

Área de Conocimiento de Tecnología de la
Información y Comuniación

Desarrollo de un sistema web para la evaluación de daños y el análisis de necesidades en las universidades de Nicaragua

**Trabajo Monográfico para optar al título de
Ingeniero de Sistemas / Ingeniero en Computación**

Elaborado por:

Br. Jose Angel
Guerrero

Carnet: 2017-0497U

Br. José Manuel
Dubón Monge

Carnet: 2015-0059U

Br. Marcos Antonio
Zúñiga Tapia

Carnet: 2014-1442U

Tutor:

Phd. Denis Eduardo
Hernández García



Área de Conocimiento de Tecnología de la Información y Comunicación

Managua, 26 de noviembre 2024

Br. José Ángel Guerrero 2017-0497U (Sistemas)
Br. José Manuel Dubón Monge 2015-0059U (Sistemas)
Br. Marcos Antonio Zúniga Tapia 2014-1442U (Computación)
Egresados Programa académico Ingeniería de Sistemas / Ingeniería en Computación
Sus manos.-

Estimados Egresados:

Reciban cordiales saludos y éxito en sus actividades.

Por medio de la presente, les comunico la aprobación del Protocolo de trabajo monográfico titulado: **“Desarrollo de un sistema web para la evaluación de daños y el análisis de necesidades en las universidades de Nicaragua”**, el cual cumple con los requisitos establecidos en el capítulo II de la normativa de trabajos monográficos de la UNI como forma de culminación de estudios, por lo que queda oficialmente aprobado por esta Dirección.

El docente responsable de acompañarle en el proceso de desarrollo de su tema monográfico es la MSc. Patricia Lacayo Cruz.

A partir de la fecha de aprobación del protocolo monográfico, tienen un máximo de doce meses para presentar los documentos correspondientes para la coordinación del proceso de pre defensa.

Atentamente,

Msc. Claudia Lucía Benavidez Rugama
Directora Área de Conocimiento de
Tecnología de la Información y Comunicación



Cc MSc. Patricia Lacayo Cruz – Tutor
MSc. Cedrick DallaTorre Parrales – Secretario Académico
Archivo DACTIC.



Móvil: (505) 8588 8333



Recinto Universitario Simón Bolívar
Avenida Universitaria.
Managua, Nicaragua.
Apdo: 5595



www.uni.edu.ni



Managua, 24 de enero 2025

Br. José Ángel Guerrero 2017-0497U (Sistemas)
Br. José Manuel Dubón Monge 2015-0059U (Sistemas)
Br. Marcos Antonio Zúniga Tapia 2014-1442U (Computación)
Egresados Programa académico Ingeniería de Sistemas / Ingeniería en Computación
Sus manos.-

Estimados Bachilleres:

Reciban cordiales saludos y éxito en sus actividades.

De acuerdo a carta recibida con fecha 22 de enero, en donde hacen solicitud de cambio de tutor del tema monográfico titulado: **"Desarrollo de un sistema web para la evaluación de daños y el análisis de necesidades en las universidades de Nicaragua"**, tomando en consideración las justificaciones expuestas en la misma, tengo a bien comunicarles que se les autoriza dicho cambio, nombrando como Tutor al **MSc. Denis Eduardo Hernández García** para dar continuidad con el proceso de su trabajo monográfico.

Esperando el cumplimiento con respecto al tiempo establecido, les reitero mis saludos.

Atentamente,




Msc. Claudía Lucía Benavidez Rugama
Directora Área de Conocimiento de
Tecnología de la Información y Comunicación

CC MSc. Denis Eduardo Hernández García– Tutor
MSc. Hazzely Orozco Miranda – Secretaria académica.
Archivo.



Móvil: (505) 8588 8333



Recinto Universitario Simón Bolívar
Avenida Universitaria,
Managua, Nicaragua.
Apdo: 5595



www.uni.edu.ni

Managua, 04 de marzo del 2026

Msc. Claudia Lucía Benavidez Rugama
Directora de Área de Conocimiento de Tecnología
de la Información y Comunicación
Universidad Nacional de Ingeniería

Distinguida Msc Benavidez.

Mediante la presente hago constar que he revisado en calidad de tutor la monografía con título **Desarrollo de un sistema web para la evaluación de daños y el análisis de necesidades en las universidades de Nicaragua**, realizada por los Bachilleres José Ángel Guerrero, José Manuel Dubón Monge y Marcos Antonio Zuñiga Tapia, la cual cumple con los requisitos necesarios según el artículo 21 de la Normativa para los trabajos monográficos de la Universidad Nacional de Ingeniería, así mismo hago constar que se han incluido las recomendaciones dadas por el Jurado examinador durante la predefensa.

Por lo anterior doy mi autorización para que se proceda a la defensa de la misma por parte de los alumnos.

Sin más a que realizar mención, me despido deseándole éxito en sus funciones.

