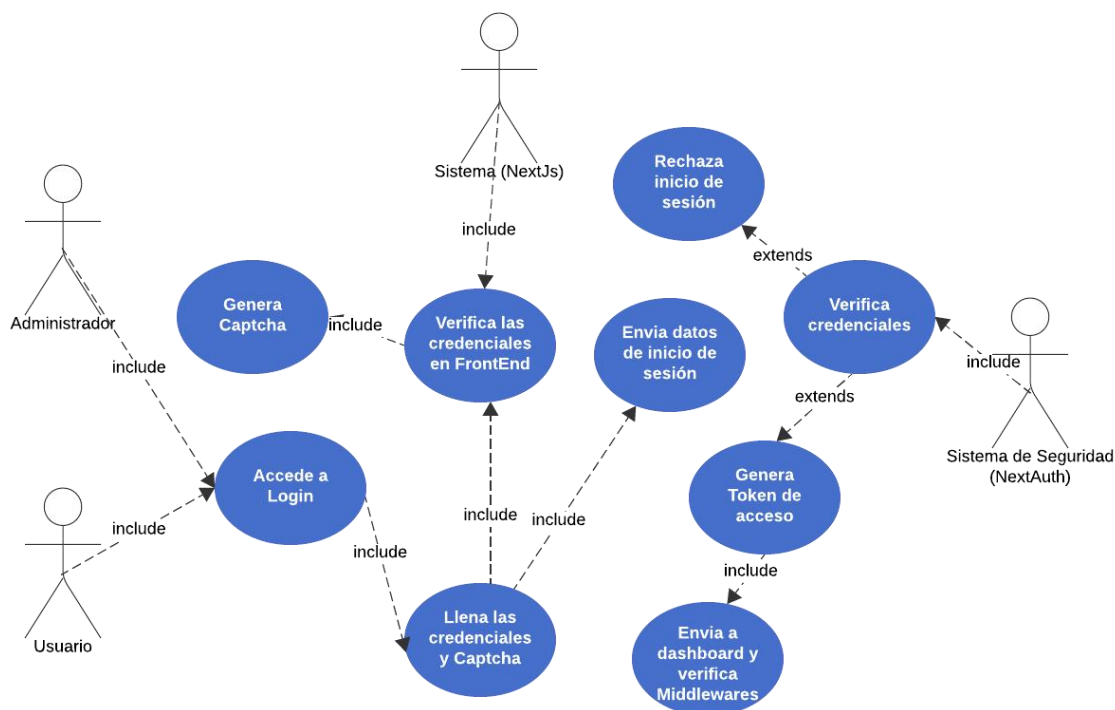


Tabla 10
Inicio de Sesión- Usuario y Administrador

Actor	Acción	Descripción
Usuario, administrador	Inicio de sesión desde el login	Realiza la acción del colocado de sus credenciales en los campos solicitados y la confirmación del Captcha de seguridad
NextAuth	Verificación de las credenciales	Tiene como objetivo la verificación de las credenciales y la redirección del usuario al dashboard en el caso de ser permitido junto a la generación del token JWT
NextJs	Verificación del lado del cliente	Es el encargado de verificar que las credenciales sean colocadas y dar mensajes de alerta en el caso de observaciones en los campos de credenciales.

3.2.3.2.7. Caso de Uso Gestión de Publicaciones

La gestión de publicaciones permite a un usuario autorizado (técnico o secretario) administrar diferentes tipos de publicaciones, convenios, pasantías, noticias, eventos y vídeos. El usuario puede agregar archivos adjuntos (como documentos, imágenes o vídeos) cuando sea necesario. Una vez registrada, la publicación se hace accesible para la comunidad universitaria según el tipo de contenido, permitiendo que los interesados puedan acceder a la información de manera clara y estructurada. Así también la gestión incluye el cambio de estado y la edición de los mismos con el respaldo de seguridad implementada en acciones que son visibles para el administrador. Las publicaciones a excepción de las becas tienen algo en común entre ellas por lo que se puede asemejar al caso de uso de convenios.

Figura 34
Caso de uso Gestión de Convenios

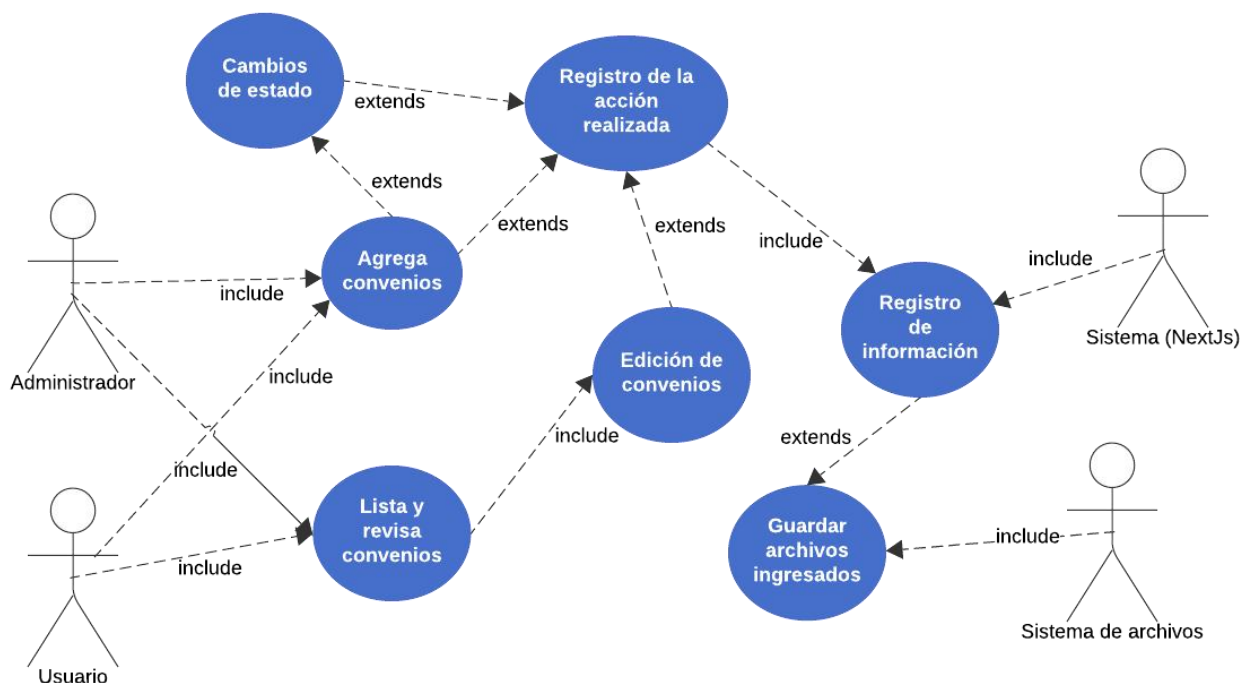


Tabla 11
Gestión de Publicaciones y Convenios

Actor	Acción	Descripción
Usuario, administrador	Gestión de convenios o publicaciones	Gestiona toda la información de los convenios y publicaciones que trabajan de manera similar, el cual concierne en la edición, creación, cambios de estado y listado
ExpressJS (archivos)	Almacenamiento de archivos de tipo documentos o multimedia	El sistema archivos aloja todo tipo de archivos como documentos, multimedia. Como ser imágenes o publicaciones que hacen el uso de vídeos
NextJs	Registro y verificación de datos	El sistema principal se encarga de la validación de datos, e incorporar los mismos en la base de datos

Figura 35
Caso de Uso Gestión de Becas

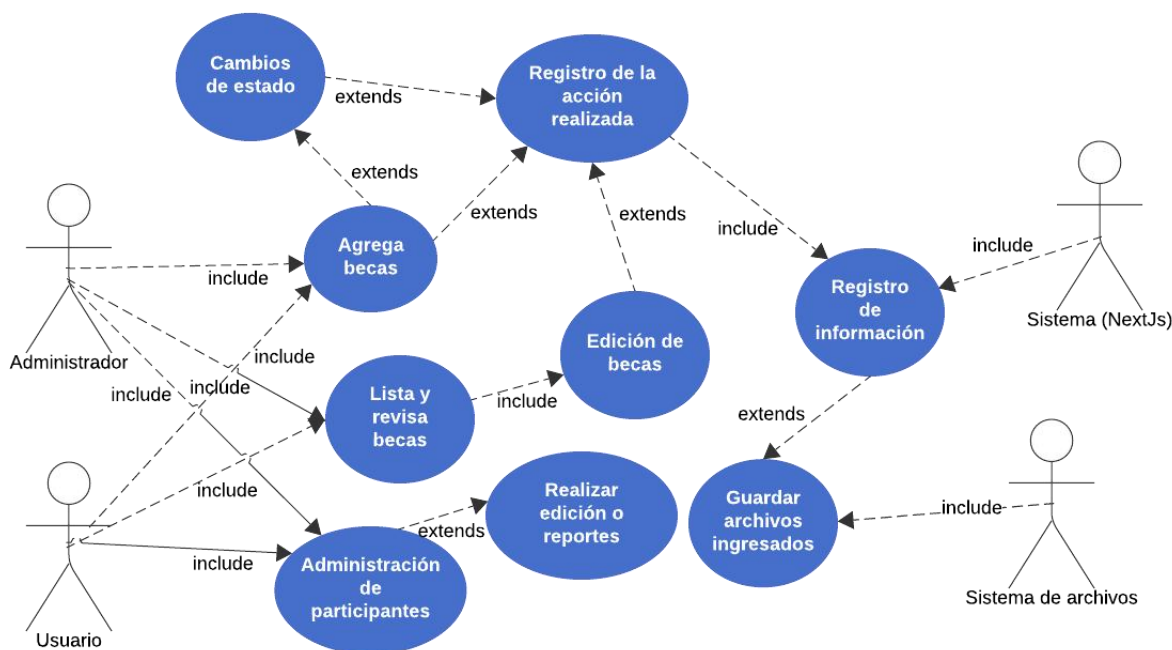


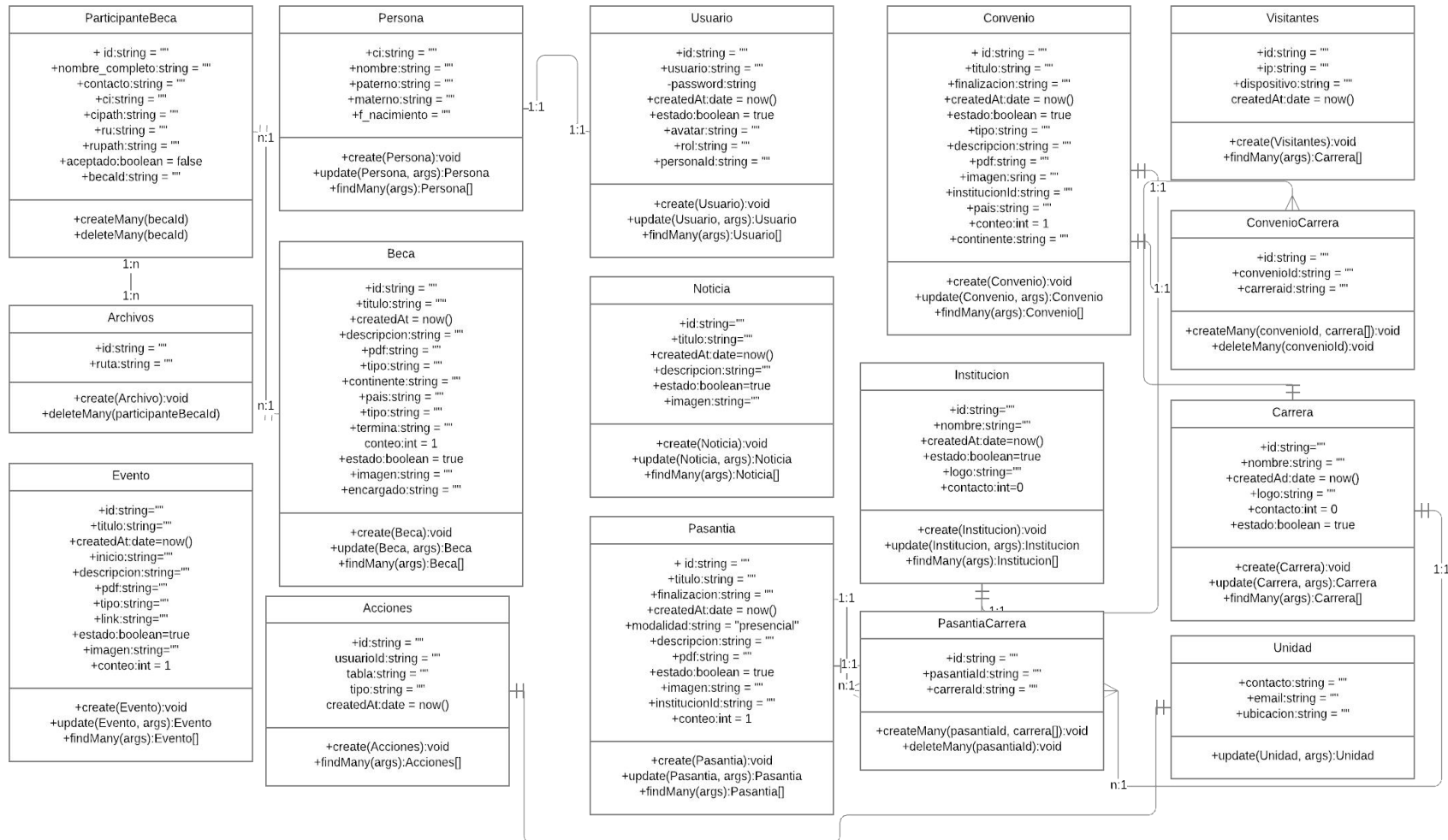
Tabla 12
Gestión de Becas

Actor	Acción	Descripción
Usuario, administrador	Gestión de becas	Gestiona toda la información de las becas, realizan la revisión de nuevos postulantes o interesados en las becas y generar reportes de los mismos, además de la típica administración relacionada en la edición, creación, cambios de estado y listado
ExpressJS (archivos)	Almacenamiento de archivos	El sistema archivos aloja todo tipo de archivos como documentos y multimedia.
NextJs	Registro y verificación de datos	El sistema principal se encarga de la validación de datos, e incorporar los mismos en la base de datos

3.2.3.2.8. Modelo UML

El modelo UML nos da una vista extensa de las entidades participantes junto con los atributos y los métodos vinculados a ellas y que es interpretación de toda la información que se alojará en la base de datos. El modelo UML nos sirve para ver las relaciones que se se tienen entre entidades y el cómo cada una de ellas crean una dependencia de información y el cómo actúan sobre la base de datos a través de funciones asociadas con cada entidad.

Figura 36
Modelo UML



3.2.3.3. Tercera Iteración

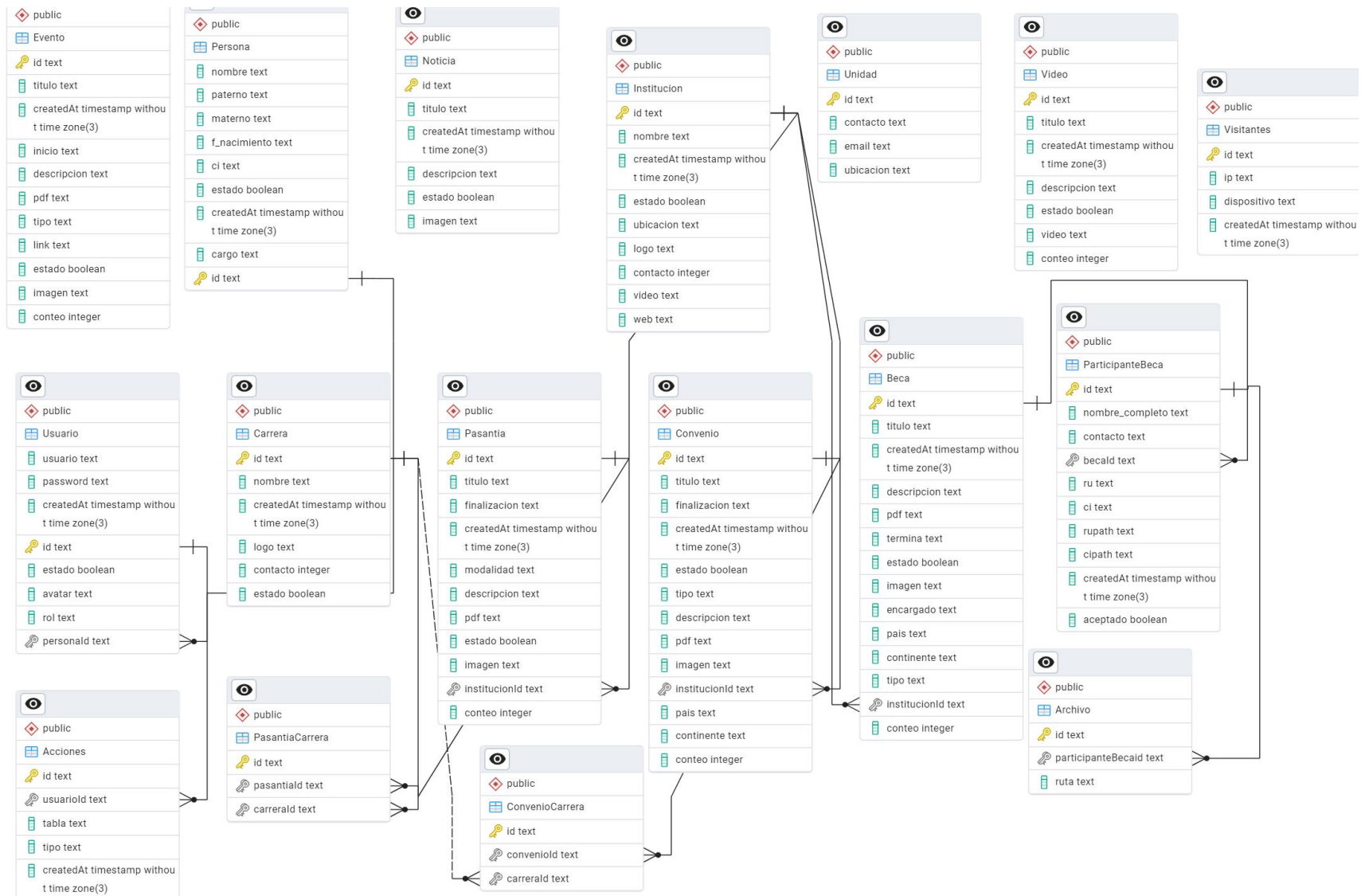
La tercera iteración nos presenta el modelo relacional que se basa en la base de datos implementada.

3.2.3.3.1. Modelo Relacional

El modelo es generado a través del DBMS de PostgreSQL, en este caso generado por PgAdmin4, el cual representa toda la base de datos con el tipo y el nombre de las tablas creadas.

Figura 37

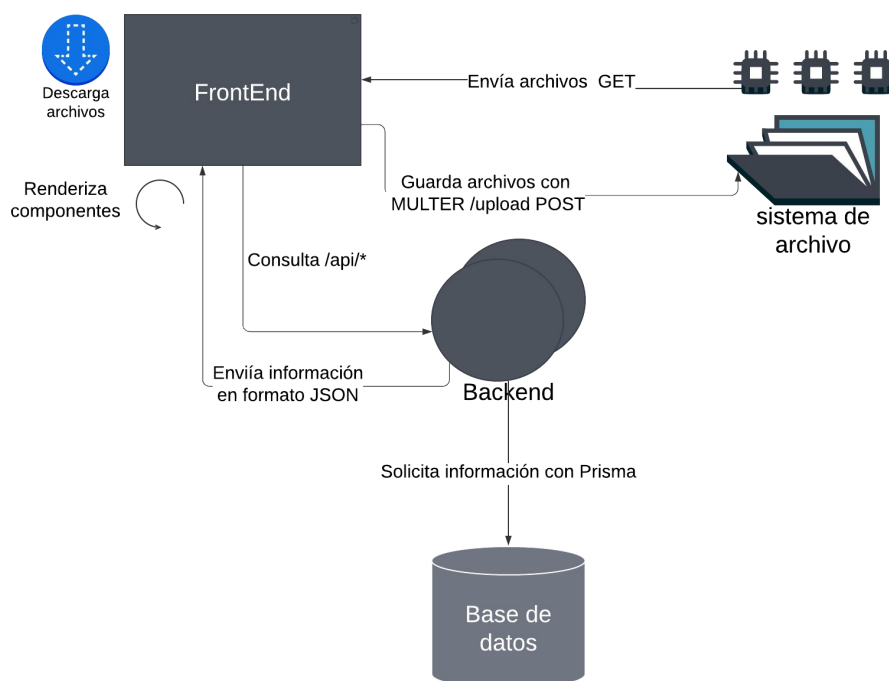
Modelo Relacional de la Base de Datos



3.2.4. Fase de Construcción

La fase de construcción es la fase en la que realizamos la codificación del proyecto, el proyecto en general presenta 2 divisiones, la primera en donde se aloja toda la funcionalidad como ser el frontend y backend que trabajan con NextJs, y la otra el cual está creada en ExpressJS y sirve como almacenamiento de documentos e imágenes que se envían a través de NextJs.

Figura 38
Esquema Funcional del Software

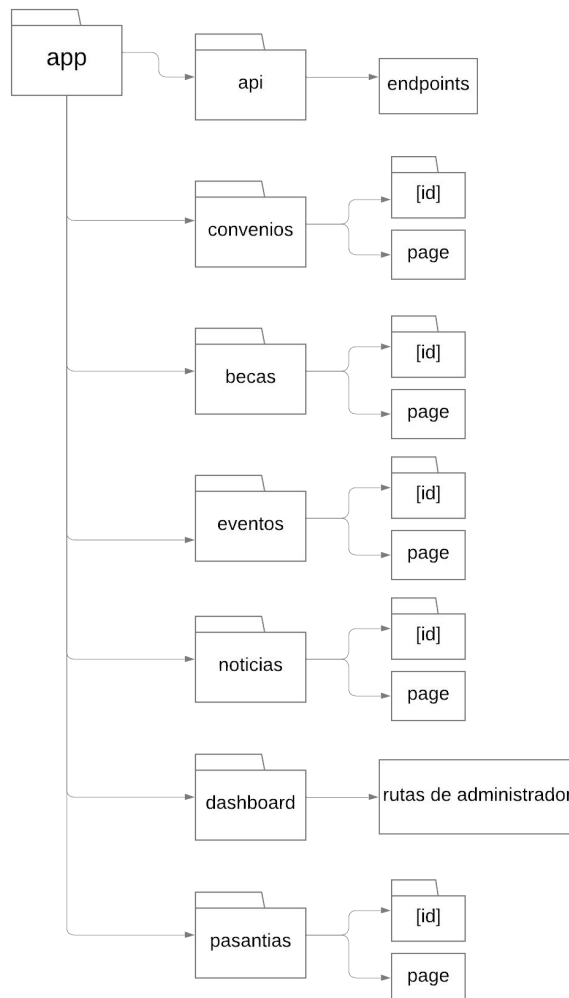


Para el proyecto, se hizo la implementación de ciertas características para definir el diseño del sistema como ser: colores, navegación, animaciones, etc. Una de ellas fue la elección de los colores y el tipo de fuente que se usarán en el sistema a través

de RSuite y la estructura para la muestra de información que se usarán en los convenios, becas, eventos y pasantías.

El proyecto posee cierta estructura base, el cual nos servirá como guía de la composición de la estructura del proyecto y los diferentes archivos principales que lo componen, la siguiente imagen nos muestra la manera las carpetas y archivos principales en la arquitectura del proyecto.

Figura 39
Esquema de la Estructura del Proyecto



Además de existir el espacio para el Frontend y Backend, existe un subsistema extra desarrollada en ExpressJS con el objetivo de almacenar cualquier tipo de archivo que se guarde en el sistema principal desarrollada en NextJs.

Figura 40

Estructura del proyecto en ExpressJS

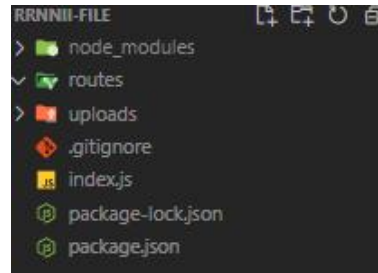
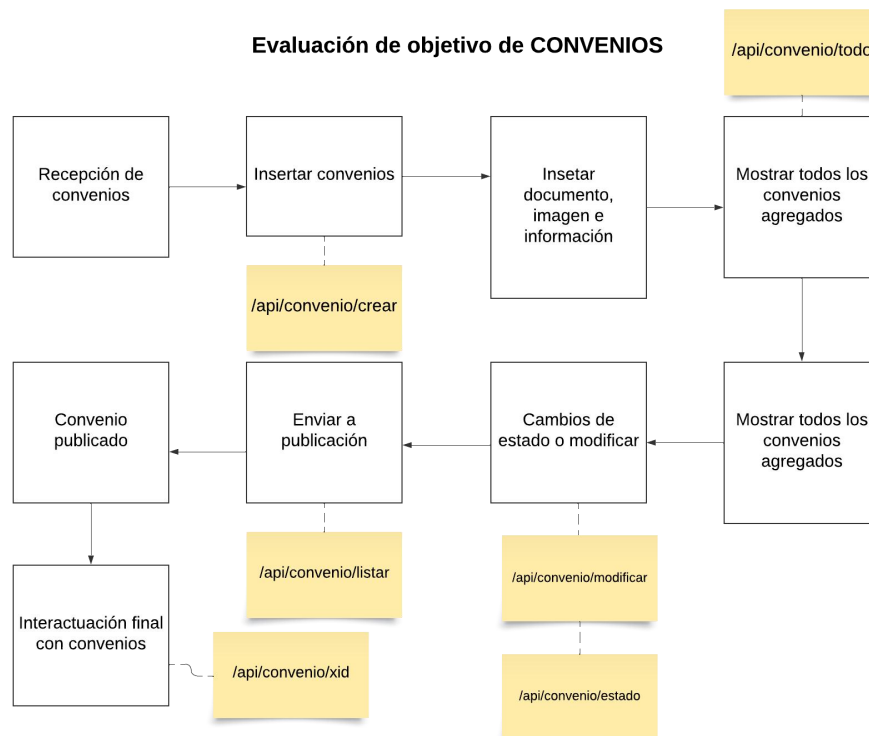


Figura 41

Ruta de Funciones de los Convenios



Gracias a la evaluación del proceso de información para cumplir los objetivos,

podemos usar la misma lógica a la hora de realizar operaciones CRUD en el sistema y recurrir a las funciones necesarias.

3.2.5. Fase de Transición

Esta fase se comprende de la transición a producción, consta de iteraciones que se encargan de cumplir esta etapa de la elaboración del software.

Los resultados obtenidos en la fase transición son aquellas que van relacionadas con la puesta a producción y los resultados finales mediante capturas del sistema final que mostramos a continuación.

3.2.5.1. Bienvenida y login del sistema

Figura 42

Pantalla Principal e Inicio de Sesión del Sistema

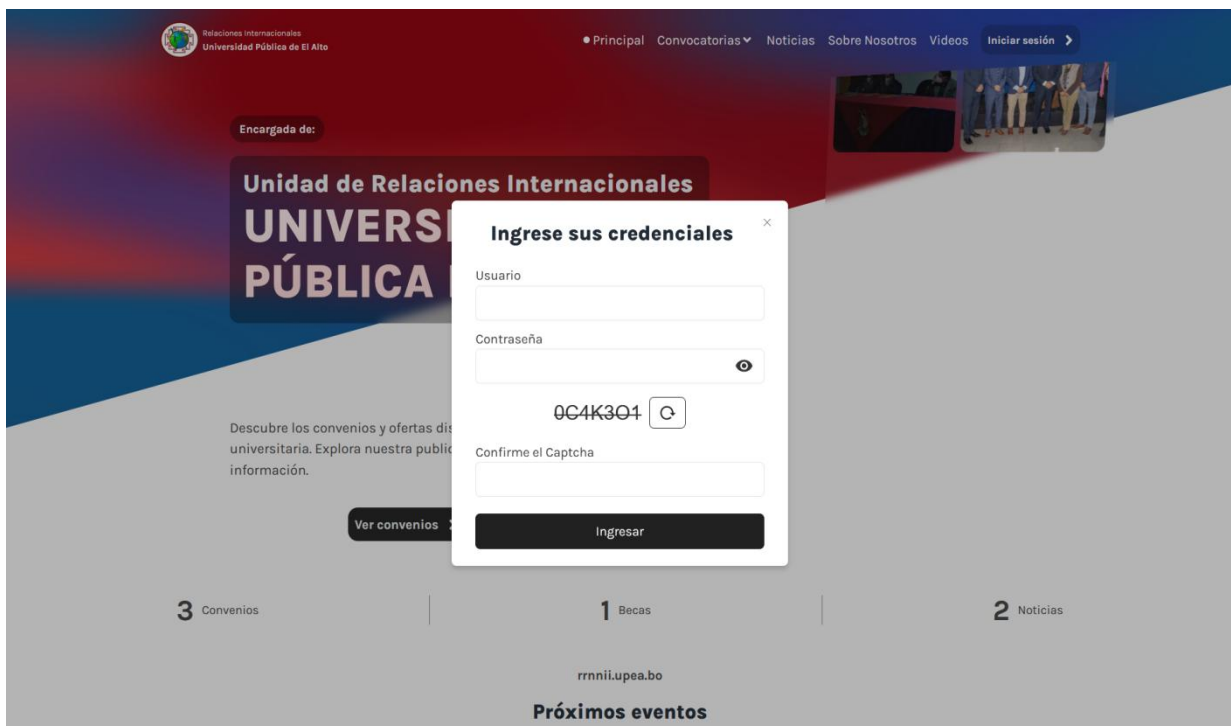


Figura 43
Vista de Gráficos del Sistema

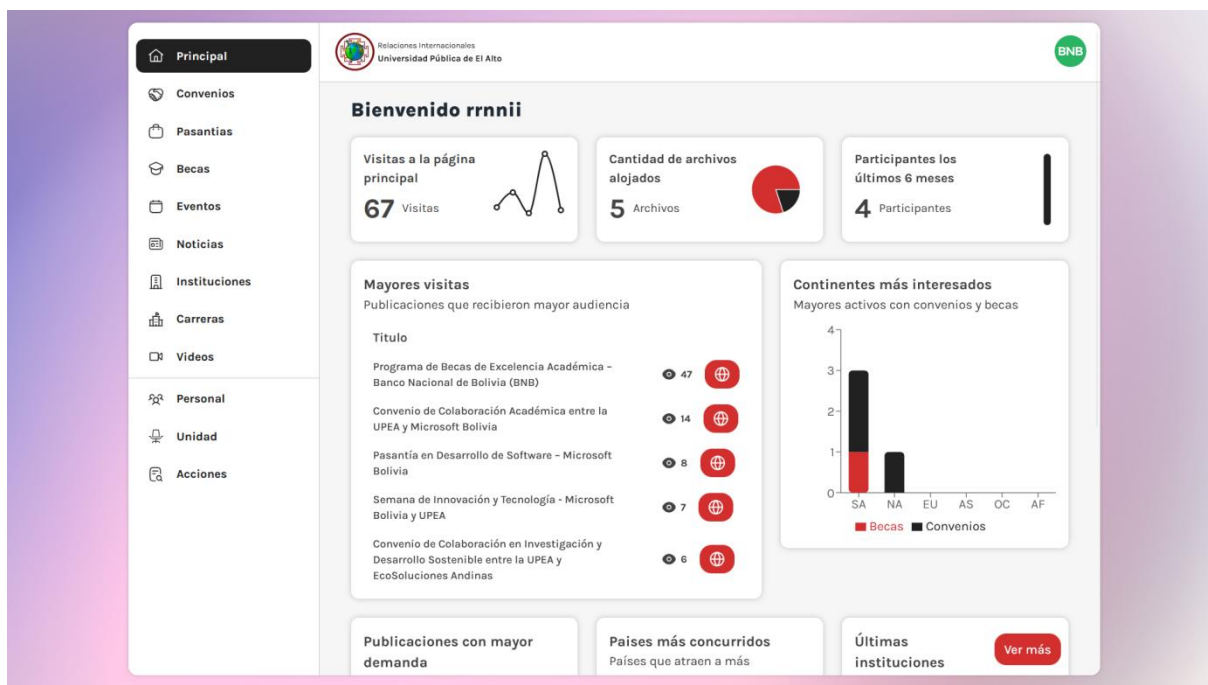
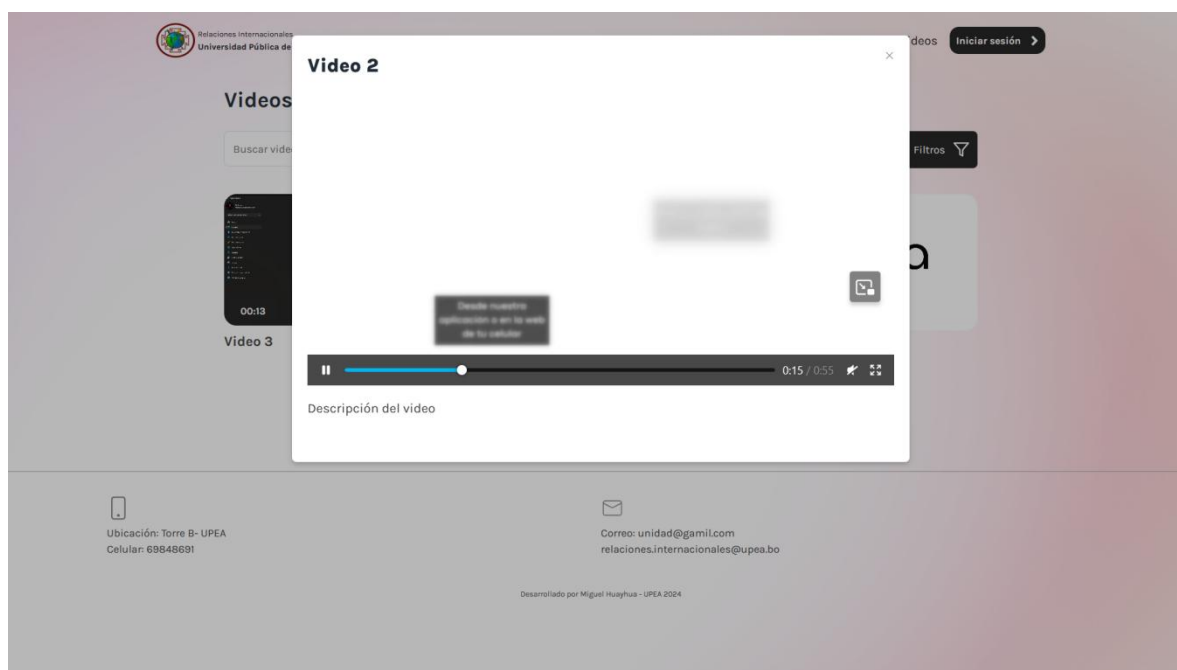


Figura 44
Listado de Vídeos del Sistema



3.2.5.2. Gestión de Convenios

Figura 45

Listado y Edición de Convenios desde Dashboard



Figura 46

Agregado de Nuevos Convenios

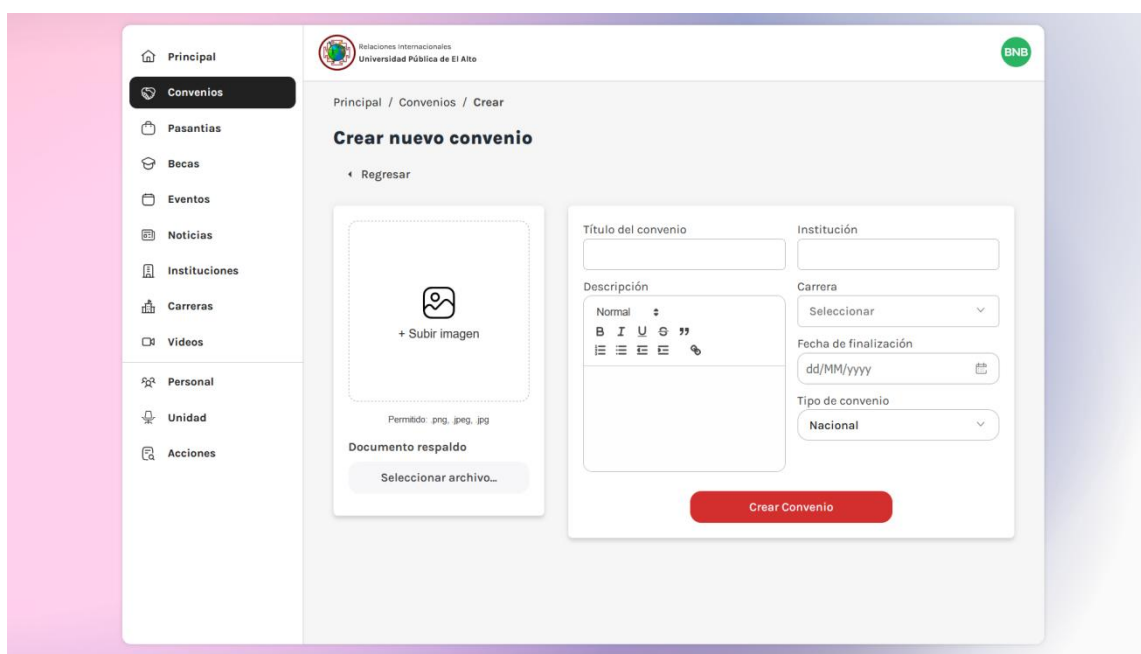


Figura 47
Listado de Convenios desde Visitante

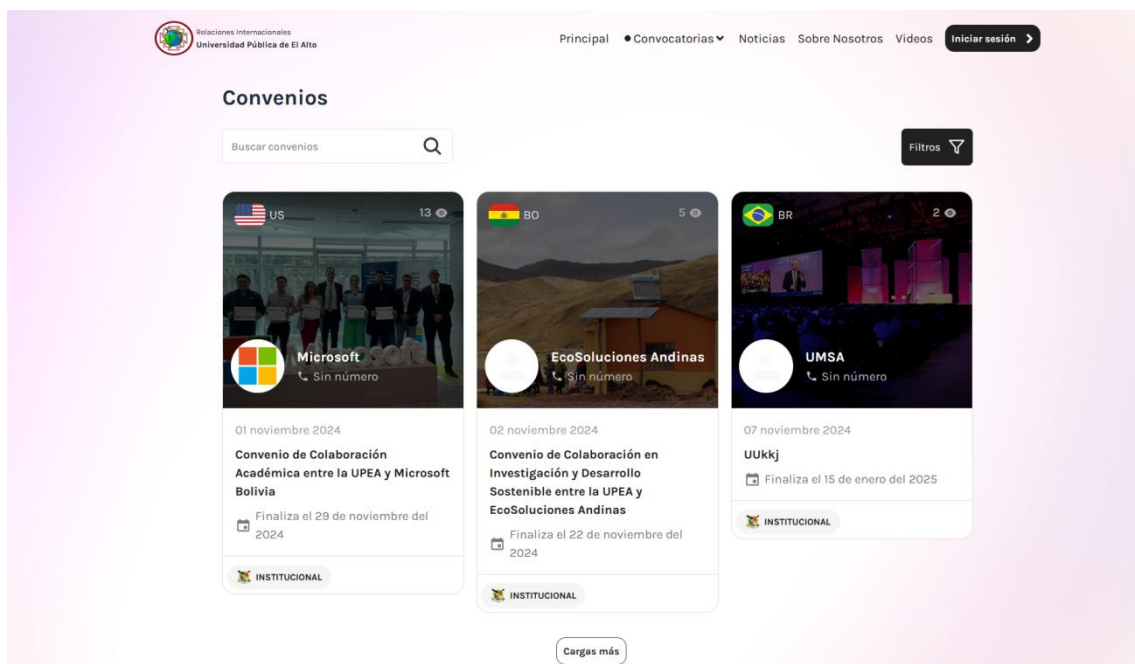


Figura 48
Visualización de Convenios



Convocatorias > Convenios > Convenio de Colaboración Académica entre la UPEA y Microsoft Bolivia

Finaliza el 29 de noviembre del 2024

Vigente

Título del convenio

Este convenio busca establecer una colaboración entre la UPEA y Microsoft Bolivia para fomentar la educación en tecnología y habilidades digitales. A través de este acuerdo, los estudiantes y docentes tendrán acceso a licencias de software, capacitaciones en desarrollo de software y

3.2.5.3. Gestión de Pasantías

Figura 49

Listado y Edición de Pasantías desde Dashboard



Figura 50

Agregado de Nuevas Pasantías

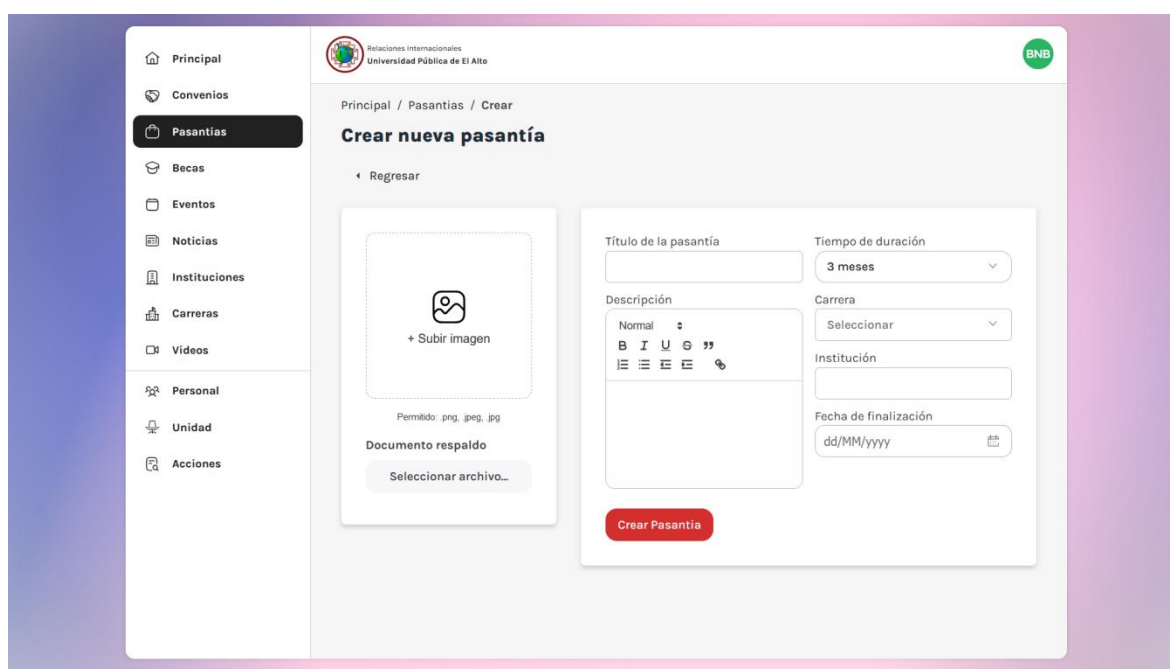


Figura 51
Listado de Pasantías desde Visitante

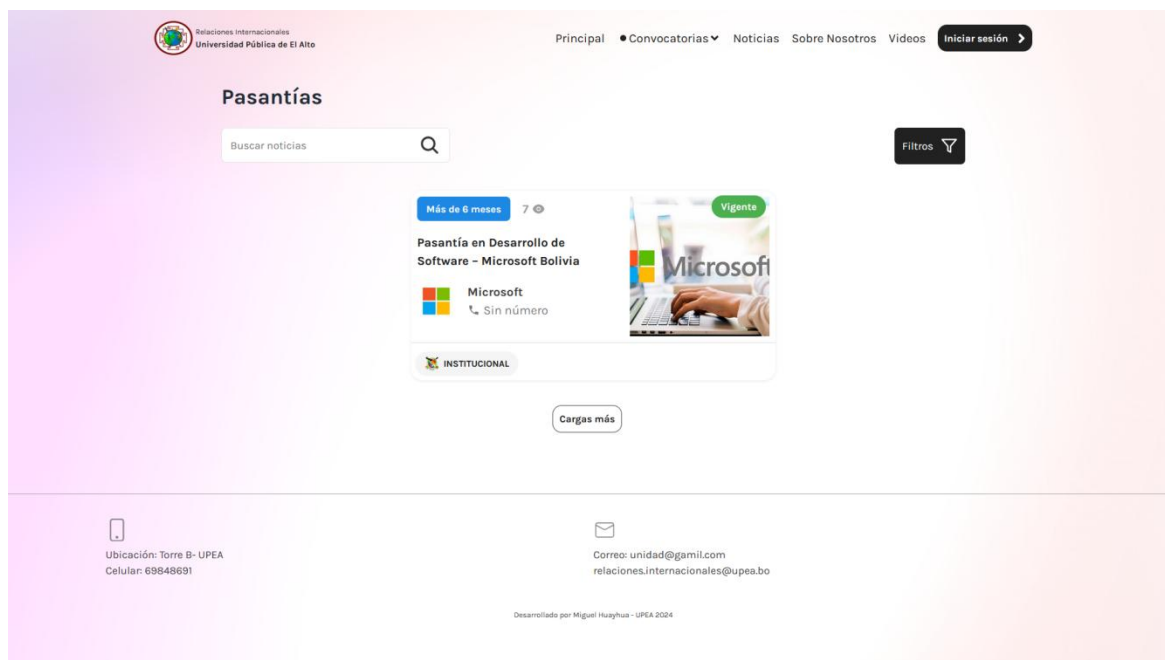


Figura 52
Visualización de Pasantías



Convocatorias > Pasantías > Pasantía en Desarrollo de Software - Microsoft Bolivia

Finaliza el 30 de noviembre del 2024

Descripción de la Pasantía:

Microsoft Bolivia, en alianza con la Universidad Pública de El Alto (UPEA), ofrece una pasantía dirigida a estudiantes de ingeniería de sistemas, informática y carreras afines. Esta pasantía se enfoca en el desarrollo de software y la implementación de soluciones basadas en la nube de Microsoft, con un enfoque práctico en la creación de aplicaciones escalables y seguras.