Atentamente,

Phd. Denis Eduardo Hernández García
Tutor de monografía
Profesor titular del Área de Conocimiento de
Tecnología de la Información y Comunicación

DEDICATORIA

Agradezco en primer lugar a Dios, por darme la fortaleza, sabiduría y perseverancia necesarias para culminar esta etapa tan importante de mi vida. A mis padres, por su apoyo incondicional, sus sacrificios y su confianza constante en mí. A todos los docentes que formaron parte de mi proceso académico, por compartir sus conocimientos y contribuir a mi crecimiento profesional. De manera especial, a nuestro tutor, por su orientación y acompañamiento durante el desarrollo de esta monografía. Finalmente, expreso mi sincero agradecimiento a Julia Margarita Trujillo Méndez y a su familia, quienes fueron un apoyo invaluable en mis primeros años de formación universitaria y en el inicio de mi vida adulta. Con amor... – *José Manuel Dubón Monge*

En primer lugar, agradezco a Dios por darme la vida, la sabiduría y la fortaleza necesarias para culminar esta etapa tan importante. Su guía y bendición han sido fundamentales para perseverar y superar cada desafío en el camino.

A mis padres y hermano, quienes han sido el pilar de mi vida y la fuerza que me ha impulsado a seguir adelante. Gracias por su amor incondicional, por sus sacrificios, por creer siempre en mí y por acompañarme en cada paso de este camino.

De manera muy especial, a mi esposa, por su amor, paciencia y comprensión durante todo este proceso. Gracias por caminar a mi lado, por brindarme ánimo en los momentos difíciles y por ser una fuente constante de inspiración para alcanzar este sueño. – *Marcos Antonio Zúniga Tapia*

Agradezco primeramente a Dios por darme la fortaleza, sabiduría y perseverancia necesarias para alcanzar esta meta.

A mi madre, quien con amor, valentía y sacrificio me ha guiado siempre. Gracias, mamá, porque a pesar de las dificultades nunca permitiste que me faltara la esperanza; este logro también es fruto de tu entrega y de tu fe.

De manera muy especial, a mi tía Rosa Erlinda, por su generosidad, apoyo y por estar presente en momentos importantes de mi formación universitaria. Gracias por creer en mí y por brindarme la ayuda necesaria para continuar este camino. – *Jose Angel Guerrero*

RESUMEN DEL TEMA

El presente estudio monográfico se centra en la creación y desarrollo de una plataforma web destinada a los formularios para la Evaluación de Daños y Análisis de Necesidades (EDAN) en el contexto universitario de Nicaragua, promovido por el Consejo Nacional de Rectores (CNR). Este sistema se ideó como respuesta a la necesidad de actualizar los métodos de recopilación, gestión y análisis de datos post-desastres naturales, los cuales tradicionalmente se han llevado a cabo utilizando formularios impresos o plantillas electrónicas no estandarizadas, lo que ha provocado demoras, duplicaciones de información y complicaciones en la integración de datos.

La solución propuesta permite la optimización del manejo de los formularios EDAN 72 y EDAN 120, que son utilizados para obtener información esencial sobre daños estructurales, empresas afectadas, población impactada y recursos disponibles en las instituciones. Mediante una plataforma web adaptable, se facilita la recolección de datos en tiempo real desde dispositivos informáticos o móviles, incluso en áreas con conectividad restringida, asegurando así la continuidad en las operaciones y un acceso rápido a la información. Esta herramienta es un progreso significativo en la capacidad de respuesta del CNR y de las universidades ante situaciones de emergencia, al mejorar la rapidez de la evaluación y la calidad de los informes técnicos.

El diseño del sistema utiliza la metodología UWE (UML-based Web Engineering) como marco de trabajo, la cual promueve el modelado exhaustivo de plataformas web a través de la creación de modelos de requisitos, contenido, navegación, presentación y despliegue. Utilizando esta metodología, se generaron diagramas UML que detallan los procesos funcionales del sistema, la estructura de su base de datos, los flujos de usuario y los componentes tecnológicos. Además, se implementó una API documentada mediante Swagger, lo que asegura transparencia, interoperabilidad y facilidad en el mantenimiento del software.

Desde un punto de vista técnico, la arquitectura del sistema integra Laravel en el backend, React con TypeScript en el frontend y MySQL como gestor de datos, todo ello desplegado en un entorno Nginx. Estas tecnologías fueron elegidas por su solidez, seguridad y apoyo comunitario, además de ser compatibles con la infraestructura

tecnológica actual del CNR. Se validó su viabilidad técnica, operativa y económica a través de modelos de estimación como Puntos de Función y COCOMO, los cuales facilitaron la proyección del esfuerzo, tiempo y personal requeridos para el desarrollo, así como del análisis costo-beneficio a corto y mediano plazo.

Entre los principales beneficios del sistema se encuentran:

- La automatización completa del proceso EDAN, acortando el tiempo de consolidación de datos de varios días a solo minutos.
- La centralización y trazabilidad de la información, lo que asegura mayor exactitud y seguridad.
- La creación de reportes y tableros de control dinámicos que simplifican la toma de decisiones.
- La compatibilidad multiplataforma y la capacidad de operar en entornos con conectividad limitada.
- La escalabilidad del sistema, lo que permite su expansión futura hacia otros sectores o instituciones relacionadas con la gestión de riesgos.

Entre las limitaciones o desventajas observadas, se destaca la necesidad de entrenamiento técnico del personal para un uso efectivo de la herramienta, así como la dependencia de una infraestructura tecnológica estable (servidores, conectividad y mantenimiento).

En síntesis, este trabajo evidencia la factibilidad y el efecto benéfico de la digitalización en la administración de riesgos en la educación superior. La plataforma web creada no solo se presenta como una herramienta tecnológica avanzada, sino que también representa un importante avance hacia la eficacia institucional, la claridad en los procedimientos de evaluación y la capacidad de recuperación del sistema universitario frente a catástrofes naturales. Su aplicación establece un hito en la modernización de las respuestas del CNR y establece los fundamentos necesarios para la transformación hacia un ecosistema informático cohesivo, sostenible y enfocado en la mejora constante.