3.2.5.4. Gestión de Becas

Figura 53

Listado y Edición de Becas desde Dashboard

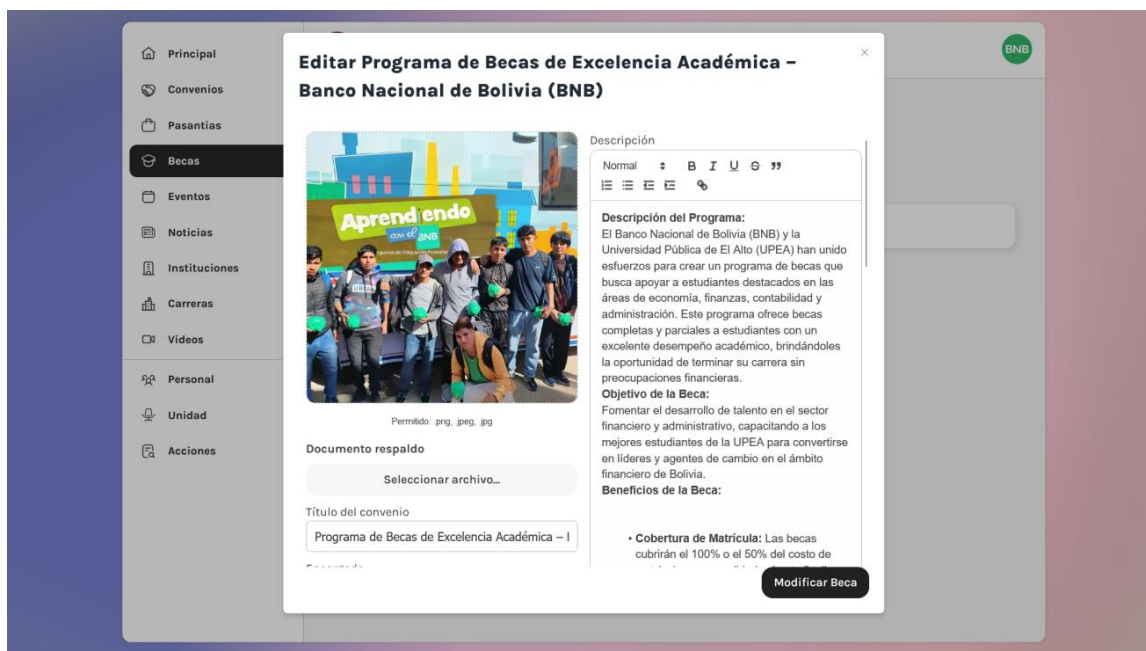


Figura 54

Creación de Nueva Beca

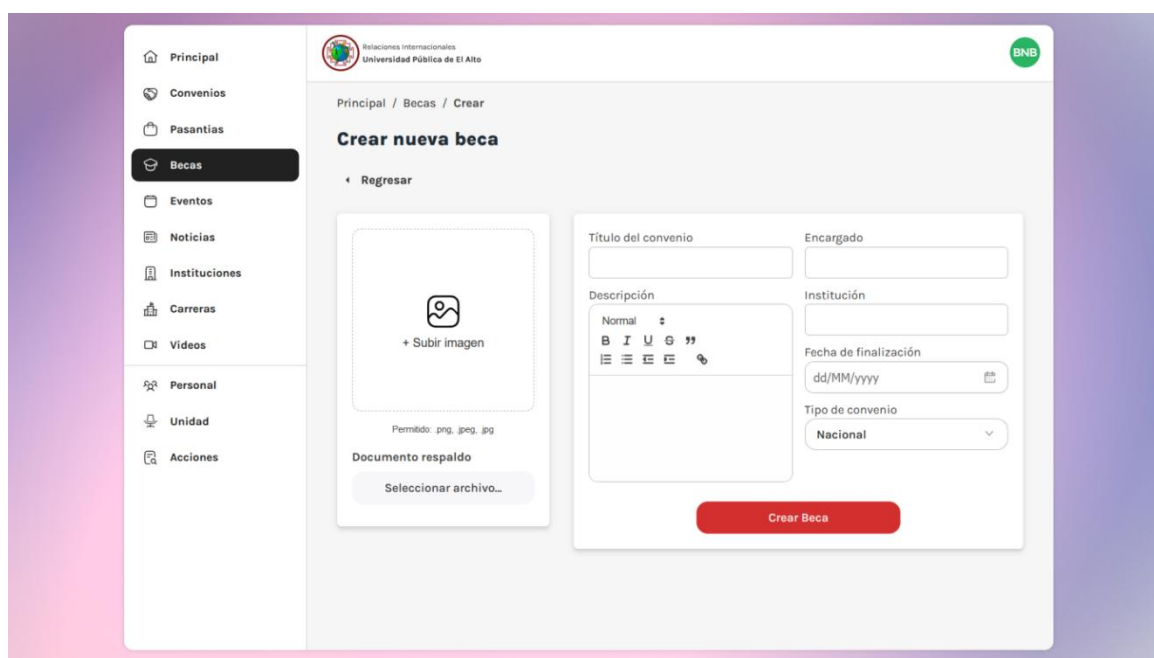


Figura 55
Seguimiento de Participantes en Becas

Principal / Becas / 6724f63e1d54191d249160e7

Programa de Becas de Excelencia Académica - Banco Nacional de Bolivia (BNB)

Sobre la beca Postulantes **4**

Generar listado

Buscar

Nombre	Registro Universitario	Cédula De Identidad	Revisado
Miguel Huayhua Condori	20002831	9190304	✓
LUIS	200	99	✓

Figura 56
Listado de Becas desde Visitantes

Principal Convocatorias Noticias Sobre Nosotros Videos Iniciar sesión

Becas

Buscar becas

Filtros

BO 46

BNB 88948855

01 noviembre 2024

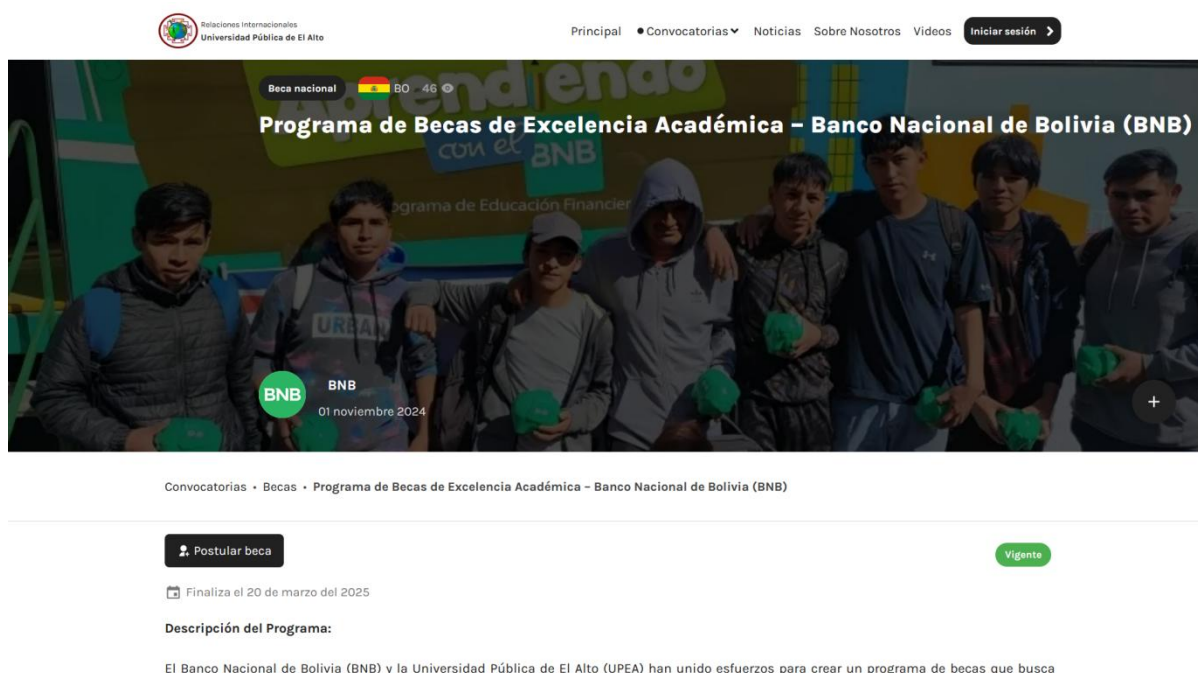
Programa de Becas de Excelencia Académica - Banco Nacional de Bolivia (BNB)

Finaliza el 20 de marzo del 2025

4 participantes

Cargas más

Figura 57
Visualización de Becas



3.2.5.5. Gestión de Eventos

Figura 58
Listado y Edición de Eventos desde Dashboard

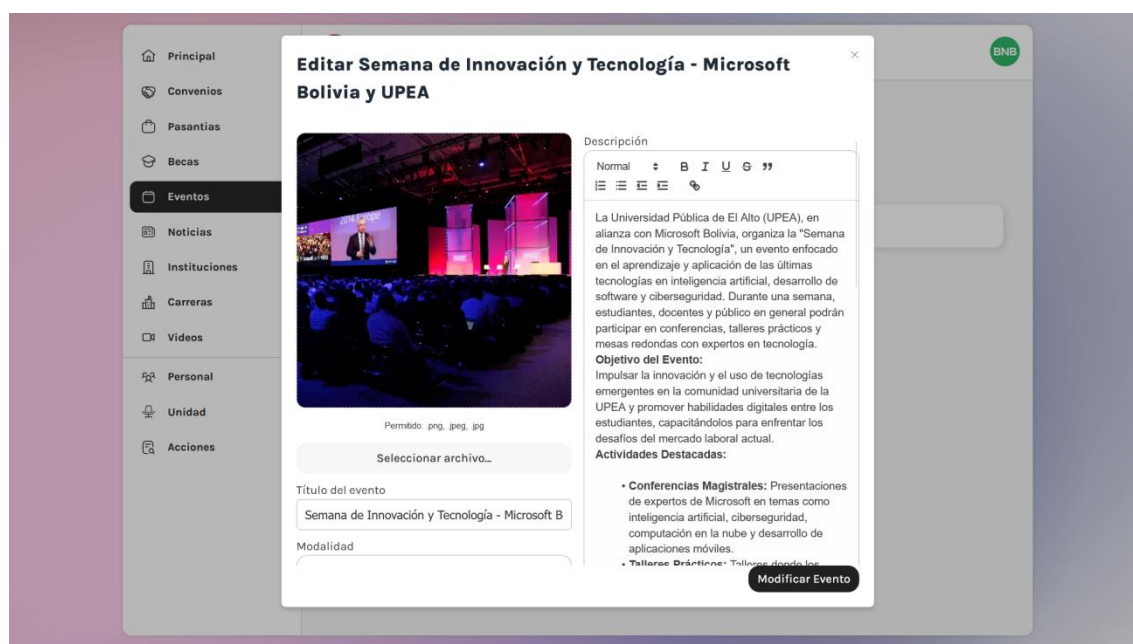


Figura 59
Creación de Nuevo Evento

The screenshot shows the 'Crear nuevo evento' (Create new event) form in the BNB system. The interface includes a sidebar menu on the left with options: Principal, Convenios, Pasantías, Becas, **Eventos** (highlighted), Noticias, Instituciones, Carreras, Videos, Personal, Unidad, and Acciones. The main content area is titled 'Principal / Eventos / Crear' and 'Crear nuevo evento'. It features a 'Regresar' (Return) link and a 'Subir imagen' (Upload image) section with a placeholder for an image and a 'Seleccionar archivo...' (Select file...) button. The form fields include: 'Título del evento' (Event title), 'Modalidad' (Modality) set to 'Online', 'Descripción' (Description) with a rich text editor, 'Comienzo' (Start) with a date picker set to 'dd/MM/yyyy', and 'Link de acceso' (Access link). A red 'Crear Evento' (Create Event) button is at the bottom right.

Figura 60
Listado de Eventos desde Visitantes

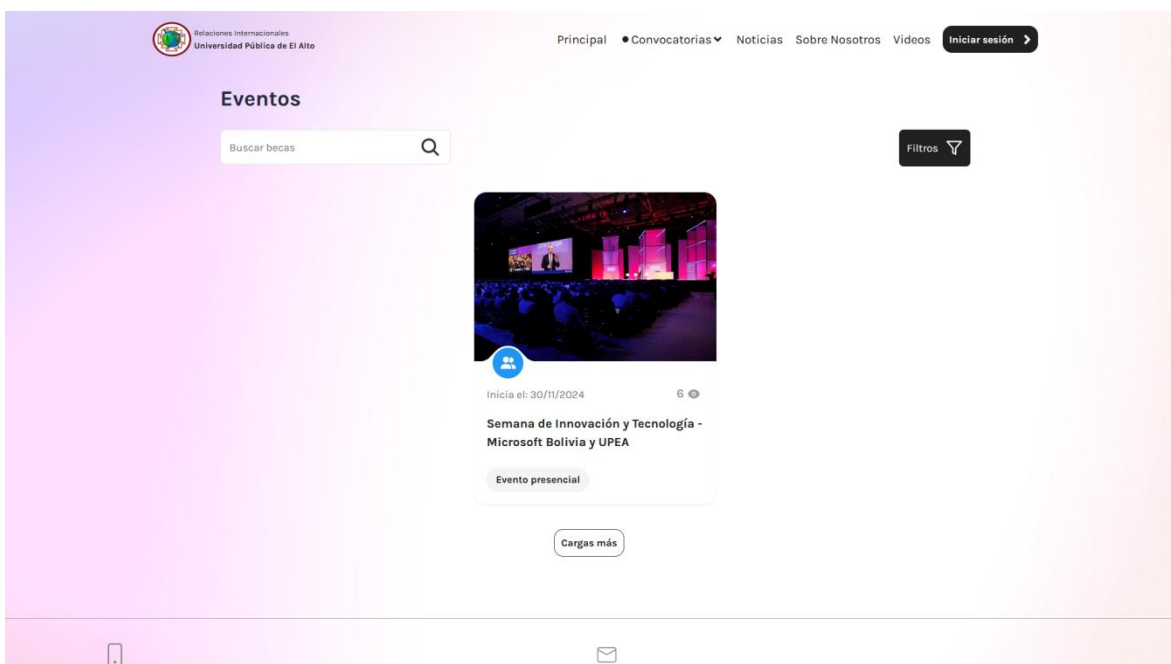
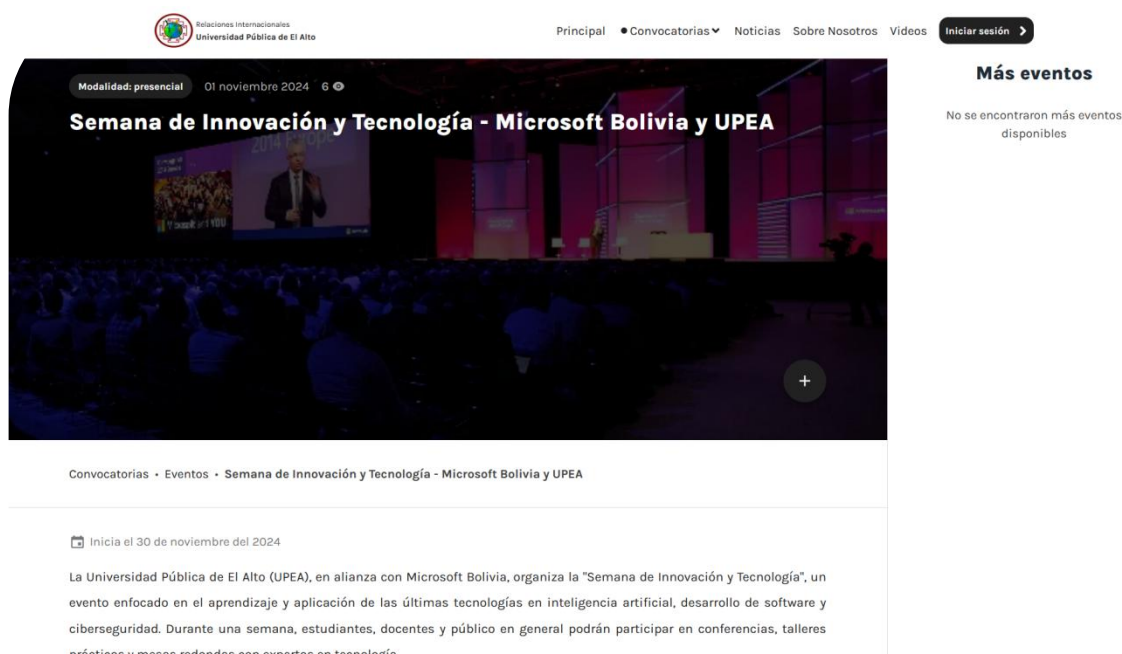


Figura 61
Visualización de Eventos



3.2.5.6. Gestión de Noticias

Figura 62
Listado y Edición de Noticias desde Dashboard

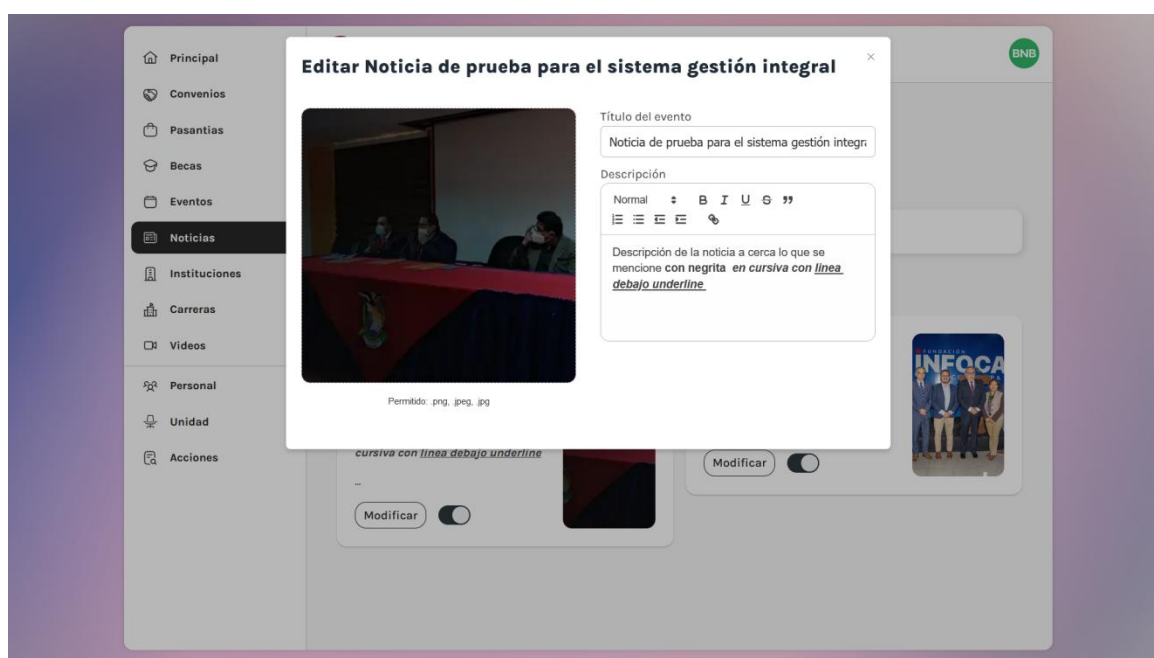
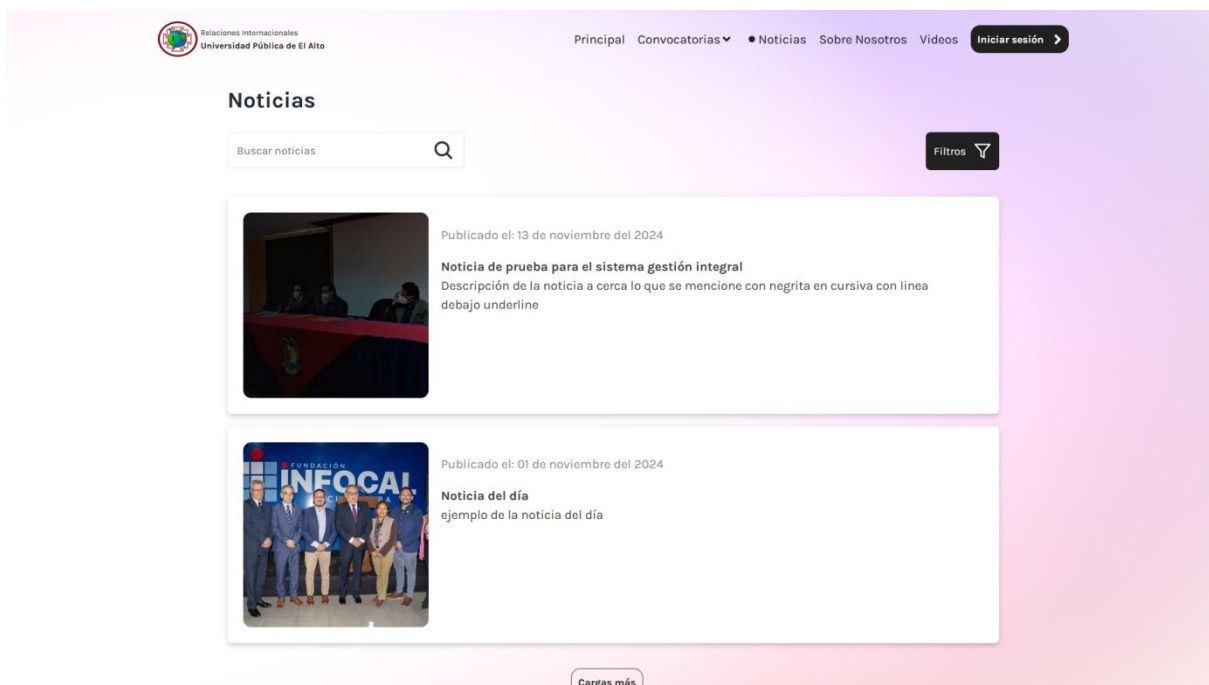


Figura 63
Creación de Nueva Noticia

The screenshot shows the 'Añadir noticia' (Add News) form in the BNB system. The interface includes a sidebar menu on the left with options: Principal, Convenios, Pasantías, Becas, Eventos, **Noticias** (highlighted), Instituciones, Carreras, Videos, Personal, Unidad, and Acciones. The main content area is titled 'Principal / Noticia / Crear' and 'Añadir noticia'. It features a 'Regresar' (Return) link and a 'Subir imagen' (Upload image) button with a camera icon. Below this, it specifies 'Permitido: png, jpeg, jpg'. The form has two main sections: 'Título de noticia' (News title) and 'Descripción' (Description). The 'Descripción' section includes a rich text editor with options for Normal, Bold (B), Italic (I), Underline (U), Strikethrough (ABC), Bulleted List (List), Numbered List (List), and Link (Link). A red 'Crear noticia' (Create news) button is at the bottom right.

Figura 64
Listado de Noticias para los Visitantes



3.2.5.7. Manual de Usuario

El manual de usuario nos permite ver las instrucciones de uso del sistema, por lo que el usuario debe seguir las pautas como sugerencia del manual para una buena administración del usuario.

3.2.5.8. Manual Técnico

El manual técnico es la principal encargada de servir como base de construcción del sistema para los desarrolladores que estén involucrados en el mantenimiento o cambios en los sistemas. Por lo cual se debe tener mucha en cuenta si se requiere una revisión y referencia que sirva de ayuda.

CAPÍTULO IV: CALIDAD, COSTOS, SEGURIDAD Y PRUEBAS

**INGENIERÍA
DE SISTEMAS**

UNIVERSIDAD PÚBLICA DE EL ALTO



4. INTRODUCCIÓN

El capítulo presente nos mostrará la calidad y costos implementados en el proyecto para tener un buen producto como software.

Para la gestión de calidad tenemos a implementar la norma ISO 9126 y para la estimación de costos utilizaremos el método COCOMO II.

4.1. GESTIÓN DE CALIDAD ISO/IEC 25000

4.1.1. Objetivos del Uso de Gestión de Calidad

Una de las dimensiones más importantes a tomar en cuenta en la medición de calidad que vienen establecidos principalmente en la ISO/IEC 25010 son las siguientes: funcionalidad, eficiencia, compatibilidad, usabilidad, mantenibilidad, portabilidad.

Para la gestión de calidad dicha norma nos permitirá evaluar atributos internos y externos de nuestro sistema para que se puedan cumplir las expectativas y los requerimientos iniciales por las partes interesadas.

Al igual que contamos con la medición de la calidad que se establecen en de la misma familia de 25000, el cual es la ISO/IEC 25023 con el realizaremos las medidas internas, externas y de calidad de uso.

4.1.2. Aplicación de la ISO 25010 (Calidad del Producto Software)

4.1.2.1. Funcionalidad

La funcionalidad nos muestra que tan capaz es nuestro sistema de ir acorde a su objetivo basado en las funciones que realiza los usuarios y el desarrollo que tiene el sistema. Por ello, se tiene un estándar para realizar la medición del mismo a través de atributos clave que todo sistema debe contar como se detalla en los siguientes parámetros de medición:

Tabla 13
Parámetros de Medición el Sistema

NRO	ATRIBUTO	CANTIDAD
1	Entradas de usuario	86
2	Salidas de usuario	45
3	Peticiones de usuario	92
4	Número de archivos	171
5	Número de interfaces externas	0

Debemos calcular los puntos de función, para ello utilizaremos factores de ponderación para obtener un resultado preciso y realizar la sumatoria.

Tabla 14
Puntos de Función sin Ajustar

PARÁMETRO	CUENTA	FACTOR			TOTAL
		SIMPLE	MEDIO	COMPLEJO	
Entradas de usuario	86	3	4	6	344
Salidas de usuario	45	4	5	7	225

Peticiones de usuario	92	3	4	6	368
Número de archivos	171	7	10	15	1710
Número de interfaces externas	0	2	7	10	0
CONTEO TOTAL					2647

Este resultado nos brinda los puntos de función sin ajustar, el cual es una forma de comprender la complejidad del sistema mediante preguntas que son cuantificadas en la tabla anterior, a continuación seleccionaremos los parámetros de medición que nos servirá para calcular los verdaderos puntos de función.

Tabla 15
Parámetros de Medición de los Puntos de Función

ESCALA		NO INFLUYE	LEVE	MODERADO	MEDIO	ALTO	CRÍTICO
		0	1	2	3	4	5
1	¿El sistema requiere copias de seguridad de la información?				x		
2	¿El sistema requiere de comunicación de datos?					x	
3	¿Existen funciones que se distribuyen dentro el sistema?					x	
4	¿El rendimiento es un elemento clave?					x	
5	¿El sistema se ejecuta en un				x		

	entorno operativo y que se utiliza?			
6	¿El sistema requiere de entrada de datos interactivos?			x
7	¿El sistema requiere de entradas que dependan de otras entradas?		x	
8	¿Los archivos se actualizan de manera interactiva?		x	
9	¿El sistema presenta de complejidad en las entradas, salidas, peticiones y archivos?		x	
10	¿El sistema presenta de complejidades en el procesamiento interno?	x		
11	¿El sistema reutiliza código?		x	
12	¿El sistema presenta de conversión para la instalación?	x		
13	¿El sistema tiene soporte para ser instalado en diferentes organizaciones?		x	
14	¿El sistema presenta un diseño para ser utilizada de manera sencilla por el usuario?			x
RESULTADOS		15	36	10
TOTAL $\sum f(i)$		61		

Los resultados serán utilizados en las siguientes fórmulas.

Ecuación 8 Puntos de Función

$$PF = \text{conteo total} * [0.65 + 0,01 * \sum f(i)] \quad (8)$$

En donde:

$\sum f(i)$ = Sumatoria de factores de ajuste

Total = Suma total de los valores de la tabla de parámetros

Reemplazando:

$$PF = 2647 * [0.65 + 0,01 * 61]$$

$PF = 3335$

Ahora calcularemos entre la suma de puntos de función máximo el cual es $\sum f(i)_{max} = 70$.

Usaremos la siguiente ecuación:

Ecuación 9 Puntos de Función Máximo

$$PF_{max} = \text{conteo total} * [0.65 + 0,01 * \sum f(i)_{max}] \quad (9)$$

Reemplazando:

$$PF_{max} = 2647 * [0.65 + 0,01 * 70]$$

$PF_{max} = 3573$

La **funcionalidad** proviene entre la relación de los puntos de función calculados

y los puntos de función máximos, dándonos la siguiente ecuación:

Ecuación 10 Funcionalidad del Sistema

$$F = PF/PF_{\max} * 100 \quad (10)$$

En donde:

F = funcionalidad del sistema

PF = Puntos de función obtenidos a partir de los parámetros de medición calculados

PF_{max} = Puntos de función máximos de la tabla de de parámetros de medición

Reemplazando:

$$F = 3335/3573 * 100$$

$F = 93,3\%$

Los resultados nos muestran que la funcionalidad del sistema es de 93,3%, el cual es un resultado bueno para medir la funcionalidad de nuestro sistema en base al cumplimiento de nuestros objetivos.

4.1.2.2. Eficiencia

La eficiencia nos mide la capacidad del sistema de utilizar los recursos otorgados para el sistema. Usaremos la siguiente tabla determinar la eficacia del sistema.

Tabla 16

Valores de Escala de Eficiencia

ESCALA DE EFICIENCIA	VALOR
Excelente	5
Bueno	4
Aceptable	3
Malo	2
Pésimo	1

Ahora nos basamos en las siguientes preguntas para identificar la eficiencia del sistema:

Tabla 17
Preguntas de ponderación de eficiencia

PREGUNTA	PONDERACIÓN
¿La interfaz permite la entrada de datos de manera eficiente?	5
¿Las operaciones se ejecutan fácilmente?	4
¿Los datos de salida son comprensible por el usuario?	5
¿Las operaciones jerárquicas son fáciles para el usuario?	4
¿El sistema responde de manera eficaz ante consultas o búsquedas?	4
TOTAL	22

Usaremos la siguiente ecuación para calcular la eficiencia del sistema:

Ecuación 11 Eficiencia del Sistema

$$EF = \frac{\sum e_i}{n^2} * 100\% \quad (11)$$

En donde:

EF = Eficiencia del sistema

$\sum e_i$ = sumatoria de los valores de ponderación de eficiencia del sistema

n = Número de preguntas de eficiencia

Reemplazando:

$$EF = \frac{22}{5} * \frac{100}{5}$$

EF = 88%

Los resultados obtenidos para el cálculo de la eficiencia nos indican que el sistema tiene porcentaje de 88%, lo cual nos indica que presenta un nivel de eficiencia satisfactorio para su funcionamiento.

4.1.2.3. Usabilidad

La usabilidad nos mide la facilidad en las que los usuarios pueden interactuar con el sistema. Enfocándose mayormente en la experiencia para el usuario y la capacidad con el que el usuario puede interactuar con el sistema.

A continuación tenemos una tabla con la que nos servirá de guía para la ponderación de valores:

Tabla 18
Escala de Usabilidad del Sistema

ESCALA DE USABILIDAD	VALOR
Excelente	5
Bueno	4
Aceptable	3
Malo	2
Pésimo	1

Tabla 19
Preguntas de Ponderación de Usabilidad

PREGUNTA (x)	PONDERACIÓN
¿El sistema es comprensible por el usuario?	5
¿El sistema presenta de una fácil accesibilidad para el usuario?	5
¿El sistema satisface las necesidades que se requieren?	5
¿La información es totalmente representativa?	5
¿Son las respuestas del sistema satisfactorias?	4
¿Está de acuerdo con el funcionamiento del sistema?	5
¿El sistema responde a las peticiones que se le indican?	5
TOTAL	34

La siguiente fórmula nos representa la usabilidad del sistema:

Ecuación 12 Usabilidad del Sistema

$$U = \left(\frac{\sum x_i}{n} * 100\% \right) / 5 \quad (12)$$

En donde:

U = Usabilidad del sistema

$\sum x_i$ = Sumatoria de los parámetros de usabilidad

n = Número de parámetros utilizados en el sistema

Reemplazando:

$$U = \left(\frac{34}{7} * 100\% \right) / 5$$

$U = 97,14\%$

Los resultados nos indican que el sistema presenta un nivel de usabilidad del 97,14%.

4.1.2.4. Mantenibilidad

La mantenibilidad nos mide lo fácil que es corregir errores, actualizar y realizar cambios dentro del sistema. Un sistema con buena mantenibilidad es fácil de entender y modificarlo a través del tiempo.

Existe una medición llamada el índice de madurez del software, en donde un resultado alto indica que el software es lo suficientemente estable lo que indica que

puede ser sujeto a mantenimiento y durar mientras transcurra el tiempo. La siguiente ecuación nos servirá para medir la mantenibilidad del software.

Ecuación 13 Índice de Madurez del Software

$$Md = \frac{Mt - (Fa + Fc + Fd)}{Mt} * 100\% \quad (13)$$

En donde:

Md = Nivel de madurez del sistema

Mt = Número de módulos totales para la última versión del sistema = 9

Fd = Número de módulos que se eliminan en la actualización = 0

Fc = Número de módulos que se modifican en cada versión = 1

Fa = Numero de módulos nuevos que se añaden a cada versión = 0

Reemplazando:

$$Md = \frac{9 - (0 + 1 + 0)}{9} * 100\%$$

Md = 88,8 %

Los resultados nos indican que el sistema presenta una mantenibilidad de 88,8%.

4.1.2.5. Portabilidad

La portabilidad se refiere al entorno donde el sistema se ejecuta, puede

ejecutarse en cualquier sistema operativo con NodeJS presente ya sea Windows o variantes de Linux. Para calcular el nivel de portabilidad de nuestro sistema, haremos el uso de la nuestra siguiente ecuación

Ecuación 14 Grado de Portabilidad del Sistema

$$P = \frac{Es}{Mn} * 100\% \quad (14)$$

En donde:

P= Portabilidad del sistema

Es = Cantidad de entornos soportados

Mn = Cantidad de modificaciones necesarias

Ilustramos en una tabla la cantidad de entornos y las modificaciones que se necesitarían en el caso de cambiar sistemas operativos.

Tabla 20
Entornos Soportados del Sistema

Entorno	Modificación	Descripción
Linux	Ninguna	El sistema funciona correctamente sin modificaciones
Windows	Modificación 1	Ajuste de rutas de archivos (\ en lugar de /)
	Modificación 2	Configuración de Prisma para permisos de archivos

Observando la cantidad de entornos y los cambios necesarios que se ven involucrados en cada caso tenemos como resultado $E_s = 2$ entornos y $M_n = 2$ modificaciones necesarias.

Reemplazando:

$$P = \frac{2}{2} * 100$$

$$P = 100 \%$$

Tenemos como resultado una portabilidad del 100% para el sistema, siendo este un resultado muy bueno para saber si nuestro sistema presentará la facilidad de ser traspasado entre estos 2 sistemas operativos.

4.1.2.6. Resultados obtenidos

Luego de haber obtenido los resultados para los aspectos importantes que nos miden la calidad del software y que nos establece la ISO/IEC 25010 que parte de la ISO/IEC 25000, tenemos el conjunto de resultados que nos muestra el desempeño general de nuestro sistema.

Tabla 21

Resultados de Calidad Aplicando la ISO 25010

Atributo evaluado	Resultado obtenido
Funcionalidad	93,3%
Eficiencia	88%
Usabilidad	97,14%

Mantenibilidad	88,8%
Portabilidad	100%
Total	93,38%

Luego de haber obtenido los resultados de calidad, tenemos la conclusión de que el sistema tiene un 93,38% de calidad de software, lo cual nos representa un porcentaje muy bueno a la hora de mostrarlo como un sistema que resuelve las necesidades de los usuarios solicitantes.

4.2. SEGURIDAD DEL SOFTWARE (ISO/IEC - 27000)

La seguridad del software implementando los estándares de la ISO 27000 se encarga de la seguridad de los datos o información que se guarda y genera dentro el sistema. El conjunto de normas de la ISO/IEC 27000 nos indican acerca de la gestión de riesgos, reglas de acceso, seguridad interna y externa, encriptación de datos y sesiones que generar el sistema para el acceso de usuarios permitidos dentro de espacios restringidos del sistema.

4.2.1. Integridad y Confidencialidad de la Información (ISO 27001)

La norma que abordaremos es la que se encarga y asegura la integridad de la información, para ello, debemos implementar 2 métodos de seguridad como ser la seguridad lógica y la seguridad física, con el objetivo de permitir el acceso solamente a usuarios autorizados mediante las tecnologías que hayamos aplicado en el trascurso del desarrollo.

4.2.1.1. Seguridad Lógica

Como su nombre lo dice, incluye las medidas de seguridad aplicadas a los datos como encriptaciones, sesiones, inicios de sesión, registro de acciones.

Las medidas de seguridad implementadas son las siguientes:

- Sesiones seguras
- Encriptación de contraseñas
- Registro de acciones en la base de datos
- Autenticación segura
- Comprobación de datos
- Privilegios de usuario

La seguridad lógica aplicada al acceso de usuarios se realiza mediante la librería NextAuth, en donde utilizamos el enfoque de Json Web Token, que viene con una seguridad extra de encriptación creando un JWE (Json Web Encryption) , lo cual nos genera una sesión con una firma segura y genera el token con un vencimiento dado, la obtención de un token se realiza con la autenticación del usuario desde el login.

Figura 65

Administración de Token JWT desde NextAuth

```

callbacks: {
  async signIn({ account }) {
    if (account!.provider === 'credentials') {
      return true
    }
    return false
  },
  jwt({ token, user, trigger, session }) {
    if (user) {
      token.rol = user.rol;
    }
    if (trigger === 'update') {
      token.name = session.usuario;
      token.picture = session.avatar;
      token.rol = session.rol;
    }
    return token
  },
  session({ session, token }) {
    session.user.rol = token.rol;
    return session;
  }
},

```

Figura 66*Token de Sesión del Sistema Gestionado en NextJs*

```

▼ next-auth.session-token:"eyJhbGciOiJIUzI1LCJlbnMiOiJBM...-8Gwxu.y72dXFs6wUwXw_vSI1606g"
  Creado:"Mon, 07 Oct 2024 17:35:03 GMT"
  Domain:"localhost"
  Expires / Max-Age:"Mon, 02 Dec 2024 17:38:45 GMT"
  HostOnly:true
  HttpOnly:true
  Path:"/"
  SameSite:"Lax"
  Secure:false
  Tamaño:376

```

Las contraseñas se almacenan mediante una encriptación de 10 saltos con la ayuda de la librería Bcrypt que se encarga de mantener las contraseñas seguras sin la posibilidad de volver a su valor original y siendo exclusivamente para los usuario con el acceso aprobado en el sistema.

Figura 67

Generación de Saltos para la Contraseña de Usuarios

```
let res = await prisma.usuario.update({
  data: {
    usuario: usuario as any,
    ...password ? {
      password: bcrypt.hashSync(password, bcrypt.genSaltSync(10))
    } : null
  },
});
```

El sistema proporciona la capacidad de modificar contraseñas, siendo el administrador con el mayor control de todos los usuarios, el personal que tiene el control de su propia cuenta y tiene permitido cambiar los datos de usuario como ser el nombre de usuario, contraseña y la foto de perfil. Los usuarios son los responsables de tener almacenado sus contraseñas de manera segura más allá del hash aplicado con Bcrypt.

Figura 68

Control de Usuarios para el Administrador

Relaciones Internacionales
Universidad Pública de El Alto

Principal / Personal / Listado

Personal

Añadir Personal

Todos 2 Activos 2 Inactivos 0

Buscar

CI	Datos Personales	Creado El	
999999	Edwin Campos Hugarte	01/11/2024 09:38:07	Modificar
999	cristina h	07/11/2024 09:30:55	Modificar

Filas 10 Página 1

Figura 69
Modificación del Perfil de Usuario

The screenshot displays the 'Perfil de usuario' (User Profile) page. On the left is a sidebar with navigation links: Principal, Convenios, Pasantías, Becas, Eventos, Noticias, Instituciones, Carreras, Videos, Personal, Unidad, and Acciones. The main content area has a header with the university logo and a 'BNB' button. Below the header, there's a 'Principal / Perfil' breadcrumb and a 'Regresar' link. The 'Perfil de usuario' section features a profile card with a 'BNB' avatar, a 'Cambiar avatar' button, and fields for 'Usuario' (rrnnli), 'Contraseña' (masked with dots), and 'Verificar contraseña' (masked with dots). A red 'Guardar cambios' button is at the bottom of the card. To the right of the card, the 'Información personal' section displays: 'Cédula de identidad: 999999', 'Nombre completo: Edwin Campos Hugarte', 'Fecha de nacimiento: 28/05/2024', 'Cargo en la unidad: jefe', and 'Registrado el 01/11/2024 - 09:38:07'.

NextAuth tiene la protección CSRF que viene incluido en las cookies del sistema y las cuales son utilizadas para realizar solicitudes con una seguridad extra y evitar las solicitudes no deseadas.

Figura 70
Token CSRF Gestionado por NextAuth

```
next-auth.csrf-token:"efcacef75ef41659ad8c251f8ecc5a4...0c96a4711fc9caad81fdad15977adc3"
Creado:"Sat, 02 Nov 2024 16:58:24 GMT"
Domain:"localhost"
Expires / Max-Age:"Sesión"
HostOnly:true
HttpOnly:true
```

Además, se valida los datos entrantes en el registro de nueva información en las tablas que tengan restricción de tipo de datos como ser: tipo de dato, espacios en blanco, fechas inválidas.

Figura 71
Validación de Datos para el Módulo de Convenios

Principal / Convenios / Crear

Crear nuevo convenio

Regresar

+ Subir imagen

Permitido: png, jpeg, jpg

Documento respaldo

Seleccionar archivo...

Título del convenio

Título no puede quedar vacío

Institución

Institución no puede quedar vacío

Selección

Seleccione al menos una carrera

dd/MM/yyyy

Finalización no puede quedar vacío

Nacional

Crear Convenio

El inicio de sesión se tiene la protección Captcha generadas manualmente con caracteres implementados con la librería de react-random-string y evitar inicios de sesión falsos en el sistema.

Figura 72
Modal de Inicio de Sesión del Usuario

Ingreso sus credenciales

Usuario

Contraseña

FCE3ACY

Confirme el Captcha

Ingresar

4.2.1.2. Seguridad Física

La seguridad física está encargada de los administradores de los servidores y encargados del área de la información, en este caso, de la unidad de sistemas información y estadística de la UPEA, que viene siendo: el sistema general de información, sistema de archivos y la base de datos.

Los usuarios contribuyen en la seguridad física en la ética de privacidad de sus datos para evitar el robo de información personal como ser el nombre de usuario o la contraseña elegida por el usuario. Lo cual garantiza el entorno seguro para la información guardada.

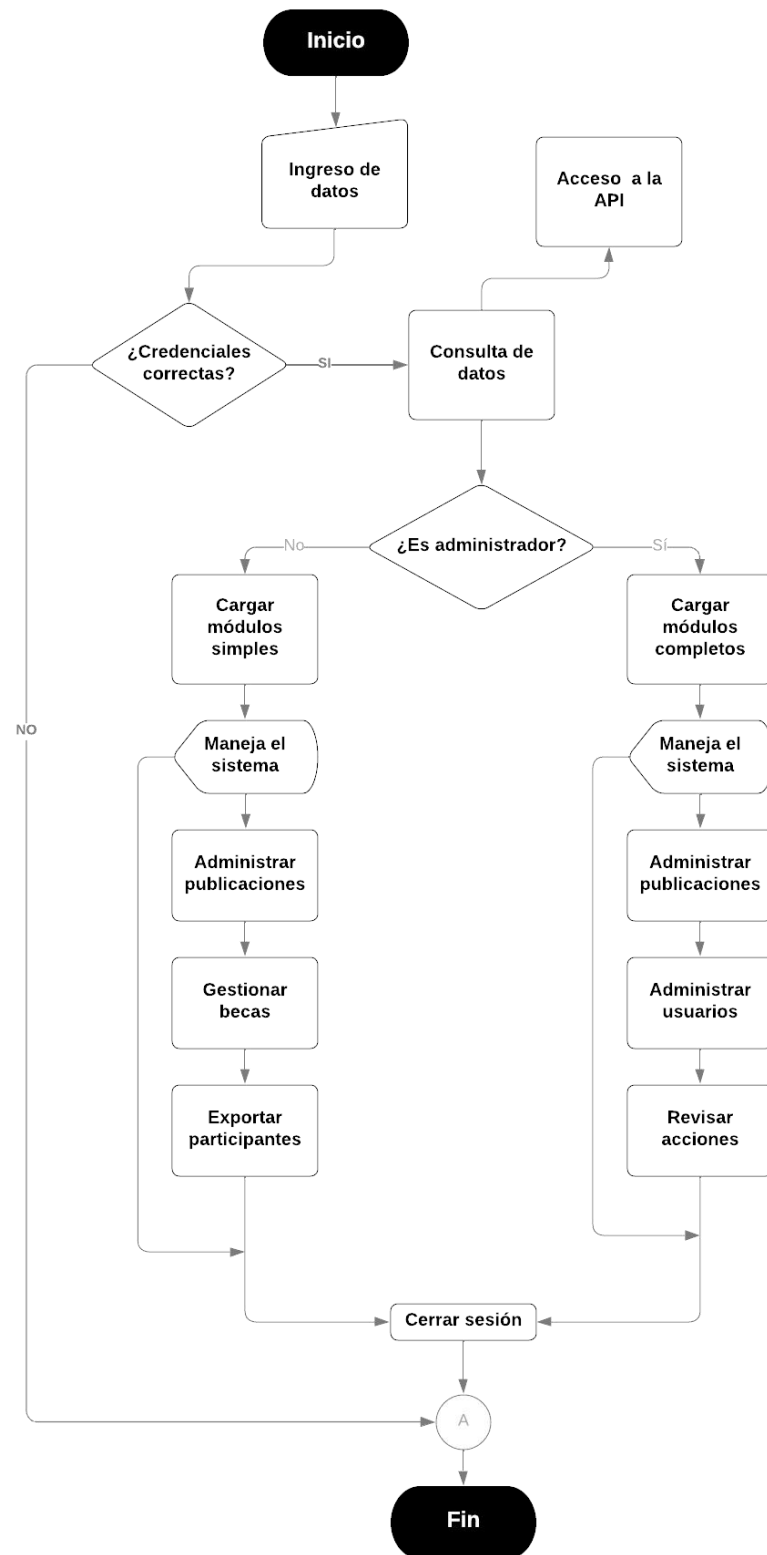
4.3. PRUEBAS DEL SOFTWARE

Las pruebas del software se dividen en prueba de caja blanca y negra, además de poner el sistema con cargas de Stress y verificar las capacidades del sistema ante peticiones demandantes de usuarios ficticios que accedan a la información solicitada en intervalos de tiempo.