INDICE

I.	INTRODUCCIÓN.....	1
II.	ANTECEDENTES.....	2
III.	PROBLEMÁTICA.....	3
IV.	OBJETIVOS	4
4.1.	Objetivo general.....	4
4.2.	Objetivos específicos	4
V.	JUSTIFICACIÓN.....	5
VI.	MARCO TEÓRICO.....	7
6.1.	Evaluación de Daños y Análisis de Necesidades (EDAN)	7
6.1.1.	EDAN 72	7
6.1.2.	EDAN 120	8
6.2.	Análisis de requerimientos.....	8
6.2.1.	Modelo del negocio.....	8
6.2.2.	Actores del negocio	9
6.2.3.	Descripción de los procesos del negocio. BPMN	9
6.2.4.	Requerimientos del Sistema: Definición de los requerimientos del sistema a partir del análisis del negocio.....	10
6.3.	Elementos de viabilidad.....	12
6.4.	Diseño basado en UWE	12
6.4.1.	Modelo de requerimientos	13
6.4.2.	Modelo de contenido	14
6.4.3.	Modelo navegacional.....	16
6.4.4.	Modelo de presentación	17
6.4.5.	Modelo de procesos	18
6.4.6.	Modelo de despliegue y componentes	18
6.4.7.	Consideraciones de diseño	19
6.5.	Tecnologías	20
6.5.1.	PHP	20
6.5.2.	Doctrine.....	20
6.5.3.	CSS.....	21
6.5.4.	Tailwind	21
6.5.5.	Javascript.....	22
6.5.6.	ReactJS.....	22
6.5.7.	MySQL Server	23
Capítulo 1.	ANÁLISIS DEL CONTEXTO ORGANIZACIONAL Y REQUERIMIENTOS DEL SISTEMA	24
1.1.	Entorno organizacional actual del departamento de evaluación de riesgos del CNR.	24
1.2.	Descripción de la situación actual	24
1.2.1.	Misión de la institución	25
1.2.2.	Visión de la institución	25

1.3.	Organigrama del departamento de evaluación de riesgos del CNR	25
1.4.	Objetivos del departamento de evaluación de riesgos del CNR.....	27
1.5.	Modelado del negocio	28
1.6.	Actores del sistema.....	30
1.7.	Descripción de los procesos del negocio.....	30
1.8.	Requerimientos del sistema.....	31
1.8.1.	Requerimientos funcionales	35
1.8.2.	Requerimientos no funcionales	41
Capítulo 2.	ANÁLISIS DEL SISTEMA	45
2.1.	Descripción de los procesos principales	45
2.1.1.	Diagrama de actividades.....	45
2.2.	Factibilidad técnica.....	49
2.2.1.	Documentación técnica.....	49
2.2.2.	Infraestructura tecnológica actual, requerimientos de hardware y evaluación de compatibilidad	50
2.2.3.	Software	52
2.2.4.	Requisitos de red y conectividad.....	53
2.2.5.	Seguridad técnica	53
2.2.6.	Compatibilidad y requisitos de integración	53
2.2.7.	Plan de mantenimiento y soporte técnico	54
2.3.	Factibilidad operacional	55
2.3.1.	Necesidades de personal.....	55
2.3.2.	Adiestramiento del personal	56
2.3.3.	Detalle del costo de utilización de medios técnicos en el adiestramiento del personal..	56
2.3.4.	Plan de capacitación.....	57
2.3.5.	Eficiencia en el uso de los recursos.....	58
2.3.6.	Impacto en la productividad.....	58
2.3.7.	Ahorro en costos operacionales.....	59
2.4.	Factibilidad económica	60
2.4.1.	Puntos de función	61
2.4.2.	COCOMO.....	63
2.4.3.	Análisis costo / beneficio	66
Capítulo 3.	DISEÑO BASADO EN UWE	72
3.1.	Modelado de requerimientos.....	72
3.1.1.	Casos de uso	73
3.1.2.	Diagrama de actividades UWE	75
3.2.	Modelo de contenido	76
3.2.1.	Diagrama de clases.....	77
3.2.2.	Diagrama de estados	78
3.2.3.	Modelo de datos	79
3.3.	Modelo de navegación	80
3.4.	Modelo de presentación	83