4.3.1. Prueba de Caja Blanca

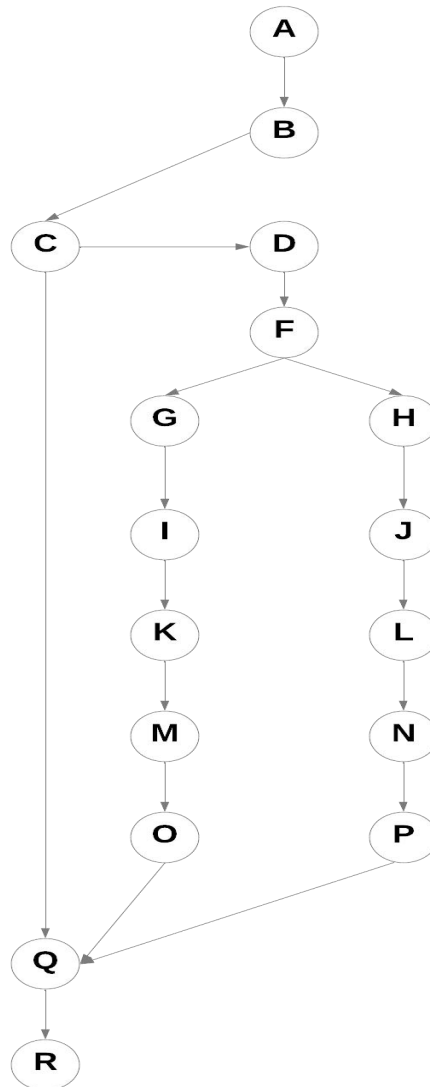
Este método nos verifica las capacidades lógicas de nuestro sistema, utilizaremos una técnica ciclomática de complejidad para colocarlo a prueba, el cual nos muestra los diversos caminos que pueden tomar los usuarios encargados en la administración del sistema. Esta prueba del cálculo de complejidad ciclomática estará enfocada en el proceso que se realiza desde el inicio de sesión hasta el cierre de sesión para los 2 tipos de usuarios.

Figura 73
Diagrama Ciclomática de Complejidad



Creemos un grafo que nos permita interpretar nuestro diagrama ciclomática del sistema.

Figura 74
Grafo Ciclomático de Complejidad



En donde:

A = Inicio de sistema

B = Ingreso de credenciales

C = Revisión de credenciales

D = Consulta de datos

F = Verificación de administrador

G = Cargado de módulos simples de gestión de información

H = Cargado de módulos completos para el administrador

I = Manejo completo del sistema para el personal

J = Manejo completo del sistema para el administrador

K = Administración de publicaciones para el personal

L = Administración de publicaciones para el administrador

M = Gestión de becas

N = Administración de usuarios

O = Exportación de participantes

P = Revisión de acciones

Q = Cierre de sesión

R = Fin del sistema

Ahora que analizamos la complejidad ciclomática, podemos evaluar la complejidad del sistema mediante la siguiente ecuación.

Ecuación 15 Complejidad Ciclométrica

$$V(G) = A - N + 2$$

(15)

En donde:

N = Número de nodos = 17

A = Número de aristas = 18

Reemplazando:

$$V(G) = 18 - 17 + 2$$

$$V(G) = 3$$

Mostrar las trayectorias o caminos que el sistema puede tomar son 3, los cuales los describiremos para entender las variables involucradas en los procesos que pueden tener los usuarios.

Tabla 22
Caminos del Sistema desde el Inicio de Sesión

CAMINO	Nodos involucrados
Camino 1	A - B - C - Q - R
Camino 2	A - B - C - D - F - G - I - K - M - O - Q - R
Camino 3	A - B - C - D - F - H - J - L - N - P - Q - R

Los caminos que pueden tomar cada usuarios es dependiendo el tipo de personal o las credenciales que son ingresadas, a continuación describimos cómo cada camino influye en la gestión de información dentro del sistema para cada

categoría de usuarios y los no admitidos.

Tabla 23

Caminos del Sistema Basado en el Tipo de Usuario

CAMINO	Descripción
Camino 1	El usuario que ingresa las credenciales no es un usuario que tenga la autorización de uso dentro el sistema, por lo que no tiene acceso para la administración y el sistema lo rechaza.
Camino 2	El usuario tiene el rol de secretario(a) o técnico de la Unidad, por lo que tienen el acceso autorizado y la administración de publicaciones como ser convenios, becas, eventos y noticias. No tiene ciertos privilegios del administrador.
Camino 3	El usuario es de tipo administrador, por lo que corresponde el uso para el jefe de la Unidad, tiene los mismos privilegios que el usuario común con a diferencia que tiene la capacidad de ver las acciones realizadas por otros usuarios y cambiar o crear al personal dependiendo las necesidades de los usuarios que dependen del mismo.

4.3.2. Prueba de Caja Negra

Realizamos las pruebas que van acorde con el comportamiento del sistema y los requisitos funcionales del sistema, veremos todos los requisitos en orden a medida que se muestran el proceso de inicio de sesión y los tipos de datos involucrados en cada campo.

Figura 75*Ingreso de credenciales desde el login*
Tabla 24*Tipos de Datos de Inicio de Sesión*

ENTRADA	TIPO	RESTRICCIONES
Usuario	Cadena de texto	Espacios en blanco
Contraseña	Cadena de texto	Espacios en blanco
Captcha	Cadena de texto	Espacios en blanco, sin confirmación de Captcha.

Una vez dentro del sistema se tiene una vista del sistema con la información acumulada, en nuestro caso verificaremos el agregado de un convenio.

Figura 76*Pantalla Principal de Bienvenida del Sistema*

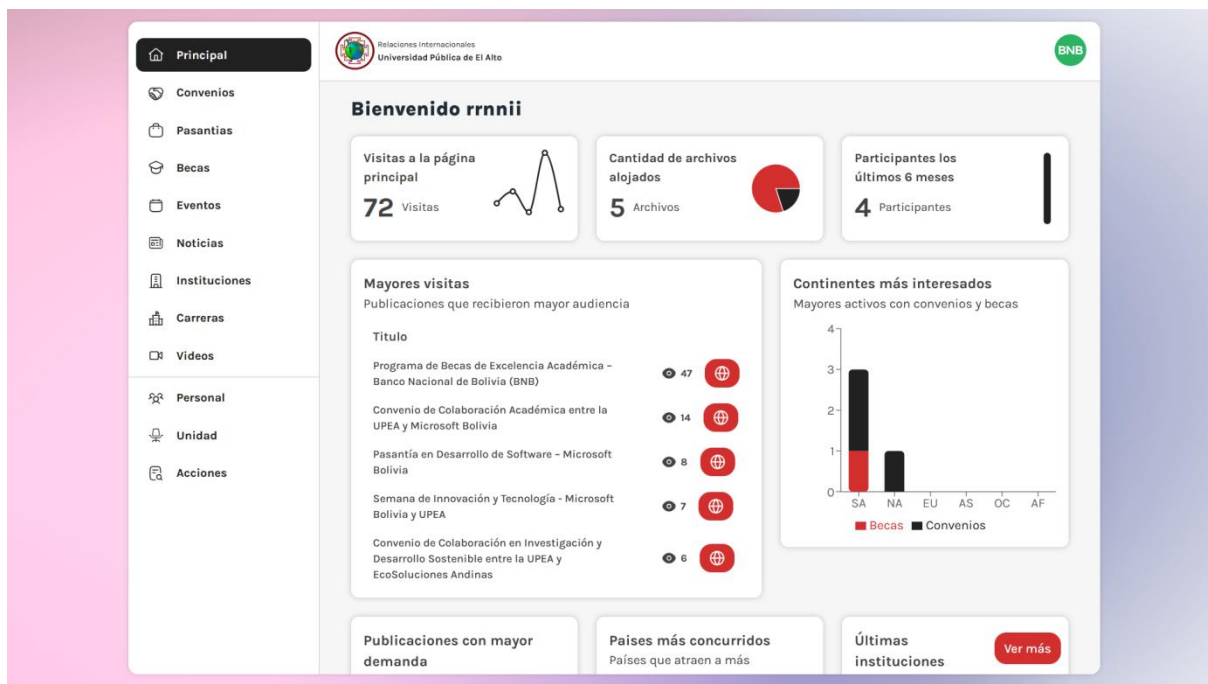


Figura 77
Prueba de Caja Negra - Creación de Convenio

Crear nuevo convenio

Principal / Convenios / Crear

Regresar

Documento respaldo

Permitido: png, jpeg, jpg

Seleccionar archivo...

4.2 Evento.pdf 205.55KB

Título del convenio
CONVENIO DE COOPERACIÓN ENTRE LA UPEA Y LA UNIVERSIDAD PRIVADA DOMINGO SAVIO

Institución
Domingo Savio

Carrera
INGENIERÍA DE SISTEMAS INSTITUCIONAL

Fecha de finalización
30/11/2024

Tipo de convenio
Nacional

Descripción

OBJETO
El presente convenio tiene como finalidad promover la cooperación entre la Universidad Pública de El Alto (UPEA) y la Universidad Privada Domingo Savio (UPDS) en actividades académicas, de investigación y de extensión universitaria, en

Los datos se verifican antes de acceder al modal de pregunta para la creación

de un nuevo convenio, a continuación, se presentan los campos y sus dichas verificación para tener el paso de alerta.

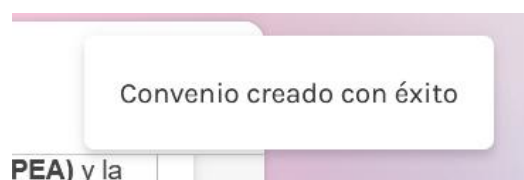
Tabla 25
Entradas de Registro de Convenio

ENTRADA	TIPO	RESTRICCIONES
Titulo	Cadena de texto	Espacios en blanco
Descripción	Cadena de texto	Sin restricciones
Imagen	Cadena de texto	Sin imagen, vacío
Pdf	Cadena de texto	Sin restricciones
Institucion	Cadena de texto	Espacios en blanco, sin seleccionar
Finalizacion	Cadena de texto	Espacios en blanco, sin seleccionar
Tipo	Cadena de texto	Espacios en blanco, sin seleccionar
Estado	Booleano	Sin restricciones, auto generado
Id	Cadena de texto	Sin restricciones, auto generado
Pais	Cadena de texto	Sin restricciones, auto generado.
Continente	Cadena de texto	Sin restricciones, auto generado
Conteo	Número	Sin restricciones, auto generado

Figura 78
Alerta del Sistema para el Usuario



Figura 79
Alerta de Nuevo Convenio Registrado



Los datos insertados y disponibles dentro el módulo de convenios son los siguientes:

Figura 80
Datos Insertados en el Módulo de Convenios

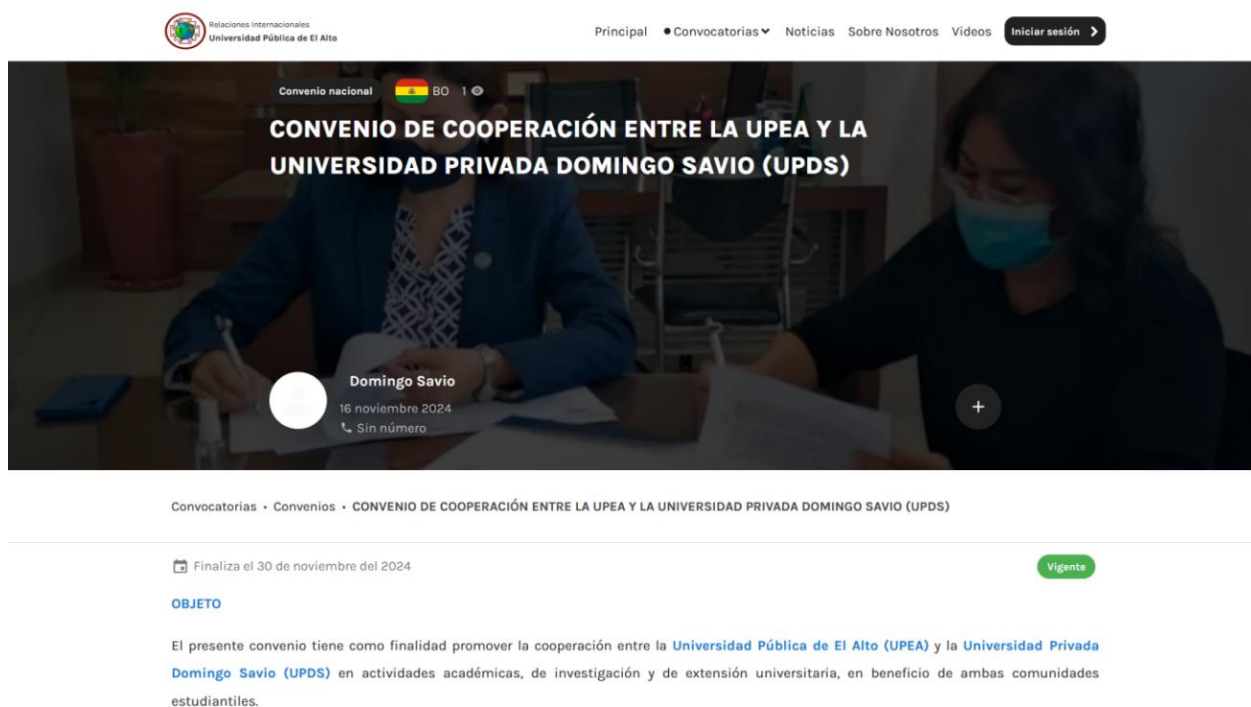
CAMPO		CONTENIDO
ENTRADAS	Titulo	CONVENIO DE COOPERACIÓN ENTRE LA UPEA Y LA UNIVERSIDAD PRIVADA DOMINGO SAVIO (UPDS)
	Pdf	/uploads/doc/convenio/file-1731815974846-953451242.pdf
	Imagen	/uploads/img/convenio/file-1731815974807-715185865.jpeg
	Descripcion	*contenido extenso*
	Tipo	nacional
	Estado	true

	Pais	BO
	Continente	SA
	Conteo	1
	Id	67396a2605bd5300a79bb5dd
	InstitucionId	67396a26f0656300a744b323
SALIDA	Agregar nuevo convenio	“Convenio agregado con éxito”
CONCLUSIÓN	Para agregar un convenio el sistema realizará la verificación de entrada en los campos de los datos.	Como resultado, el sistema guardará los campos agregados mediante la solicitud de agregar en la API y se realizará el almacenado de archivos como ser imágenes y documentos.

El resultado es una nueva publicación, en nuestro caso es un convenio creado de manera ficticia, el cual se realiza de manera exitosa y será visible para cualquier visitante de la página en el módulo de vista de convenios.

Figura 81

Publicación Activa del Convenio Creado



4.3.3. Pruebas de Stress

Las pruebas de Stress evalúan el rendimiento del sistema ante carga de solicitudes masivas mediante usuarios ficticios y con el uso de herramientas encargadas para este paso.

En esta prueba se utilizó un entorno en la nube en Vercel, el cual simula al sistema tradicional desplegado en los servidores propios. En donde nos proporciona recursos limitados para poder lograr nuestro objetivo de sobrecarga con recursos mínimos. A continuación mostramos la configuración empleada para realizar las pruebas simulando la cantidad de usuarios, solicitudes y tiempos de respuesta usando Postman.

Para esta prueba usaremos los siguientes parámetros:

Tabla 26
Parámetros Utilizados para la Prueba de Stress

PARÁMETRO	VALOR
Número de usuarios	40 usuarios virtuales
Tiempo de duración	1 minuto
Tipo de prueba	Prueba fijada (usuarios fijos)
URL de prueba	/convenios/buscar

Ejecutamos los parámetros usando Postman durante 1 minuto de manera fija para los 40 usuarios y obtener el promedio de solicitudes por segundo y el tiempo que se tarda en obtener la pantalla de la ruta de búsqueda de convenios.

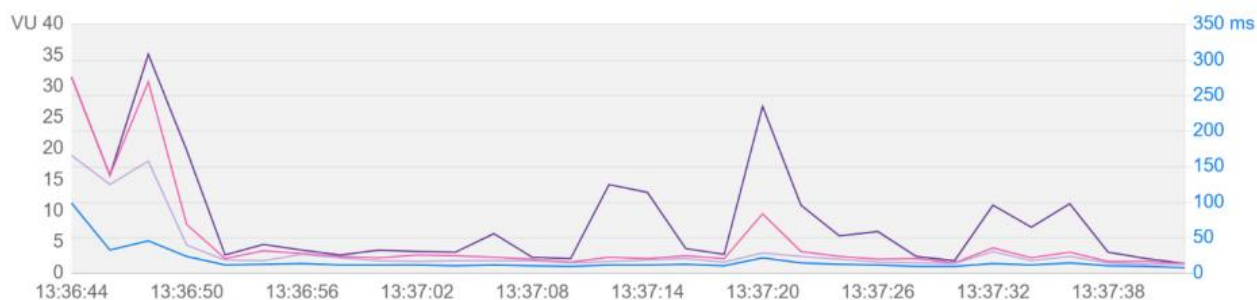
4.3.3.1. Resultados

Estos fueron los resultados proporcionados por Postman tras la aplicación de los parámetros.

Tabla 27
Parámetros Utilizados para la Prueba de Stress

REPORTE	RESULTADO
Total de solicitudes realizadas	1884 totales
Solicitudes por segundo	26,65 solicitudes/segundo
Tiempo de respuesta promedio	14 mili-segundos
URL de prueba	/convenios/buscar
Solicitud con respuesta rápida	5 mili-segundos

Solicitud con respuesta tardía	308 mili-segundos
Porcentaje de error	0%

Figura 82*Solicitudes por Segundo Durante un Minuto***Figura 83***Tiempo de Respuesta Durante un Minuto*

Los resultados nos muestran que el tiempo de respuesta del sistema es de 14 mili-segundos, entrando en el rango ideal de respuesta para los sistemas web que es de 5 - 20 mili-segundos.

Ahora lo probaremos con 100 usuarios usando la ruta principal del sistema "/", teniendo el siguiente resultado.

Figura 84*Resultado para la Ruta Principal con 100 Usuarios***Test setup**

Virtual users	Start time	Load profile
100 VU	Nov 5, 14:33:22 (GMT-4)	Fixed
Duration	End time	Environment
1 minute	Nov 5, 14:34:32 (GMT-4)	-

1. Summary

Total requests sent	Throughput	Average response time	Error rate
3,086	44.43 requests/second	30 ms	0.00 %

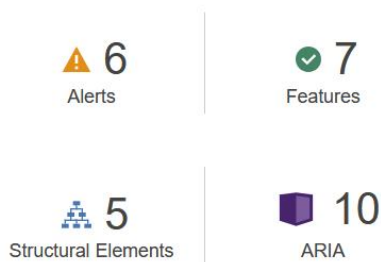
Tenemos un resultado promedio de respuesta de 30 mili-segundos y 44,43 solicitudes por segundo, teniendo un sistema con disponibilidad asegurada ante solicitudes y cargas de usuario constantes.

4.3.4. Pruebas de Accesibilidad

Decidiremos nuestro nivel de conformidad según las Pautas de Accesibilidad para el Contenido Web (WCAG) seleccionaremos la de nivel AA (intermedio), recomendado como estándar de sitios web públicos.

Seleccionaremos nuestras páginas de muestra que vamos a utilizar para esta prueba como ser: página de inicio, contenido que es visible para los usuarios y luego mostrar sus resultados y ver las alertas más comunes y su respectiva solución de accesibilidad.

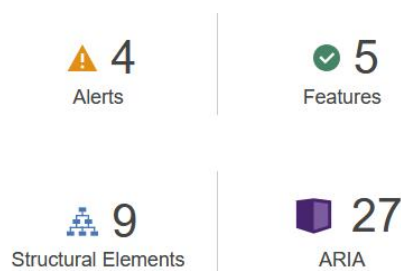
Figura 85*Análisis de Accesibilidad para la Página de Inicio*



Nota. Resultados en las pruebas de accesibilidad de la ruta de inicio, obteniendo solo alertas como ser: aria-label, Wave, 2024.

Figura 86

Análisis de Accesibilidad de Contenido



Nota. Resultados para el contenido de ejemplo en Becas, las alertas fueron relacionadas con el contraste de las imágenes, Wave, 2024

Antes de pasar a los formularios explicaremos las observaciones, agregamos la solución a los aria-label para la lectura de personas no videntes.

Figura 87

Aria-label, cabecera de ejemplo para becas

```
<Titulo
  lang="es" aria-label="Título de la beca" variant="h1" sx={{
    position: 'absolute',
```

Luego obtuvimos la observación de contraste en ciertas imágenes de las publicaciones como ejemplo la siguiente imagen de portada de un convenio.

Figura 88*Imagen de Portada de Convenio*

 **Contrast Errors**
Very low contrast

Nota. La portada del convenio tiene como alerta el bajo contraste de imagen, por lo que marca un error.

La solución a esta observación general en casi todo el sistema, lo hicimos con la ampliación del mismo a través de un modal extra y fácil de detectar mediante la puesta del cursor sobre la imagen misma.

Figura 89*Modal de Muestra para las Portadas de Publicaciones*

Además de pruebas de accesibilidad nos mencionan la experiencia de usuario a través de dispositivos móviles, requiriendo un diseño responsivo para el administrador como para los visitantes.

Figura 90
Responsividad del Sistema para Celulares y Tablet

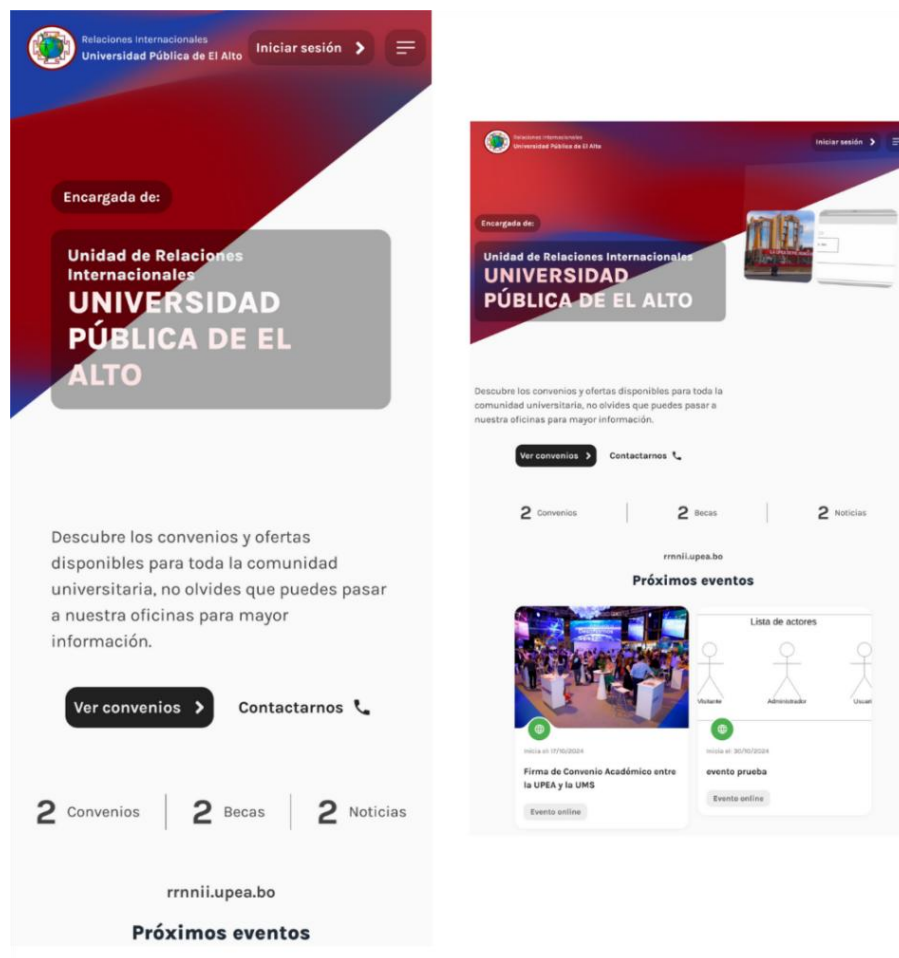


Figura 91
Responsividad del Sistema para Computadoras

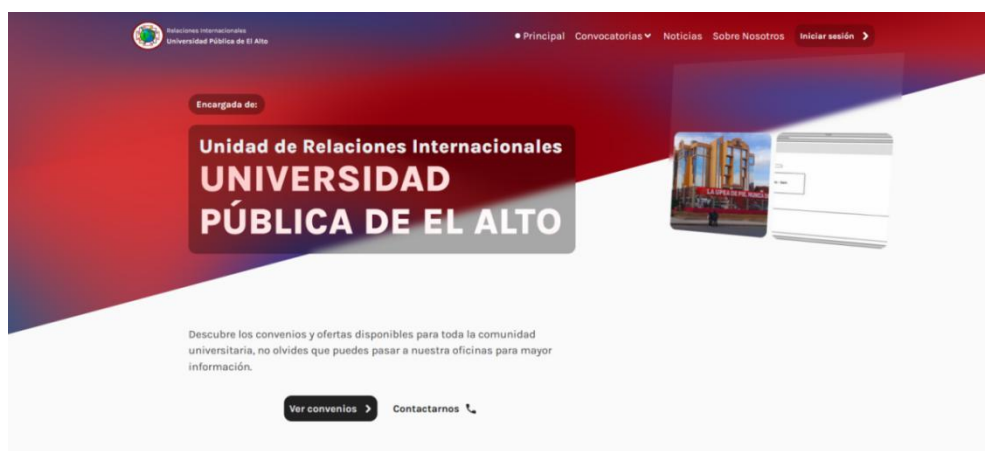


Figura 92
Acceso Dashboard desde Movil

The screenshot shows a mobile application interface for creating an event. At the top, there is a header with the logo of 'Relaciones Internacionales Universidad Pública de El Alto' on the left, a hamburger menu icon in the center, and a green circular logo with 'BNB' on the right. Below the header, the form is titled 'Titulo del evento' and has a text input field. Underneath is the 'Descripción' section, which includes a rich text editor with options for 'Normal', bold (B), italic (I), underline (U), strikethrough (ABC), and quote ("). Below the description is a 'Modalidad' dropdown menu currently set to 'Online'. The 'Comienzo' section has a date input field with the placeholder 'dd/MM/yyyy' and a calendar icon. Finally, there is a 'Link de acceso' text input field. The entire form is enclosed in a light gray border.

4.4. ANÁLISIS DE COSTOS

4.4.1. INTRODUCCIÓN

Cuando se desarrolla un software, algo que se debe tomar en cuenta es el esfuerzo proporcionado para su creación, lo cual es proporcional a la inversión económica que será utilizado. El modelo que estimación de costos para este propósito será COCOMO II, en donde tiene las métricas de esfuerzo y costos.

4.4.2. USO DE COCOMO II

Usar COCOMO II nos ayudará a realizar la estimación de costos de manera clara y detallada a los proyectos de software, para ello, COCOMO II toma en cuenta

las líneas de código utilizados en el proyecto para poder estimar el proyecto con una precisión concisa, el cual facilita la planificación en la asignación de recursos.

COCOMO II tiene diversas maneras de elección para estimar costos, en donde lo dividimos en 3 modelos, el modelo orgánico, el modelo semi acoplado y el modelo empotrado.

A continuación detallamos un poco más acerca de los modelos que se utilizan en COCOMO II.

Tabla 28
Tipo de Modelo COCOMO II

MODELO	DETALLES
Orgánico	Contiene a proyectos que ocupen menos de 5000 líneas de código. Viene centrado en el proceso de los datos y que tengan transacciones simples.
Semi acoplado	Contiene proyectos que abarquen un tamaño intermedio, con una mayor complejidad y restricciones.
Empotrado	Contiene proyectos complejos, en donde se promueve una mayor innovación y poca experiencia en la creación del mismo.

Estos mismos presentan constantes que van acorde a los modelos que se vayan a seleccionar.

Tabla 29
Constantes Aplicados a los Tipos de Modelo COCOMO II

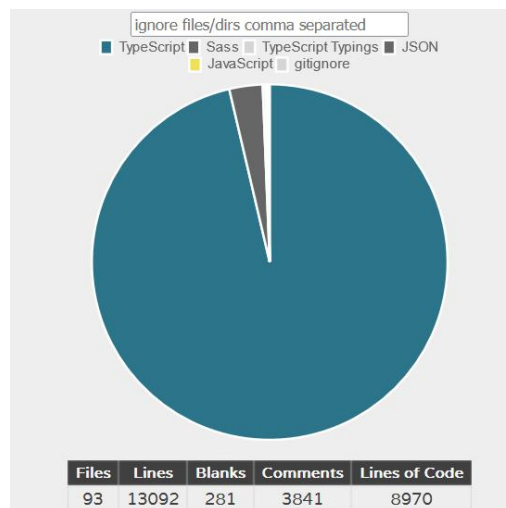
MODELO USADO	CONSTANTES			
	A	B	C	D
Orgánico	2.4	1.05	2.5	0.38
Semi acoplado	3	1.12	2.5	0.35
Empotrado	3.6	1.2	2.5	0.32

A la hora de estimar el esfuerzo, es necesario la variable KLDC (líneas de código en miles), con el que podremos evaluar el trabajo implementado, lo cual esta métrica nos será efectiva para poder medir el esfuerzo de una mejor forma.

Al momento de elegir las líneas de código utilizados, podemos utilizar ecuaciones para la medición o herramientas de medición que encontramos disponibles en la web. En nuestro caso, utilizaremos una herramienta en línea llamada Codetabs. A continuación mostramos los resultados obtenidos.

Figura 93

Líneas de Código del Sistema



Nota. Medida de las líneas de código LOC del proyecto, Codetabs, 2024, fuente: Codetabs (sistema de medición de Líneas de Código)

Si lo interpretamos en miles obtendríamos el siguiente resultado:

KLDC = 8,9

Al obtener los resultados, nos indica que el sistema pertenece al modelo de Semi acoplado y es en cual nos basaremos en el uso de las constantes en las próxima ecuación a utilizar y que nos brinda COCOMO II.

Para continuar con la estimación del costo y el esfuerzo, vamos a evaluar el sistema con las características del factor de ajuste, y luego continuar con el promedio obtenido con el valor de los atributos calculados.

Tabla 30
Atributos Seleccionados como Valor de Ajuste

ATRIBUTOS	MUY BAJO	BAJO	NORMAL	ALTO	MUY ALTO	EXTRA ALTO
Atributos de software						
Complejidad	0,75	0,88	1	1,15	1,40	
Volumen de la base de datos		0,94	1	1,08	1,16	
Fiabilidad	0,7	0,85	1	1,15	1,3	1,65

Atributos de hardware						
Restricción en el uso de la memoria virtual			1	1,06	1,21	1,56
Restricción en el tiempo de ejecución			1	1,11	1,3	1,66
Volatilidad de máquina virtual	0,87		1	1,15	1,3	
Tiempo de respuesta del servidor	0,87		1	1,07	1,15	
Atributos del personal						
Capacidad de análisis	1,46	1,19	1	0,86	0,71	
Experiencia en desarrollo	1,29	1,13	1	0,91	0,82	
Calidad de los programadores	1,42	1,17	1	0,86	0,7	
Experiencia en máquinas virtuales	1,21	1,1	1	0,9		
Experiencia en el lenguaje	1,14	1,07	1	0,95		
Atributos del proyecto						
Restricciones en el tiempo de desarrollo	1,24	1,1	1	0,91	0,82	
Utilización de herramientas de software	1,24	1,1	1	0,91	0,83	
Técnicas actualizadas de programación	1,22	1,08	1	1,04	1,1	
TOTAL			0,85			

Las características presentadas en la tabla se las conoce como FAE, los cuales se obtienen con el promedio de los valores seleccionados en la tabla.

Valor del resultado FAE:

$$\text{FAE} = 0,85$$

Seleccionamos los valores de las constantes, en donde son nuestras coeficientes que nos servirán para reemplazar en las ecuaciones que usaremos, el cual en este caso, nos enfocaremos en el modo orgánico. Por lo cual los valores serán:

$$A = 3 \quad B = 1,12 \quad C = 2,5 \quad D = 0,35$$

4.4.2.1. Esfuerzo aplicado al software

Los coeficientes seleccionados serán utilizados en el cálculo del esfuerzo y que se ajustan con las características seleccionadas anteriormente.

Ecuación 16 Esfuerzo Aplicado al Software

$$E_f = A * KLDC^B * \text{FAE} \tag{16}$$

En donde:

A = Coeficiente de COCOMO II = 3

B = Coeficiente de COCOMO II = 1,12

FAE = Factores de atributos de esfuerzo = 0,85

KLDC = Líneas de código en Miles = 8,9

Ef = Esfuerzo aplicado al proyecto

Reemplazando:

$$Ef = 3 * 8,9^{1,12} * 0,85$$

$Ef = 29,5$

4.4.2.2. Tiempo de desarrollo del sistema

Para calcular el tiempo de desarrollo del sistema necesario se hará el uso de los resultados previamente calculados.

Ecuación 17 Tiempo Requerido para Desarrollar el Sistema

$$Tdev = C(Ef)^D, \text{ calculado en meses} \quad (17)$$

En donde:

C = Coeficiente de COCOMO II = 2,5

D = Coeficiente de COCOMO II = 0,35

Ef = Esfuerzo incorporado en el desarrollo del software = 29,5

Tdev = Tiempo de desarrollo aplicado al sistema

Reemplazando:

$$Tdev = 2,5 * (29,5)^{0,35}, \text{ calculado en meses}$$

$$T_{dev} = 8,1 \text{ meses}$$

4.4.2.3. Cantidad de Desarrolladores a Incorporar

Basándonos en el esfuerzo y el tiempo de desarrollo, se puede sacar la cantidad de desarrolladores necesarios para el sistema.

Ecuación 18 Número de personas Necesarias para el Proyecto

$$Ps = \frac{Ef}{T_{dev}}, \text{ calculado en personas} \quad (18)$$

En donde:

Ef = Esfuerzo requerido para el sistema = 29,5

T_{dev} = Tiempo de desarrollo requerido = 8,1

Ps = Cantidad de personas para el proyecto

Reemplazando:

$$Ps = \frac{Ef}{T_{dev}}, \text{ calculado en personas}$$

$$Ps = 3$$

4.4.2.4. Costo total

Calcularemos el costo del sistema utilizado para este proyecto. Dado el salario mínimo nacional en Bolivia para el 2024, está en los 2500 Bs., y el cual nos sirve

como base para tomar en cuenta en el proyecto. La estimación proporcionada por COCOMO II es la teoría que nos propone el mismo. Llevando el proyecto a la realidad de optimización de costos y realistas tomaremos el ejemplo de 1 desarrollador que se enfocó en la realización del proyecto como es en nuestro caso dando así: $Ps = 1$.

Ecuación 19 Costo Total del Proyecto

$$Ct = Smensual * Ps * Tdev \quad (19)$$

En donde:

$Smensual$ = Sueldo mensual mínimo nacional de Bolivia

Ps = Cantidad de personas involucradas en proyecto = 1

$Tdev$ = Tiempo de desarrollo requerido en meses

Ct = Costo total del proyecto

Reemplazando:

$$Ct = 2500 * 1 * 8,5$$

$Ct = 21,250 \text{ Bs.}$

Viendo los resultados obtenidos se concluye que el proyecto necesitará de 1 programador que trabaje en un periodo de tiempo de 8,5 meses, donde recibe alrededor de 2500. Bs. mensuales, llevando un gasto total para el proyecto de

21,250Bs., concluyendo con el cálculo de los costos del software usando COCOMO II.

Tabla 31
Estimación Aproximada Usando COCOMO II

RESUMEN	
Número de desarrolladores involucrados en el proyecto	1
Tiempo de desarrollo	8,5 meses
Coste total del proyecto	21,250 Bs

CAPÍTULO V: CONCLUSIONES Y RECOMENDACIONES

**INGENIERÍA
DE SISTEMAS**
UNIVERSIDAD PÚBLICA DE EL ALTO



5. CONCLUSIONES Y RECOMENDACIONES

5.1. CONCLUSIONES

El proyecto desarrollado y aplicado en la unidad de relaciones internacionales tiene como conclusión el logro del objetivo principal “**Desarrollar un sistema integral de gestión de información para la Unidad de Relaciones Internacionales de la Universidad Pública de El Alto, con el fin de optimizar los procesos administrativos y facilitar la difusión de convenios y sus beneficios derivados**”, cumpliendo el propósito que tiene en cuanto a la publicaciones de información desglosadas a partir de los convenios como ser: becas, eventos, noticias, pasantías teniendo el respaldo necesario y permitir a que los universitarios tengan la capacidad de asistir como interesados en ciertas becas que vean conveniente; así también, la información acerca de la unidad en donde le permite tener la presencia ante nuestra comunidad.

En cuanto a nuestro proyecto de **SISTEMA INTEGRAL PARA LA GESTIÓN DE RELACIONES NACIONALES E INTERNACIONALES** tiene cumplido los siguientes objetivos específicos como ser:

- El desarrollo de un entorno web que permite la publicación de becas y que los interesados puedan visualizarlo de una manera más sencilla desde cualquier dispositivo.
- Se modularizó las distribuciones que vienen a partir de los convenios como ser las becas, pasantías, eventos, noticias y así mismo el convenio en general.

- El sistema cuenta con la capacidad de almacenar los archivos que sirven de respaldo de cada beca, así también el contenido multimedia generado.
- El sistema pasó a una nueva interfaz actualiza a su anterior versión, teniendo una mejor respuesta e cuanto a la eficiencia del mismo.
- El sistema puede realizar el registro de nuevos participantes que tienen por objetivo la participación en una de las becas ofrecidas por la universidad.

5.2. RECOMENDACIONES

- Es recomendable que los usuarios sepan resguardar sus contraseñas en un lugar seguro para evitar problemas.
- El administrador debe tener en cuenta que la información proporcionada sea gestionada y respaldada por parte de la unidad encargada para evitar problemas a futuro.
- No sobrecargar el sistema con archivos en formato vídeo que ocupen demasiado espacio dentro del servidor de archivos.
- Que los futuros usuarios reciban la capacitación del personal actual para el usar el sistema.

BIBLIOGRAFÍA

Adhemar. (2020). *Prisma, un toolkit para bases de datos (¿ORM?) para TypeScript y Node.js*

<https://dev.to/denispixi/prisma-un-toolkit-para-bases-de-datos-orm-para-typescript-y-node-js-3g9>

ADIF. (s.f.). *Vesta - Soteria - JSON Web Token*.

https://contrataciondelestado.es/wps/wcm/connect/86dbf55d-3fad-4c8f-9ce7-5850ac8abccf/002_0000636157_VestaSoteriaJsonWebTok.PDF?MOD=AJPERES

Agesic. (2022). *Metodología de Evaluación de accesibilidad*.

<https://www.gub.uy/agencia-gobierno-electronico-sociedad-informacion-conocimiento/sites/agencia-gobierno-electronico-sociedad-informacion-conocimiento/files/documentos/publicaciones/Metodologia%20de%20evaluacion%20de%20accesibilidad.pdf>

Aplyca. (2022). *NextJs ¿el futuro de la web?*.

<https://www.aplyca.com/blog/nextjs-el-futuro-web-que-es-nextjs>

Arsys. (s.f.). *Axios Javascript: analizamos las características de este ligero cliente HTTP*.

<https://www.arsys.es/blog/axios>

Aruquipa Coronel, J. L. *Sistema web integrado de administración y control de bibliotecas*. [Proyecto de grado, Universidad Pública de El Alto]. Archivo digital

<https://repositorio.upea.bo/jspui/handle/123456789/225>

C. E. Gómez, C. A. Candela, and L. E. Sepúlveda, "Seguridad en la configuración del Servidor Web Apache" INGE CUC, vol. 9, no. 2, pp. 31–38, 2013.

<https://revistascientificas.cuc.edu.co/ingecuc/article/view/3/72>

Cabezas. (1998). *¿Qué es una pasantía?*, IV Jornadas de Psicología Universitaria.

<https://psico.edu.uy/sites/default/files/que%20es%20una%20pasantia.pdf>

Castelblanco (2011). *Eventos académicos*.

<https://maguieventos.blogia.com/2011/061502-eventos-academicos.php>

Castro (2015). *Tipo de Becas Universitarias para Estudiantes*.

<https://es.scribd.com/document/287296773/Tipos-de-Becas-Universitarias-Para-Estudiantes>

Cathalifaud. (1998). *“Introducción a los Conceptos Básicos de la Teoría General de Sistemas”*, Cinta de Moebio, vol 3.

<https://www.redalyc.org/pdf/101/10100306.pdf>

CIDI (2013). *Manual de procedimientos de los programas de becas y capacitación de la organización de los estados americanos.*

<https://www.oas.org/es/becas/ManualdeProcedimientos.pdf>

Domínguez Chávez, J. (2019). *Fundamentos de PostgreSQL*. IEASS.

Estrada Castillo, C. M. (2022). *Diseño de un Sistema web para el control de Currículum Vitae Electrónico de personal docente basado en una arquitectura orientada a servicios (API REST)*. RITI Journal, 28.

<https://dialnet.unirioja.es/servlet/articulo?codigo=8336547>

Fuentes Vargas, J. (2021). *Plan de mejoramiento en el sistema de información y atención al usuario institucional de una institución prestadora de servicios de salud Córdoba* [Trabajo de grado, Universidad de Córdoba]. Archivo digital

<https://repositorio.unicordoba.edu.co/entities/publication/2fe75aa6-77c8-483b-bd73-4eeb9f3ad72c>

García Grandes. (s.f.). *Introducción a Express.js: conexión a una base de datos PostGIS*

<https://mappinggis.com/2024/03/introduccion-a-express-js-conexion-a-una-base-de-datos-postgis/>

Gaytan A. (2016). *Un sistema Integral de Información*.

<https://es.scribd.com/document/312772883/Un-Sistema-Integral-de-Informacion>

Glencore, J. M. (2012). *The art of software testing*. John Wiley & Sons.

Geeksforgeeks. (2024) *React Suite*.

<https://www.geeksforgeeks.org/react-suite/>

González Sánchez. (2024). *Diseña Aplicaciones Web con Material UI*.

<https://community.listopro.com/disena-aplicaciones-web-con-material-ui/>

Gutiérrez. (s.f.) *¿Qué es un framework web?*.

<https://www.cssblog.es/guias/Framework.pdf>

GrupoACMS. (2022). *¿Qué es ISO 25000? Certificación de Calidad de Software*.

<https://grupoacms.com/blog/iso-25000-certificacion-de-la-calidad-del-software>

López. (2010). *Pruebas de Stress*.

<https://es.scribd.com/document/43521148/Pruebas-de-Stress>

Ian. (2011). *Ingeniería de Software*.

https://gc.scalahed.com/recursos/files/r161r/w25469w/ingdelsoftwarelibro9_compressed.pdf

IBNORCA. (2013). *Gestión de la seguridad de la información ISO/IEC 27001*.

IBNORCA.

Mamani Quispe, F. A. (2018). *Sistema web de administracion de recursos humanos*.

[Proyecto de grado, Universidad Mayor de San Andrés]. Archivo digital
<https://repositorio.umsa.bo/handle/123456789/17471>

Mamani Ato, F. (2020). *Sistema de información para el seguro social universitario*.