3.5.	Modelo de procesos	84
3.6.	Modelo de despliegue y componentes	87
3.6.1.	Diagrama de componentes	87
3.6.2.	Diagrama de despliegue.....	88
Capítulo 4.	Arquitectura y desarrollo del sistema	89
4.1.	Visión general de la arquitectura del sistema	89
4.2.	Arquitectura del backend	89
4.2.1.	Base de datos y persistencia de la información	90
4.2.2.	Gestión de la lógica de negocio	91
4.2.3.	Validación y manejo de datos.....	91
4.2.4.	Gestión de seguridad y control de acceso	92
4.2.5.	Manejo de errores y respuestas del sistema	92
4.2.6.	Organización y estructura del proyecto	92
4.3.	Documentación del backend	93
4.4.	Arquitectura del frontend	93
4.4.1.	Enfoque arquitectónico general	94
4.4.2.	Capa central y recursos comunes (core)	94
4.4.3.	Organización por funcionalidades (features).....	95
4.4.4.	Separación interna de capas dentro de cada feature	95
4.4.5.	Inyección de dependencias y control de acoplamiento	96
4.4.6.	Comunicación con el backend e integración de servicios	96
4.4.7.	Gestión del estado y flujo de información	96
4.4.8.	Interfaz de usuario y experiencia de uso	97
4.4.9.	Herramientas de construcción y calidad del código	97
4.5.	Herramientas y entorno de desarrollo	97
VII.	CONCLUSIONES.....	98
VIII.	RECOMENDACIONES.....	99
IX.	Bibliografía	100
X.	ANEXOS.....	101
	Anexo A. Gastos de capacitación	101
	Anexo B. Pesos de complejidad de los componentes de los puntos de función	105
	Anexo C. Multiplicador de lenguaje de programación.....	105
	Anexo D. Valores de referencia Factores de escala y Factores del producto.....	106
	Anexo E. Factores de escala	108
	Anexo F. Factores del producto	110
	Anexo G. Estimación del ahorro y gasto de software de escala	112
	Anexo H. Cálculo de πEM_i	112

Anexo I. Costos de consumo de energía	113
Anexo J. Plantillas de Coleman	114
Anexo K. Modelo de datos	123
Anexo L. Otros diagramas	124
Anexo M. Imágenes del sistema	140
Anexo N. Estructura de carpetas backend y frontend	146
Anexo O. Formatos EDAN.....	148

INDICE DE FIGURAS

Figura 1 Etapas de los requerimientos funcionales y no funcionales	11
Figura 2 Organigrama de la Comisión de Universidades del CNR para la gestión integral de riesgos y desastres	26
Figura 3 Diagrama del modelado del negocio (Situación actual)	29
Figura 4 Diagrama de actividad del proceso principal de asignar roles	45
Figura 5 Diagrama de actividad del proceso principal de reasignar roles	46
Figura 6 Diagrama de actividad del proceso principal de recolectar los datos	46
Figura 7 Diagrama de actividad del proceso principal del análisis de la información	48
Figura 8 Caso de uso: Gestión de usuarios, roles y permisos	75
Figura 9 Diagrama de clases	77
Figura 10 Diagrama de estados	78
Figura 11 Modelo de navegación por roles (Superadministrador y Administrador).....	80
Figura 12 Modelo de navegación por roles (Evaluador y Consultor)	82
Figura 13 Modelo de presentación Login.....	83
Figura 14 Modelo de presentación Vista de inicio	83
Figura 15 Modelo de presentación Módulo de catalogos	83
Figura 16 Diagrama de proceso - Buscar incidencia	84
Figura 17 Diagrama de proceso - Crear incidencia	85
Figura 18 Diagrama de proceso - Editar incidencia.....	85
Figura 19 Diagrama de proceso - Eliminar incidencia	86
Figura 20 Diagrama de componentes del sistema.....	87
Figura 21 Diagrama de despliegue del sistema	88
Figura 22 Oferta ofrecida en el mercado laboral	101
Figura 23 Oferta ofrecida en el mercado laboral	101
Figura 24 Oferta ofrecida en el mercado laboral	102
Figura 25 Oferta de computadora portátil	103
Figura 26 Oferta de computadora portátil	103
Figura 27 Oferta de computadora portátil	104
Figura 28 Pesos de complejidad de los componentes de los puntos de función (Panda, Sharma, & Roy, 2018)	105
Figura 29 Valores de referencia Factores de escala (Panda, Sharma, & Roy, 2018)	106
Figura 30 Valores de referencia Factores de producto (Panda, Sharma, & Roy, 2018).....	107
Figura 31 Modelo de datos.....	123
Figura 32 Caso de uso: Gestión de incidencias.....	124
Figura 33 Caso de uso: Gestión de catálogo.....	124
Figura 34 Diagramas de actividades de incidencias	125
Figura 35 Diagrama actividad de buscar universidades.....	126
Figura 36 Diagrama de actividad de actualizar universidad	126
Figura 37 Diagrama de actividad de dar de baja universidad	127
Figura 38 Diagrama de actividad de crear usuario.....	127
Figura 39 Diagrama de actividad de editar usuario	128
Figura 40 Diagrama de actividad de dar de baja un usuario	129
Figura 41 Diagrama actividad de creación de rol.....	130
Figura 42 Diagrama de actividad de dar de baja un rol	130
Figura 43 Diagrama de actividad de editar un rol.....	131
Figura 44 Diagrama de actividad de creación de permiso	132
Figura 45 Diagrama de actividad de dar de baja un permiso	132
Figura 46 Diagrama de actividad edición de permiso.....	133
Figura 47 Diagrama de proceso - Actualizar universidad	134
Figura 48 Diagrama de proceso - Dar de baja universidad	134
Figura 49 Diagrama de proceso - Descargar informe	135

Figura 50 Diagrama de proceso - Crear permiso	135
Figura 51 Diagrama de proceso - Dar de baja a permiso	136
Figura 52 Diagrama de proceso - Editar permiso.....	136
Figura 53 Diagrama de proceso - Crear rol	137
Figura 54 Diagrama de proceso - Dar de baja a rol	137
Figura 55 Diagrama de proceso - Editar rol.....	138
Figura 56 Diagrama de proceso - Crear usuario	138
Figura 57 Diagrama de proceso - Dar de baja a usuario	139
Figura 58 Diagrama de proceso - Editar usuario.....	139
Figura 59 Inicio de sesión del sistema	140
Figura 60 Vista de inicio	140
Figura 61 Vista principal de gestión de incidencias	141
Figura 62 Creación de incidencia.....	141
Figura 63 Formulario Edan 120	142
Figura 64 Formulario Edan 72.....	142
Figura 65 Modulo de gestión de usuarios	143
Figura 66 Módulo de gestión de catalogos	143
Figura 67 Detalle de un catalogo.....	144
Figura 68 Reporte departamental	144
Figura 69 Reporte institucional.....	145
Figura 70 Reporte regional	145
Figura 71 Estructura de carpetas del backend	146
Figura 72 Estructura de carpetas del frontend	147
Figura 73 Formato EDAN 72	148
Figura 74 Formato EDAN 120	149

INDICE DE TABLAS

Tabla 1 Actores del sistema	30
Tabla 2 OBJ-001 - Automatizar formularios EDAN 71 y EDAN 120	32
Tabla 3 OBJ-002 - Gestionar el acceso al sistema según jerarquías de roles y permisos definidos	32
Tabla 4 OBJ-003 - Permitir a los administradores crear incidencias y asignarlas a los grupos encargados de recopilar la información	33
Tabla 5 OBJ-004 - Facilitar la gestión de catálogos internos del sistema	33
Tabla 6 OBJ-005 - Ofrecer funciones de consulta y reportes para apoyar en la toma de decisiones	34
Tabla 7 OBJ-006 - Centralizar la configuración de usuarios, roles y permisos para facilitar su gestión y mantenimiento	34
Tabla 8 FRQ-001 - El sistema debe permitir a los administradores crear incidencias en base a los formularios EDAN 72 o EDAN 120	35
Tabla 9 FRQ-002 - El sistema debe permitir a los administradores asignar un grupo de usuarios a cada incidencia	36
Tabla 10 FRQ-003 - El sistema debe permitir a los usuarios autorizados llenar los formularios EDAN asignados	36
Tabla 11 FRQ-004 - El sistema debe permitir a los administradores eliminar incidencias existentes	37
Tabla 12 FRQ-005 - El sistema debe permitir a los administradores crear, editar y eliminar usuarios	37
Tabla 13 FRQ-006 - El sistema debe permitir asignar usuarios a uno o más grupos	38
Tabla 14 FRQ-007 - El sistema debe permitir gestionar roles y permisos	38
Tabla 15 FRQ-008 - El sistema debe permitir la administración de catálogos internos	39
Tabla 16 FRQ-009 - El sistema debe validar que los formularios usen únicamente valores activos en los catálogos	39
Tabla 17 FRQ-010 - El sistema debe generar reportes de incidencias filtradas por fecha, ubicación, tipo de evento o universidad	40
Tabla 18 FRQ-011 - El sistema debe permitir exportar reportes en formato PDF o Excel	40
Tabla 19 NFR-001 - Compatibilidad con navegadores modernos	41
Tabla 20 NFR-002 - Autenticación segura	41
Tabla 21 NFR-003 - Control de acceso basado en roles	42
Tabla 22 NFR-004 - Protección contra ataques comunes	42
Tabla 23 NFR-005 - Base de datos segura con respaldo	43
Tabla 24 NFR-006 - Registro de acciones en log de auditoría	43
Tabla 25 NFR-007 - Modularidad y escalabilidad	44
Tabla 26 NFR-008 - Adaptabilidad a dispositivos	44
Tabla 27 Infraestructura actual, requerimientos mínimos y recomendados	51
Tabla 28 Adiestramiento del personal	56
Tabla 29 Impacto en la productividad	58
Tabla 30 Ahorro en materiales y servicios	59
Tabla 31 Puntos de función sin ajustar	61
Tabla 32 Características generales del sistema	62
Tabla 33 Factor de ajuste	63
Tabla 34 Puntos de función ajustados	63
Tabla 35 Estimación total de líneas de código	63
Tabla 36 Estimación del esfuerzo	64
Tabla 37 Calculo del tiempo de desarrollo	65
Tabla 38 Estimación de personal	66
Tabla 39: Estimación de personal	66
Tabla 40 Esfuerzo, tiempo de desarrollo y cantidad de personas por etapa	66
Tabla 41 Distribución del costo de fuerza de trabajo y consumo de energía por etapa	67
Tabla 42 Costo de materiales	68
Tabla 43 Costos directos	69
Tabla 44 Costos indirectos	69

Tabla 45 Costo total de inversión del proyecto	69
Tabla 46 Multiplicador de lenguaje de programación.....	105
Tabla 47 Factores de escala calculados	109
Tabla 48 Indicadores del producto calculados	111
Tabla 49 Estimación del ahorro y gasto de software de escala.....	112
Tabla 50 Cálculo de πEM_i	112
Tabla 51 Consumo de energía por hora de una PC promedio (Ce).....	113
Tabla 52 Costo de energía promedio (CKH)	113
Tabla 53 Tiempo de trabajo al mes (NoH).....	113
Tabla 54 Costo de consumo de energía (Cce).....	113
Tabla 55 Gestionar incidencias (fuente: elaboración propia)	114
Tabla 56 Gestionar catálogos (fuente: elaboración propia)	115
Tabla 57 Gestionar usuarios (fuente: elaboración propia)	117
Tabla 58 Gestionar roles (fuente: elaboración propia)	119
Tabla 59 Gestionar permisos (fuente: elaboración propia)	121

I. INTRODUCCIÓN

En los últimos años, Nicaragua ha sido afectada por diversos eventos naturales que han impactado significativamente a distintos sectores, entre ellos la educación superior. Ante esta situación, se ha vuelto imprescindible contar con mecanismos eficientes que permitan evaluar y documentar de forma oportuna los daños ocasionados en las universidades y sus recintos. El Consejo Nacional de Rectores (CNR), como organismo rector de las instituciones de educación superior del país, ha identificado esta necesidad y ha promovido acciones orientadas a fortalecer la gestión de la información posterior a emergencias.