[Proyecto de grado, Universidad Pública de El Alto]. Archivo digital
<https://repositorio.upea.bo/jspui/handle/123456789/168?mode=full>

Mamani Condori, W. (2020). *Sistema web de control de seguimiento de servicios y gestión de clientes para la empresa Consultora Contadores Públicos & Auditores AYS S.R.L.* [Proyecto de grado, Universidad Mayor de San Andrés]. Archivo digital

<https://repositorio.umsa.bo/handle/123456789/27967>

Miss J. (2020). *Método Cocomo II*.

<https://pdfcoffee.com/metodo-cocomo-ii-pdf-5-pdf-free.html>

Molero Suarez. (2021). *React*.

https://imaster.academy/contenidos-tematicos/desarrolloweb/Unidad2/03_Que%20es%20React.pdf

Pachari Rodriguez, R. D. *Portal Web para la gestión de información administrativa y académica*. [Proyecto de grado, Universidad Pública de El Alto]. Archivo digital

<https://repositorio.upea.bo/jspui/handle/123456789/164>

Pérez Jiménez. (2019). *Qué es Sass: ventajas, desventajas y ejemplos de desarrollo.*

<https://openwebinars.net/blog/que-es-sass-ventajas-desventajas-y-ejemplos-de-desarrollo/>

Puerta González. (2014). *Desarrollo de una API para la descripción y gestión de Servicios Web REST.*

<https://core.ac.uk/download/pdf/61473672.pdf>

Roa, P., Morales, C., Gutiérrez, P. (2015). *Norma ISO/IEC 25000*, 3(2), <https://geox.udistrital.edu.co/index.php/tia/article/view/8373/11349>

Rozentals, N. (2017). *Mastering TypeScript. Packt Publishing.* <https://github.com/NedyUdombat/Free-Web-Development-eBooks/blob/master/book/TypeScript/Mastering%20TypeScript%20-%20Second%20Edition.pdf>

Ríos Salgado, S. (2013). *Aplicación de la metodología OpenUP en el desarrollo del sistema de difusión de gestión del conocimiento de la ESPE.*

<https://repositorio.espe.edu.ec/bitstream/21000/6316/1/AC-SISTEMAS-ESPE-047042.pdf>

Sommerville, I. (2011). *Software engineering 9th edition* (L. M. Cruz Castillo, Trad., 9ª ed.). Addison Wesley. (Original Work published 2011)

Skysnag. (2023). *¿Qué es bcrypt?*

<https://www.skysnag.com/es/blog/what-is-bcrypt/>

UTPL. (s.f.). *Guía de contratos y convenios.*

<https://procuraduria.utpl.edu.ec/SiteAssets/Documentos%20Relacionados/GU%C3%8DA%20DE%20CONTRATOS%20Y%20CONVENIOS%202015.pdf>

Valverde Ramos & Hernández-Mora de Fuentes. (2016). *TypeScript*

<https://blog.educalix.com/wp-content/uploads/2023/03/Manual-TypeScript.pdf>

Vera Tejena, G. N. (2022). *Desarrollo de un sistema web de gestión de nómina para la automatización de procesos del área de recursos humanos en la Empresa Reciclajes y Procesos Ecológicos S.A.* [Trabajo de grado, Universidad de Guayaquil]. Archivo digital

<https://repositorio.ug.edu.ec/handle/redug/60078>

Vessuri, (2024). *Autenticación y contro de Acceso Basado en Roles en Next.js*.

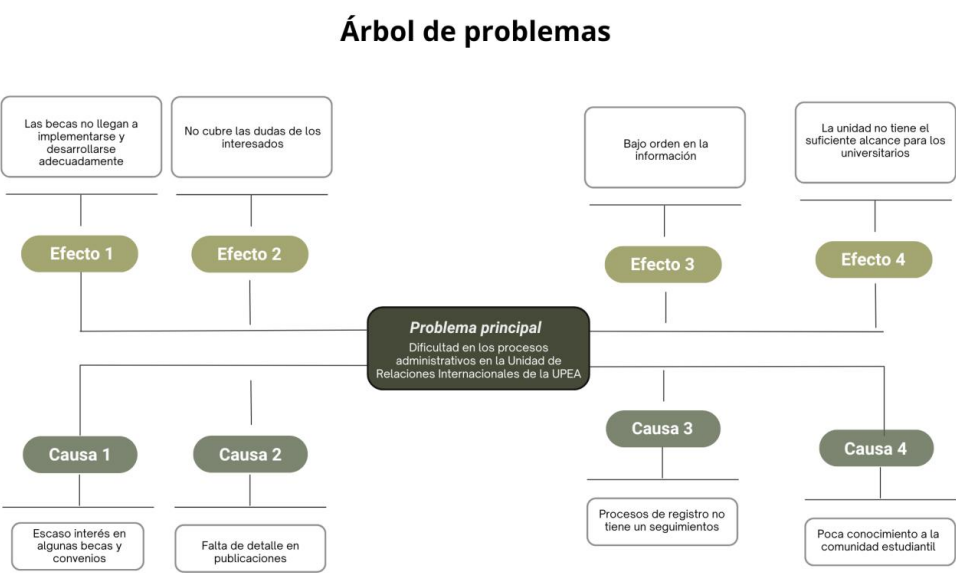
<https://7sabores.com/blog/autenticacion-y-control-de-acceso-basado-en-roles-en-nextjs>

VRAC, (2024). *Convenios Académicos*.

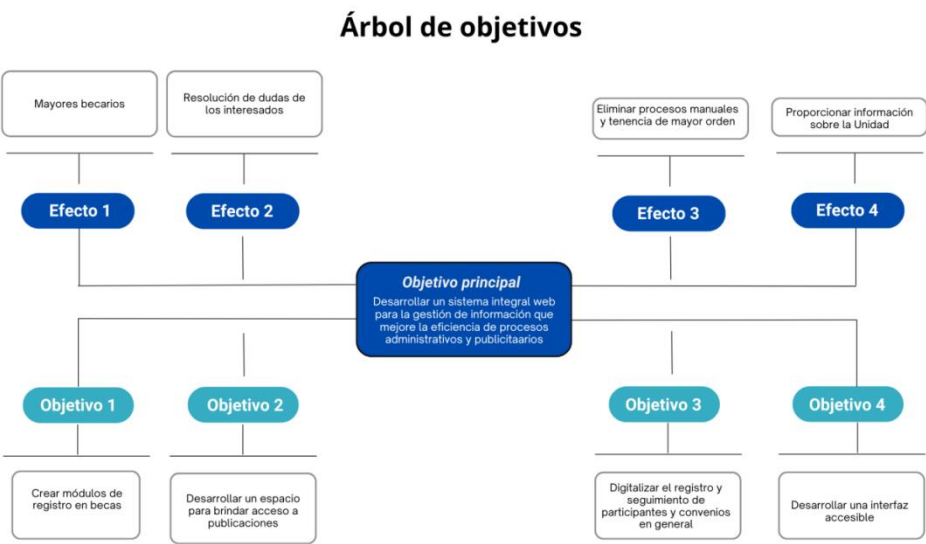
<https://viceacademica.univalle.edu.co/es/convenios-academicos2>

ANEXOS

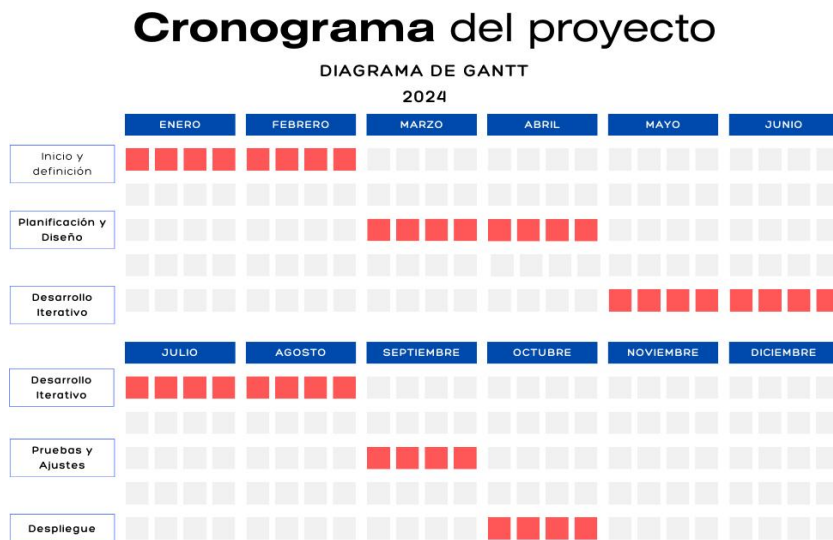
1. Árbol de Problemas del Proyecto



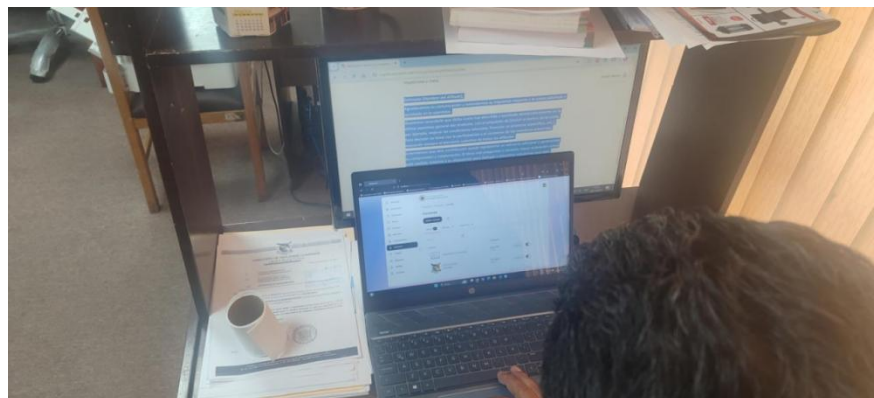
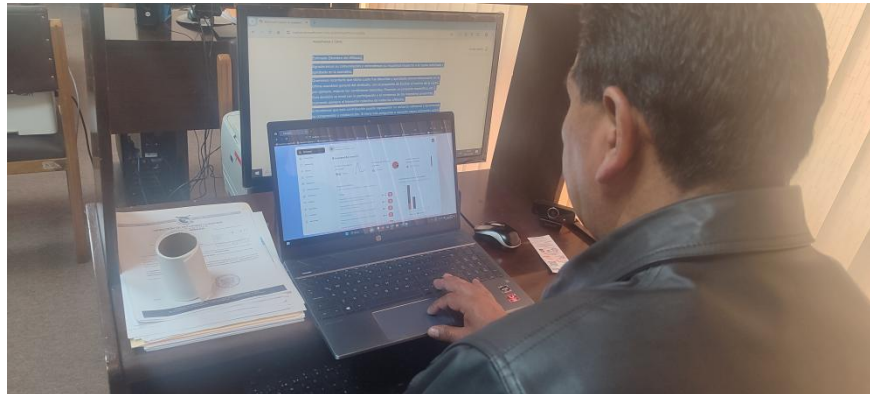
2. Árbol de Objetivos del Proyecto



3. Cronograma del Proyecto



4. Encargado de la Unidad Utilizando el Sistema



El Alto, noviembre de 2024

Señor:

Tutor metodológico

Lic. Ing. Helen Fanny Suntura Escobar

Carrera Ingeniería de Sistemas

Universidad Pública de El Alto

Presente. -

REF.: AVAL DE CONFORMIDAD DE PROYECTO DE GRADO

Distinguido Tutor:

Mediante la presente, informo sobre el avance del proyecto **"Sistema Integral para la Gestión de Relaciones Nacionales e Internacionales, Caso: Unidad de Relaciones Internacionales Universidad Pública de El Alto"**, a cargo del universitario Miguel Huayhua Condori (R.U. 200028431, C.I. 9190304).

En mi rol de tutor especialista, he acompañado al universitario en el proceso de estructuración y desarrollo del proyecto, con el objetivo de consolidar una solución tecnológica alineada a las necesidades de la unidad. A la fecha, el universitario ha logrado importantes avances y demostrado un compromiso notable con los objetivos establecidos.

Atentamente:

Lic. Ing. Yolanda Escobar Mancilla
Tutor Especialista

El Alto, noviembre de 2024

Señor:

Tutor metodológico

Lic. Ing. Helen Fanny Suntura Escobar

Carrera Ingeniería de Sistemas

Universidad Pública de El Alto

Presente. -

REF.: AVAL DE CONFORMIDAD DE PROYECTO DE GRADO

Distinguido Tutor:

Por medio de la presente, informo que he revisado el proyecto titulado "**Sistema Integral para la Gestión de Relaciones Nacionales e Internacionales**, Caso: **Unidad de Relaciones Internacionales Universidad Pública de El Alto**", desarrollado por el universitario Miguel Huayhua Condori (R.U. 200028431, C.I. 9190304).

Durante el proceso de revisión, he constatado un avance significativo en los objetivos propuestos y una adecuada implementación de los aspectos técnicos y metodológicos. El Universitario ha cumplido con las observaciones planteadas y ha demostrado compromiso en el desarrollo del proyecto.

Atentamente:

Lic. Santos Zenon Quispe Apaza
Tutor Revisor

El Alto, noviembre de 2024

Señor:

Director de Carrera

Lic. Ing. William Roque Roque

Carrera Ingeniería de Sistemas

Universidad Pública de El Alto

Presente. -

REF.: AVAL DE CONFORMIDAD DE PROYECTO DE GRADO

Estimado Señor Director:

Mediante la presente, me permito informar sobre el avance del proyecto titulado "**Sistema Integral para la Gestión de Relaciones Nacionales e Internacionales**, Caso: **Unidad de Relaciones Internacionales Universidad Pública de El Alto**", desarrollado por el universitario Miguel Huayhua Condori (R.U. 200028431, C.I. 9190304).

Como tutor metodológico, he acompañado al estudiante en la estructuración y cumplimiento de los lineamientos metodológicos del proyecto, asegurando la aplicación adecuada de los criterios académicos y técnicos requeridos. A la fecha, el universitario ha logrado avances satisfactorios, cumpliendo con los objetivos establecidos y demostrando un desempeño acorde con los estándares de la carrera.

Atentamente:

Lic. Ing. Helen Fanny Suntura Escobar
Tutor Metodológico



Universidad Pública De El Alto

Creada por ley 2115 de 5 de septiembre del 2000 y Autónoma por ley 2556 de 12 de noviembre del 2003

El Alto, Octubre de 2024

Lic. Ing. Helen Fanny Suntura Escobar

TUTOR METODOLÓGICO

TALLER DE GRADO II

Presente. -

REF.- AVAL DE CONFORMIDAD

Distinguido tutor metodológico:

Mediante la presente tengo a bien de comunicarlo mi conformidad del Trabajo de Grado:

TÍTULO: SISTEMA INTEGRAL PARA LA GESTIÓN DE RELACIONES NACIONALES E INTERNACIONALES

CASO: UNIDAD DE RELACIONES INTERNACIONALES UNIVERSIDAD PÚBLICA DE EL ALTO

MODALIDAD: PROYECTO DE GRADO

UNIVERSITARIO: MIGUEL HUAYHUA CONDORI

REGISTRO UNIVERSITARIO: 200028431

CÉDULA DE IDENTIDAD: 9190304 LP

De tal forma cabe recalcar que el **SISTEMA** satisface los requerimientos de la institución, de esta forma se dio cumplimiento de los objetivos del presente.

El presente SISTEMA fue **IMPLEMENTADO** satisfactoriamente en la institución.

En cuanto certifico, en honor a la verdad, para fines consiguientes del interesado para su defensa pública y evaluación correspondiente a la materia de Taller de Grado II, de acuerdo al reglamento vigente de la Carrera de Ingeniería de sistemas de Universidad Pública de El Alto.

Atentamente:



MANUAL DE USUARIO

Guía de uso del sistema de Relaciones Internacionales
de la Universidad Pública de El Alto

Miguel Huayhua Condori
Ingeniería de Sistemas 2024

1. Acceso al Sistema

Ingresa al modal de inicio de sesión mediante el botón de **iniciar sesión** en la parte superior derecha.

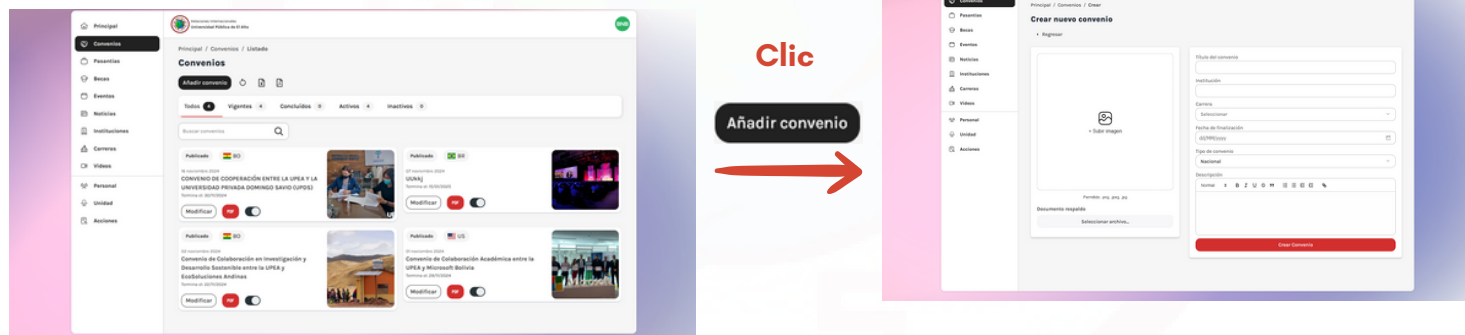


2. PUBLICAR INFORMACIÓN

2.1 Crear publicación

Acceda a la pestaña que desea entrar (convenios)

Agregue información del publicación



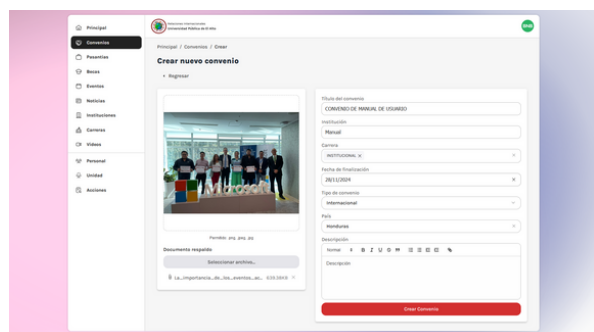
El sistema le pedirá campos de manera obligatoria

Documento respaldo

Seleccionar archivo...

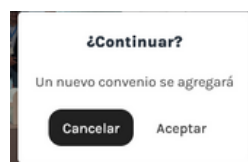
Respalde la publicación si es necesario

Título del convenio
Título no puede quedar vacío



Clic

Crear Convenio



Una vez seguro, el publicación se creará

Asegúrese de incluir toda la información que el sistema le pida

Convenio creado con éxito

2.2 Editar publicación

Use el modal para editar



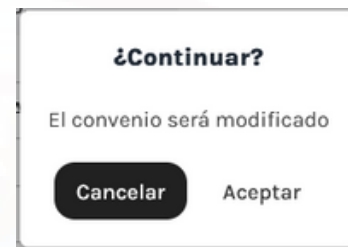
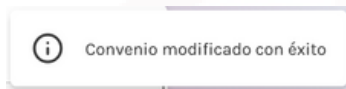
Encuentre la ficha de la publicación

Clic

Modificar

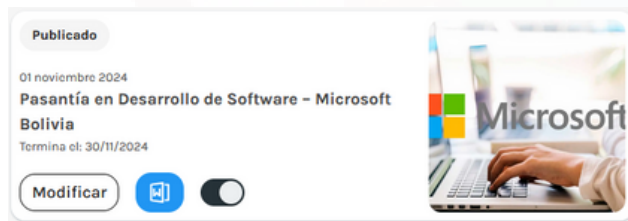
Clic

Modificar Convenio



Una vez seguro, el publicación se modificará

2.3 Activar / Desactivar Publicación



Encuentre la ficha de la publicación

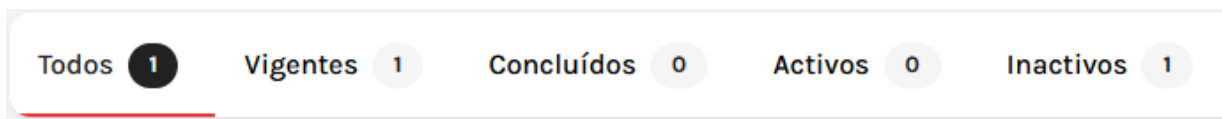
Clic



Pasantía Desactivada

La publicación estará desactivada y no será visible para los visitantes

2.4 Tipos de Listado de Publicaciones



Todas las publicaciones

Aquellas que no vencieron

Aquellas que si vencieron

Publicaciones con el estado activado

Publicaciones con el estado desactivado

2.4 Tipos de Botones de Publicaciones



3. VÍDEOS

3.1 Cargar Vídeo



3.2 Modificar Vídeo



3.3 Cambiar Estado de Vídeo

Título del video

17/11/2024
22:33:15

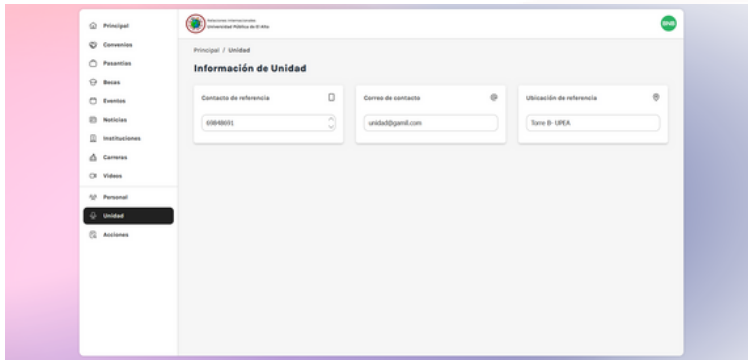
Encuentre su vídeo en la tabla



La vídeo estará desactivado
y no será visible para los
visitantes

4. INFORMACIÓN DE LA UNIDAD

4.1 Cambiar Información



Acceda a la pestaña de la Unidad

Contacto de referencia

69848691

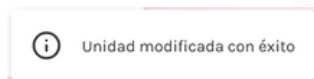
Correo de contacto

unidad@gamil.com

Ubicación de referencia

Torre B- UPEA

Puede cambiar desde contacto de
referencia, correo de la unidad y
ubicación de la unidad.