En este contexto, los formularios EDAN 72 y EDAN 120 (Anexo O) constituyen una herramienta fundamental para la recopilación de información sobre las afectaciones provocadas por eventos adversos. Sin embargo, su gestión tradicional, basada en procesos manuales o en archivos digitales no estandarizados, ha limitado la agilidad, precisión y disponibilidad de los datos necesarios para el análisis y la toma de decisiones, evidenciando la necesidad de una solución tecnológica que optimice dicho proceso.

El presente proyecto de monografía tiene como propósito el diseño y desarrollo de un sistema web responsivo que facilite la recolección, gestión y consulta de la información contenida en los formularios EDAN. La plataforma está orientada al uso interno del CNR y busca apoyar tanto a los agentes de campo encargados de la recopilación de datos en zonas afectadas, como al personal administrativo responsable del análisis y la toma de decisiones. El sistema ha sido concebido para operar en computadoras y dispositivos móviles, priorizando la usabilidad, accesibilidad y escalabilidad, de acuerdo con las condiciones reales de trabajo de los usuarios.

Para el desarrollo del proyecto se realizó un análisis del entorno organizacional y de los requerimientos del sistema, así como el diseño de modelos conceptuales y técnicos basados en el enfoque UWE (UML-based Web Engineering). Asimismo, se evaluó la factibilidad técnica y tecnológica de la solución propuesta, garantizando su compatibilidad con la infraestructura institucional existente.

II. ANTECEDENTES

En Nicaragua, la gestión de información post-desastres en el sector educativo superior aún depende en gran medida de procesos manuales. Aunque el Consejo Nacional de Rectores (CNR) y el Sistema Nacional para la Prevención, Mitigación y Atención de Desastres (SINAPRED) han trabajado en protocolos de evaluación de daños, no existe un sistema digital unificado para el registro y análisis automatizado de los formularios EDAN 72 y EDAN 120 (Anexo O) en universidades. Actualmente, estos datos se recopilan en formatos físicos o archivos digitales dispersos, lo que dificulta su consolidación y retrasa la toma de decisiones (CNR, 2023).

A nivel regional, experiencias como las de Costa Rica y El Salvador demuestran los beneficios de la digitalización en la gestión de riesgos educativos. Costa Rica ha implementado el Sistema Nacional de Gestión del Riesgo (SNGR), plataforma que incorpora módulos específicos para registrar afectaciones en centros educativos, optimizando la respuesta institucional ante emergencias informado por la Comisión Nacional de Emergencias (CNE, 2022). Caso similar es el de El Salvador, donde la Dirección General de Protección Civil en coordinación con el Ministerio de Medio Ambiente ha propuesto el Sistema de Evaluación de Riesgos Climáticos (SERC), herramienta que permite evaluaciones solicitados por las instituciones nacionales, informado por el Ministerio de Medio Ambiente (MARN, 2017). Estas implementaciones contrastan con la realidad nicaragüense, donde persiste la dependencia de métodos tradicionales en el ámbito universitario.

A nivel global, organizaciones como la UNESCO (United Nations Educational, Scientific and Cultural Organization) y la UNDRR (United Nations Office for Disaster Risk Reduction) han impulsado el uso de tecnologías para la gestión de riesgos en educación. Un ejemplo destacado es el "Global Library of School Infrastructure (GLOSI)", que recopila datos estandarizados sobre daños en escuelas y universidades (UNDRR, 2023). Además, Education in Emergencies (EiE) promueve el uso de sistemas digitales para respuestas rápidas (INEE - Inter-agency Network for Education in Emergencies, 2022).

III. PROBLEMÁTICA

En el contexto nicaragüense, la gestión post-desastre en el ámbito universitario enfrenta serias limitaciones derivadas de la falta de herramientas digitales especializadas para la recolección y análisis de datos. Actualmente, los formularios EDAN 72 y EDAN 120 (instrumentos fundamentales para evaluar daños y necesidades en las primeras 72 y 120 horas tras un evento adverso) se gestionan mediante archivos físicos o planillas digitales no estandarizadas, lo que dificulta su consolidación, retrasa la toma de decisiones y expone a errores en la información recolectada.

Esta situación representa un reto significativo para el Consejo Nacional de Rectores (CNR), institución responsable de coordinar y supervisar el accionar de las universidades ante emergencias. Aunque se han dado pasos importantes en el diseño de los formatos EDAN y en la adopción de protocolos de evaluación, la carencia de un sistema automatizado impide optimizar el flujo de trabajo, garantizar la integridad de los datos y responder con agilidad a los desastres naturales.

La inexistencia de una solución digital adaptada a las necesidades del entorno universitario pone en riesgo la eficiencia de los procesos de evaluación, análisis y generación de reportes, al tiempo que limita la capacidad del CNR para actuar de manera oportuna y fundamentada. Se hace, por tanto, imperativo diseñar e implementar un sistema web robusto, accesible y estandarizado, que permita automatizar los formularios EDAN y facilitar su uso tanto en computadoras como en dispositivos móviles, garantizando una adecuada gestión de la información ante emergencias.

IV. OBJETIVOS

4.1. Objetivo general

- Desarrollar un sistema web eficiente para la evaluación de daños y análisis de necesidades en el entorno universitario de Nicaragua basado en los formatos EDAN, utilizando la metodología UWE.

4.2. Objetivos específicos

- Analizar los requerimientos técnicos y funcionales necesarios para la automatización de los formularios EDAN, considerando tanto las necesidades del CNR como las condiciones operativas de los agentes de campo.
- Diseñar los modelos conceptuales, estructurales y navegacionales del sistema web aplicando la metodología UWE, asegurando la coherencia entre los procesos de negocio, la interfaz de usuario y el flujo de información.
- Desarrollar una plataforma web adaptable a diferentes dispositivos, que permita la gestión eficiente de las incidencias, la recolección de datos en campo y la generación de reportes automatizados para la toma de decisiones institucionales.

V. JUSTIFICACIÓN

El presente trabajo monográfico tiene como propósito optimizar y digitalizar el proceso de Evaluación de Daños y Análisis de Necesidades (EDAN), actualmente implementado mediante formatos EDAN 72 y EDAN 120 (Anexo O) en hojas de cálculo Excel. Estos formatos, desarrollados para recopilar datos en las primeras 72 y 120 horas posteriores a un evento adverso, constituyen la herramienta esencial para registrar información crítica relacionada con la infraestructura, los servicios y la población afectada. No obstante, la dependencia de una gestión manual y de la transferencia de archivos entre dispositivos genera retrasos prolongados en la consolidación de datos, riesgo de pérdida o duplicación de registros y dificultades para mantener la coherencia de la información recolectada en distintas localidades.

Para superar estas limitaciones, se plantea el diseño de una plataforma web de interfaz adaptativa, cuyos requerimientos funcionales y no funcionales se analizan bajo la metodología UWE (UML for Web Engineering), reconocida por su enfoque integral en proyectos de sistemas de información web. UWE facilita la especificación rigurosa de requisitos a través de su modelo de requerimientos y proporciona herramientas para diseñar el modelo navegacional, el modelo de contenido y el modelo de presentación, asegurando coherencia entre la estructura de datos, la interacción de usuario y los procesos de negocio. Dicha propuesta contempla la digitalización íntegra de los formularios EDAN, la captura de datos, junto con mecanismos de validación automática de campos, comprobación de rangos y formatos para garantizar la calidad de los registros. Asimismo, el estudio incluye la definición de un modelo de gestión de roles y permisos que permite registrar detalladamente cada acción de los usuarios, asegurando la trazabilidad y la integridad de la base de datos.

La centralización de la información y la automatización de los procesos de consolidación reducen los tiempos de obtención de reportes de varios días a minutos. Adicionalmente, la generación de reportes parametrizables y de tableros de indicadores facilita la visualización de métricas clave, tales como el número de incidencias por zona, el estado de avance de las evaluaciones y los tiempos de respuesta promedio. Estas funcionalidades resultan imprescindibles para que las

autoridades académicas y de gestión de desastres, como el CNR y el SINAPRED, puedan tomar decisiones fundamentadas de manera oportuna.

Se prevé que, tras la puesta en marcha de la plataforma web, y la formación práctica de evaluadores y personal de apoyo, se optimice significativamente el uso de sus funcionalidades. Esta fase formativa potenciaría las competencias técnicas y metodologías necesarias, promoviendo una cultura institucional orientada a la resiliencia y la gestión integral del riesgo, y garantizaría que el equipo humano disponga de los conocimientos adecuados para emplear la herramienta de manera efectiva ante futuras contingencias.

La arquitectura tecnológica propuesta se fundamenta en la selección de herramientas consolidadas. En el backend, se propuso el uso de Laravel por su arquitectura modular, su sistema de migraciones y su middleware integrado de seguridad; en el frontend, React se estableció como una opción idónea gracias a su modelo de componentes reutilizables y a la eficacia de su DOM virtual. Para el almacenamiento de los datos, se opta por MySQL, debido a su fiabilidad, rendimiento y amplio soporte en entornos de producción. La combinación de estas tecnologías garantiza escalabilidad, mantenibilidad y un rendimiento óptimo, alineándose con los estándares definidos en el análisis UWE.

VI. MARCO TEÓRICO

6.1. Evaluación de Daños y Análisis de Necesidades (EDAN)

La información presentada en este apartado se fundamenta en los aportes obtenidos mediante entrevista realizada al Dr. Armando Ugarte Solís, miembro de la Comisión de Universidades del Consejo Nacional de Rectores (CNR) para la gestión del riesgo de desastres, quien describió el uso, clasificación y aplicación de los formularios EDAN en el contexto nacional (Ugarte Solís, comunicación personal, 22 de mayo de 2025).

La Evaluación de Daños y Análisis de Necesidades (EDAN) es un instrumento técnico utilizado en la gestión de riesgos y atención de emergencias, cuyo propósito es recopilar información sistemática y confiable sobre las afectaciones ocasionadas por un evento adverso, así como identificar las necesidades prioritarias para la respuesta y recuperación.