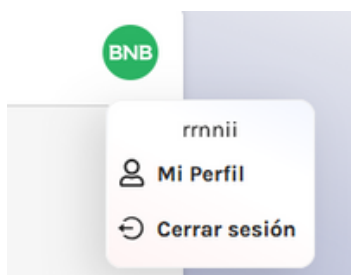
Guardar

Botón disponible cuando se cambie algo
en la pantalla

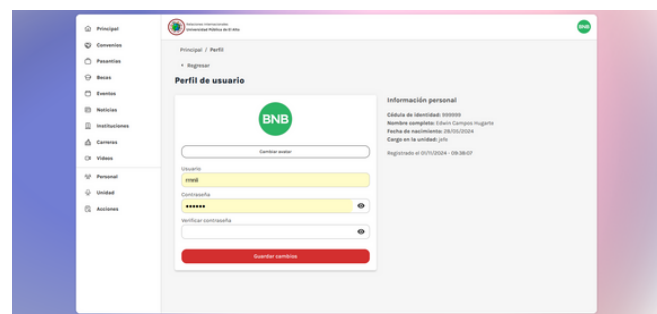
Clic

5. PERFIL DE USUARIO

5.1 Cambiar Información



Vaya a su perfil en la parte superior
derecha



Acceda a su perfil

El sistema te informará si
tu nombre de usuario está
en uso

Clic

Guardar cambios

Usuario

rrnnii

Usuario en uso



Avatar de usuario modificado con éxito

En el caso de modificar el avatar



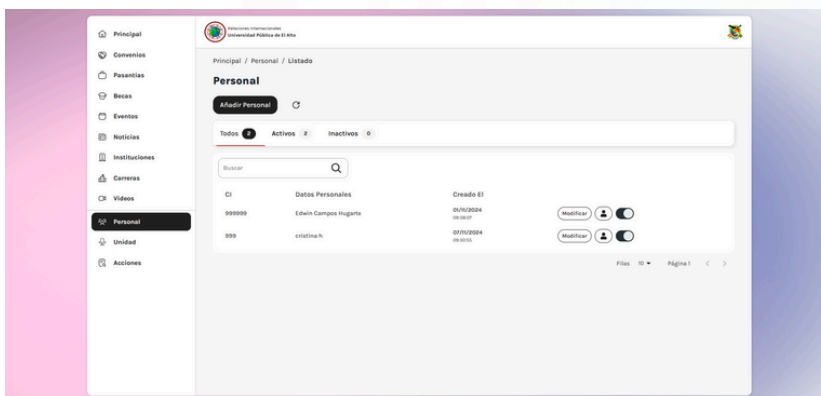
Usuario modificado con éxito

En el caso de modificar
la contraseña o nombre
de usuario

Al igual que un perfil convencional,
puede cambiar su contraseña, foto de
perfil y nombre de usuario

6. ADMINISTRACIÓN DE USUARIOS

6.1 Registrar Nuevos Usuarios



Clic

Añadir Personal

Acceda a la pestaña de **personal**

Completar los datos requeridos como
ser los datos de usuario y datos
personales

¿Continuar?

Se agregará un nuevo personal

Cancelar

Aceptar

Crear nuevo personal

Clic

Una vez seguro, el
personal se creará



Personal creado con éxito

6.2 Modificar Datos Personales

Edite los datos personales mediante el modal

12345678TESTING17/11/202423:18:14

Clic

Modificar

Una vez seguro, el personal se modificará

¿Continuar?

El personal será modificado

Cancelar

Aceptar

Clic

Guardar cambios

Modificar Personal

Nombres

Edwin

Apellido Paterno

Campos

Apellido Materno

Huarte

Cédula Identidad

999999

Fecha de nacimiento

28/05/2024

Cargo de Unidad

Jefe de unidad

Personal modificado con éxito

6.3 Modificar Credenciales de Usuario

Edite los datos de usuario mediante el modal

12345678TESTING17/11/202423:18:14

Clic

Modificar

¿Continuar?

El usuario será modificado

Cancelar

Aceptar

Clic

Guardar cambios

Modificar Usuario

Usuario

mmal

Contraseña

Verificar contraseña

Tipo de usuario

Administrador

Guardar cambios

Una vez seguro, el usuario se modificará

7. CONTROL DE BECAS

7.1 Gestionar Participantes de Becas

Entrar al apartado de visualización individual de becas

PublicadoBO56

01 noviembre 2024

Programa de Becas de Excelencia Académica – Banco Nacional de Bolivia (BNB)

Termina el: 20/03/2025

Modificar

4 participantes

Clic

Programa de Becas de Excelencia Académica – Banco Nacional de Bolivia (BNB)

Inicio

Participantes

Descripción del Programa:

El Banco Nacional de Bolivia (BNB) y la Universidad Pública de El Alto (UPA) han creado este programa para apoyar a estudiantes destacados en los campos de economía, finanzas, contabilidad y administración. Este programa ofrece becas completas y parciales a estudiantes con un excelente desempeño académico, brindándoles la oportunidad de mejorar su carrera sin preocupaciones financieras.

Fecha de publicación:

01 de noviembre del 2024 a las 10:00:00

Fecha de expiración:

20 de marzo del 2025

Organismo:

Walter Huayhuac Condori

Objetivo de la Beca

Sobre la beca

Postulantes 4

Generar listado

Buscar

Nombre	Registro Universitario	Cédula De Identidad		Revisado
Miguel Huayhua Condori	20002831	9190304	①	✓
LUIS	200	99	①	✓
JHBUJ	2300	874	①	✓
helen	20029884	9190304	①	✓

Postulantes 4

Entrar a la pestaña de postulantes

Listado de participantes con documentación revisada





MANUAL TÉCNICO

Guía para desarrolladores del sistema de Relaciones
Internacionales de la Universidad Pública de El Alto

Miguel Huayhua Condori
Ingeniería de Sistemas 2024

ÍNDICE DE CONTENIDO

INTRODUCCION 4

OBJETIVO 4

REQUISITOS DEL SISTEMA - VERSION 3.0 4

HERRAMIENTAS DE SOFTWARE UTILIZADAS PARA DESARROLLO DEL 7

SISTEMA 7

CONTENIDO TECNICO 7

1. MÓDULOS 7

1.1. Módulos Principales 7

2. GESTIÓN DE ARCHIVOS ESTÁTICOS Y CARGADOS 12

3. Cómo Codificar en NextJs 13

3.1. Uso de App Router 13

4. LIBRERÍAS UTILIZADAS 15

ÍNDICE DE TABLAS

Tabla 1	Tabla de Requisitos para Implementar el Software	5
Tabla 2	Requisitos Mínimos para Implementar el Software	6
Tabla 3	Herramientas para el Funcionamiento del Sistema	7
Tabla 4	Módulos Principales del Sistema	7
Tabla 5	Módulo de Inicio de Sesión - NextAuth	8
Tabla 6	Módulo de Publicaciones	9
Tabla 7	Módulos de la API	10
Tabla 8	Secciones del Módulo de Administración	11
Tabla 9	Gestión de Archivos Almacenados	12

INDICE DE FIGURAS

Figura 1	Estructura de un Proyecto en NextJs	13
Figura 2	Manejo de Ruta de tipo API	14
Figura 3	Principales Librerías en el Proyecto Next.js	15

INTRODUCCION

Este manual técnico proporciona una guía completa para el uso, instalación y mantenimiento del sistema de gestión de información desarrollado para la Unidad de Relaciones Internacionales de la UPEA. Diseñado con tecnologías modernas como Next.js, Prisma y React Suite, el sistema prioriza la accesibilidad, seguridad y escalabilidad, adaptándose a las necesidades específicas de la institución.

En las siguientes secciones, se detallan la estructura del sistema, las funcionalidades de cada módulo y las instrucciones para su configuración y operación. Además, se incluyen guías para resolver problemas comunes y realizar mantenimientos preventivos y correctivos, asegurando un manejo eficiente y sostenible del sistema.

OBJETIVO

Proporcionar una guía técnica del Sistema de Relaciones Internacionales de la Universidad Pública de El Alto.

REQUISITOS DEL SISTEMA - VERSION 3.0

REQUISITO	DESCRIPCIÓN	VERSIÓN REQUERIDA
Node.js	Entorno de ejecución de JavaScript necesario para el funcionamiento del sistema.	v18.0 o superior
Npm	Gestor de paquetes para instalar dependencias del proyecto.	v8.0 o superior
Next.js	Framework para aplicaciones React con SSR y SSG.	v14.0.0 o superior
Prisma	ORM para interactuar con bases de datos.	v5.0.0 o superior
PostgreSQL	Base de datos relacional utilizada por el sistema.	v13.0 o superior
NextAuth.js	Solución de autenticación para aplicaciones Next.js,	v4.24.7 o

	soporta múltiples proveedores.	superior
TypeScript	Lenguaje que extiende JavaScript con tipado estático.	v5.0.0 o superior
React	Biblioteca principal para construir interfaces de usuario.	v18.0 o superior
@mui/material	Biblioteca de componentes UI basada en Material Design.	v5.15.0 o superior
Multer	Middleware para la carga de archivos en el backend.	v1.4.5 o superior

Tabla 1

Tabla de Requisitos para Implementar el Software

USUARIO	REQUERIMIENTO MINIMO DE HARDWARE
Desarrollador	PARA DESARROLLO DEL SISTEMA ✓ Procesador: 2.1 GHz, AMD Ryzen 5, 4 núcleos ✓ Memoria RAM: 12 GB ✓ Disco Duro: 275 GB ✓ Otros: Tarjeta de red, controlador wifi, controladores de sonido y vídeo. PARA PRODUCCION ✓ Servidor: Dell PowerEdge (T140 o R240 u modelos similares) ✓ Procesador: Modelo Inten XON E-2224 o similares; características mínimas; 4 nucleos, 3.4 GHz ✓ Memoria RAM: 16 GB ✓ Disco Duro: 1 TB ✓ Otros: Tarjeta o antena de Red, fuentes de poder, teclado, mouse, monitor.
Cliente	PARA HACER USO DEL SISTEMA Computadora Personal, como primera opción:

- ✓ **Procesador:** 1 GHZ
- ✓ **Memoria RAM:** 512 MB
- ✓ **Disco Duro:** 8 GB
- ✓ **Otros (Dispositivo):** Tarjeta o antena de Red, fuentes de poder, teclado, teclado, mouse, monitor.

Teléfono Móvil, como segunda opción:

- ✓ **Procesador:** 1.2 GHZ
 - ✓ **Memoria RAM:** 1 GB
 - ✓ **Memoria interna:** 4GB
 - ✓ **Pantalla móvil:** LCD u OLED
-

Tabla 2

Requisitos Mínimos para Implementar el Software

USUARIO	REQUERIMIENTO MINIMO DE SOFTWARE
Desarrollador	PARA DESARROLLO Y PRODUCCION DEL “RRNNII” ✓ Sistema Operativo: Linux Ubuntu 22.04 ✓ Red: Conexión a la red local y a Internet.
Cliente	PARA HACER USO DEL “RRNNII” Computadora Personal, como primera opción: ✓ Sistema Operativo: Linux o Microsoft Windows Vista en adelante. ✓ Red: Conexión a internet. ✓ Navegador de Internet: Google Chrome, Firefox, Opera, Microsoft Edge, Safari. ✓ Privilegios: Según roles de usuario del sistema. Teléfono Móvil, como segunda opción: ✓ Sistema Operativo: Android 5.0 en adelante. ✓ Red: Conexión a internet. ✓ Navegador de Internet: Google Chrome, Firefox, Opera, Microsoft Edge, Safari. ✓ Privilegios: Según roles de usuario del sistema.

HERRAMIENTAS DE SOFTWARE UTILIZADAS PARA DESARROLLO DEL SISTEMA

Las herramientas de Software utilizadas para el desarrollo del “RRNNII”, se detalla en la siguiente tabla:

Tabla 3 Herramientas para el Funcionamiento del Sistema

SOFTWARE	DETALLE
Lenguajes de programación	TypeScript, JavaScript
Frameworks	Next.js
Lenguaje de marcado	JSX, HTML 5
Lenguaje de estilos	CSS 3, SCSS
Librería para gráficas	Chart.js, Recharts
Sistema de Gestión de Base de Datos	Prisma
Administrador de Base de Datos	pgAdmin, Prisma Studio

CONTENIDO TECNICO

A continuación, se describen brevemente cada una de las estructuras técnicas del “RRNNII” (versión 3.0).

1. MÓDULOS

Los controladores contienen las clases y métodos del “RRNNII” (versión 3.0); los cuales se describen a continuación:

1.1. Módulos Principales

Tabla 4

Módulos Principales del Sistema

MÓDULO	DESCRIPCIÓN
Inicio de Sesión	Gestiona la autenticación de usuarios mediante NextAuth, soportando diferentes métodos como credenciales
Publicaciones	Permite la creación, edición, visualización y eliminación de noticias, eventos, y contenido multimedia.
Administración	Panel de control para gestionar usuarios, roles, y entidades relacionadas como becas, convenios, y pasantías.
API	Proporciona endpoints para manejar solicitudes relacionadas con becas, eventos, usuarios, y más, construida en Next.js.

1.1.1. Módulo de Sesiones NextAuth

Este módulo gestiona la autenticación de los usuarios mediante NextAuth. También implementa sesiones con jwt para mayor seguridad y permite restringir el acceso a rutas según los roles definidos para cada usuario.

Tabla 5

Módulo de Inicio de Sesión - NextAuth

CARACTERÍSTICA	DESCRIPCIÓN
Autenticación	Utiliza NextAuth para la gestión de usuarios con opciones de login por credenciales y autenticación externa.
Sesiones	Implementa manejo de sesiones con jwt y soporte para tokens personalizados.

Protección de Rutas	Utiliza middleware para restringir acceso a rutas según el rol del usuario.
Base de Datos	Gestión de usuarios y roles utilizando Prisma y una base de datos relacional como PostgreSQL.

1.1.2. Módulo de Publicaciones

Este módulo permite a los usuarios crear, editar, visualizar y eliminar noticias, eventos y contenido multimedia. Incluye funcionalidades para subir imágenes y vídeos, además de mostrar placeholders animados mediante Skeletons mientras se carga el contenido.

Tabla 6

Módulo de Publicaciones

CARACTERÍSTICA	DESCRIPCIÓN
Convenios	Permite gestionar los convenios institucionales, incluyendo detalles como fechas, instituciones asociadas y objetivos.
Becas	Maneja la información sobre las becas disponibles, sus requisitos, y los procesos de aplicación.
Pasantías	Administra las oportunidades de pasantías, mostrando empresas asociadas, fechas, y plazas disponibles.
Eventos	Gestiona eventos institucionales, incluyendo su descripción, fechas, ubicaciones y participantes.
Noticias	Permite crear y publicar noticias relacionadas con las actividades y logros de la institución.
Vídeos	Facilita la subida y administración de contenido multimedia en formato de video para complementar las publicaciones.

1.1.3. Módulo API

Cada ruta base incluye subrutas específicas, como /crear, /listar, /modificar, /estado, entre otras, que corresponden a las acciones permitidas en el módulo. Esto garantiza una estructura organizada y fácil de extender ,puede variar el tipo de información que se quiere de cada tabla.

Tabla 7

Módulos de la API

MÓDULO	DESCRIPCIÓN	ruta base
Auth	Módulo para la autenticación y autorización de usuarios mediante NextAuth. Incluye inicio de sesión y sesiones.	/api/auth/
Beca	Maneja las operaciones relacionadas con las becas: creación, listado, modificación y administración de participantes.	/api/beca/
Carrera	Gestiona la información de las carreras, permitiendo crear, listar, modificar y actualizar su estado.	/api/carrera/
Convenio	Administra los convenios entre instituciones, incluyendo creación, listado, estado y detalles generales.	/api/convenio/
Dashboard	Proporciona estadísticas y métricas sobre las diferentes entidades del sistema, permitiendo consultas rápidas.	/api/dashboard/
Evento	Permite la gestión completa de eventos: creación, listado, modificación y actualización de estados.	/api/evento/
Institucion	Administra las instituciones asociadas, soportando operaciones como creación, modificación y listado.	/api/institucion/

Noticia	Gestiona el contenido de noticias, permitiendo crear, listar y actualizar su estado.	/api/noticia/
Pasantia	Maneja la información de pasantías, incluyendo la creación, modificación y su estado actual.	/api/pasantia/
Persona	Permite gestionar información personal como nombres, apellidos, identificación, y otros datos relacionados.	/api/persona/
Unidad	Administra las unidades organizacionales, incluyendo su creación, listado, y actualización de información.	/api/unidad/
Usuario	Gestiona los usuarios del sistema, soportando creación, modificación, eliminación y asignación de roles.	/api/usuario/
Video	Facilita la subida, listado, y gestión de contenido multimedia en formato de video.	/api/video/
Count	Endpoint para contar registros o métricas específicas, útil para estadísticas rápidas.	/api/count/

1.1.4. Módulo de Administración

El módulo de administración centraliza las operaciones y funcionalidades necesarias para gestionar de manera eficiente las principales entidades del sistema. Está diseñado para ser intuitivo y permitir el control de usuarios, roles y publicaciones. A continuación, se detallan sus secciones principales:

Tabla 8

Secciones del Módulo de Administración

SECCIÓN	DESCRIPCIÓN
---------	-------------

Gestión de usuarios	Permite administrar los usuarios del sistema, incluyendo la creación, modificación, eliminación y asignación de roles específicos.
Roles y permisos	Controla el acceso a diferentes funcionalidades del sistema según el rol asignado, asegurando un uso seguro y organizado.
Configuración de publicaciones	Proporciona herramientas para gestionar publicaciones como becas, convenios, pasantías, eventos, noticias y vídeos. Incluye funciones para crear, editar y eliminar.
Dashboard administrativo	Muestra estadísticas, gráficas dinámicas y métricas relevantes del sistema para facilitar el monitoreo general.
Reporte y análisis	Genera reportes detallados sobre las actividades del sistema, como registro de usuarios, interacciones y estados de las publicaciones.
Personalización	Incluye opciones para ajustar configuraciones generales, como el tema visual, las políticas del sistema y la información institucional.

2. GESTIÓN DE ARCHIVOS ESTÁTICOS Y CARGADOS

En el sistema, se gestionan dos tipos de archivos: archivos estáticos (assets) utilizados en la interfaz de usuario y archivos cargados por los usuarios. Los archivos estáticos se almacenan en la carpeta public/assets de Next.js, mientras que los archivos cargados se manejan mediante Express.js y Multer en la carpeta /uploads.

Tabla 9

Gestión de Archivos Almacenados

TIPO DE ARCHIVO	DESCRIPCIÓN	ruta/ubicación	TECNOLOGÍAS USADAS
-----------------	-------------	----------------	--------------------

Archivos Estáticos (Assets)	Archivos esenciales para el funcionamiento del sistema.	/public/assets (carpeta raíz de Next.js)	Next.js (public folder)
Archivos Subidos por Usuarios	Archivos cargados por los usuarios, gestionados con Express y Multer.	/uploads (carpeta generada por Express.js)	Express.js, Multer
Upload (Carga de Archivos)	Función para cargar archivos al servidor.	/uploads (carpeta generada dinámicamente)	Multer, Express.js
Delete (Eliminación de Archivos)	Función para eliminar archivos previamente cargados.	/uploads (carpeta de archivos)	Express.js, Multer

3. Cómo Codificar en NextJs

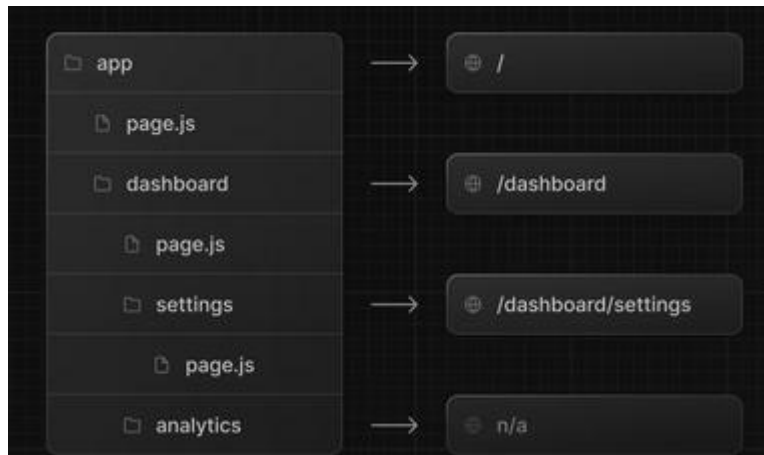
3.1. Uso de App Router

La estructura de carpetas dentro de la carpeta app en Next.js define cómo se gestionan las rutas y vistas. Cada carpeta dentro de app corresponde a una ruta específica de la aplicación.

app/

Figura 1

Estructura de un Proyecto en NextJs



El App Router de Next.js es una nueva forma de manejar las rutas y vistas de tu aplicación. Organiza las rutas en carpetas dentro de la carpeta `app`, donde cada carpeta representa una ruta de tu aplicación.

- `page.tsx` dentro de una carpeta define qué se muestra en esa ruta.
- `layout.tsx` se utiliza para definir la estructura común de la aplicación (por ejemplo, cabecera, barra lateral).
- Puedes tener rutas dinámicas con nombres entre corchetes, como `[id].tsx`, para manejar URLs personalizadas.

Esto permite crear rutas más fáciles de gestionar, con un renderizado eficiente y modular, y carga solo de lo necesario para cada vista.

Figura 2

Manejo de Ruta de tipo API



4. LIBRERÍAS UTILIZADAS

Esta tabla resume las librerías esenciales utilizadas en el proyecto para gestionar la interfaz de usuario, la autenticación, la interacción con bases de datos, el procesamiento de imágenes, y la gestión de formularios, entre otros. Cada librería tiene un papel crucial en el funcionamiento del sistema, optimizando el rendimiento, la escalabilidad y la experiencia del usuario.

Figura 3
Principales Librerías en el Proyecto Next.js

LIBRERÍA	DESCRIPCIÓN
Next	Framework principal para el desarrollo de aplicaciones React con características como SSR, SSG, y App Router.
React	Biblioteca principal para la construcción de interfaces de usuario.
React-dom	Proporciona el DOM para React, usado en la renderización de componentes.
Next-auth	Solución de autenticación para aplicaciones Next.js, soporta autenticación con múltiples proveedores.
@mui-material	Biblioteca de componentes de interfaz de usuario de Material Design para React.
Axios	Cliente HTTP para realizar solicitudes a APIs.
Prisma	ORM para interactuar con bases de datos, simplificando las consultas SQL.
React-hook-form	Biblioteca para gestionar formularios en React, enfocada en el rendimiento.
Dayjs	Librería para el manejo de fechas y horas, ligera y rápida.

@emotion/react	Librería de estilos CSS-in-JS para crear interfaces personalizadas en React.
@emotion/styled	Solución de styled-components para estilizar componentes con CSS-in-JS.
Recharts	Librería para crear gráficos interactivos y personalizables en React.
Rsuite	Conjunto de componentes UI de React para crear interfaces modernas.
Sharp	Librería para el procesamiento de imágenes (redimensionado, conversión, etc.).
Sass	Soporte para usar Sass (CSS preprocesado) en tu proyecto Next.js.
Typescript	Lenguaje que extiende JavaScript con tipado estático para un desarrollo más robusto.
Eslint	Herramienta de linting para mantener la calidad del código y evitar errores.
@iconify/react	Librería para usar iconos SVG en tu aplicación React.
Use-file-picker	Componente para seleccionar archivos en la interfaz de usuario.