El EDAN se aplica posterior a la ocurrencia de eventos naturales, socio-naturales o provocados por el ser humano, tales como huracanes, inundaciones, sismos, incendios, accidentes industriales u otros fenómenos que impactan la infraestructura, los servicios, la población y el medio ambiente.

6.1.1. EDAN 72

El formulario EDAN 72 se utiliza durante las primeras 72 horas posteriores a la ocurrencia de un evento adverso. Su objetivo principal es obtener una evaluación rápida y preliminar de los daños, permitiendo a las autoridades identificar de manera temprana las afectaciones más críticas y priorizar acciones inmediatas de respuesta.

Este formato se caracteriza por la recopilación de información básica sobre daños estructurales, afectaciones a servicios esenciales, impacto en la población y condiciones generales del área evaluada.

6.1.2. EDAN 120

El formulario EDAN 120 se aplica una vez transcurridas 120 horas posteriores al evento. A diferencia del EDAN 72, este instrumento permite una evaluación más detallada y consolidada de los daños y necesidades, incorporando información ampliada sobre infraestructura, sectores productivos, medio ambiente y requerimientos para la etapa de rehabilitación y recuperación.

6.2. Análisis de requerimientos

Es una fase fundamental en el ciclo de vida del desarrollo de software. Esta etapa consiste en identificar, definir y documentar las necesidades y expectativas que el sistema debe satisfacer. Los requerimientos son la base sobre la cual se construirá el sistema y guiarán su diseño, implementación y pruebas.

A veces, existe una brecha entre lo que las partes interesadas dicen y lo que realmente quieren. El objetivo del análisis de requisitos es garantizar que todos los requisitos empresariales, de software y de producto representen con exactitud las necesidades de las partes interesadas, y el análisis de requisitos consiste en comprender claramente las necesidades de las partes interesadas para que el producto final se ajuste a sus expectativas. (Saradhi & Jena, 2022)

6.2.1. Modelo del negocio

El "Modelado del Negocio" implica la representación gráfica y descriptiva de la estructura, procesos y relaciones dentro de una organización.

“Un modelo es la representación gráfica de un objeto empleado dentro del proceso de negocio o actividades comúnmente realizadas; mientras que negocio es la actividad que permite generar ingresos a una organización.” (Molina & Pedreira, 2019, p. 49)

Con base en lo explicado, se concluye que un modelo de negocios consiste en la definición de las actividades comunes que realiza la organización, detallando las siguientes preguntas:

¿Cuál es la organización de la empresa?

¿De qué manera interactúan las personas o actores dentro de la misma?

¿Qué objetivos busca cumplir?

¿Cuáles son las actividades que realiza la empresa?

¿De qué manera son efectuadas las actividades?

6.2.2. Actores del negocio

Los "Actores del Negocio" son las personas, organizaciones o sistemas externos que interactúan con el sistema que estás desarrollando.

6.2.3. Descripción de los procesos del negocio. BPMN

BPMN (Business Process Model and Notation) ofrece una notación gráfica estandarizada para el modelado de procesos de negocio. Según la OMG (Object Management Group), permite diseñar gráficamente procesos de negocio en su totalidad. La especificación describe dos niveles de abstracción en el proceso de modelado: un nivel detallado que contiene únicamente un conjunto de elementos gráficos utilizados por el analista de negocio, y un nivel que establece la correspondencia entre estos elementos y un lenguaje de ejecución. (Reyes Hernández, Hernández Cárdenas, Moreno Montes de Oca, Rodríguez Morffi, & Moreno Rodríguez, 2019)

El que BPMN sea un estándar de múltiples proveedores lo convierte en un marco de trabajo robusto y confiable, mantenido por una organización líder en gestión de estándares. La adopción generalizada por parte de modeladores y proveedores de herramientas subraya la aceptación y utilidad que ha alcanzado esta notación en la práctica empresarial. Con esta sólida base, BPMN se ha convertido en un vehículo indispensable para la representación, análisis y mejora de los procesos de negocio.

En palabras simples, BPMN no solo es un conjunto de símbolos y reglas de notación, sino una herramienta esencial para comprender, comunicar y optimizar los procesos empresariales.

6.2.4. Requerimientos del Sistema: Definición de los requerimientos del sistema a partir del análisis del negocio

Las características y funcionalidades del sistema objetivo se describen en los requerimientos del sistema. Las expectativas de los usuarios sobre el producto de software se comunican a través de los requerimientos. Desde la perspectiva del cliente, los requerimientos pueden ser claramente esperados u ocultos, reconocidos o desconocidos, y previstos o imprevistos. (Saradhi & Jena, 2022)

A menudo, los requerimientos del sistema de software se clasifican como requerimientos funcionales o requerimientos no funcionales:

6.2.4.1. Requerimientos funcionales

Son enunciados acerca de servicios que el sistema debe proveer, de cómo debería reaccionar el sistema a entradas particulares y de cómo debería comportarse el sistema en situaciones específicas. En algunos casos, los requerimientos funcionales también explican lo que no debe hacer el sistema.

Los requerimientos funcionales especifican la funcionalidad del software que los desarrolladores deben incorporar al producto para que los usuarios puedan realizar sus tareas, satisfaciendo así los requerimientos de negocio. Los requerimientos funcionales se documentan en una especificación de requerimientos de software (SRS). (Saradhi & Jena, 2022)

6.2.4.2. Requerimientos no funcionales

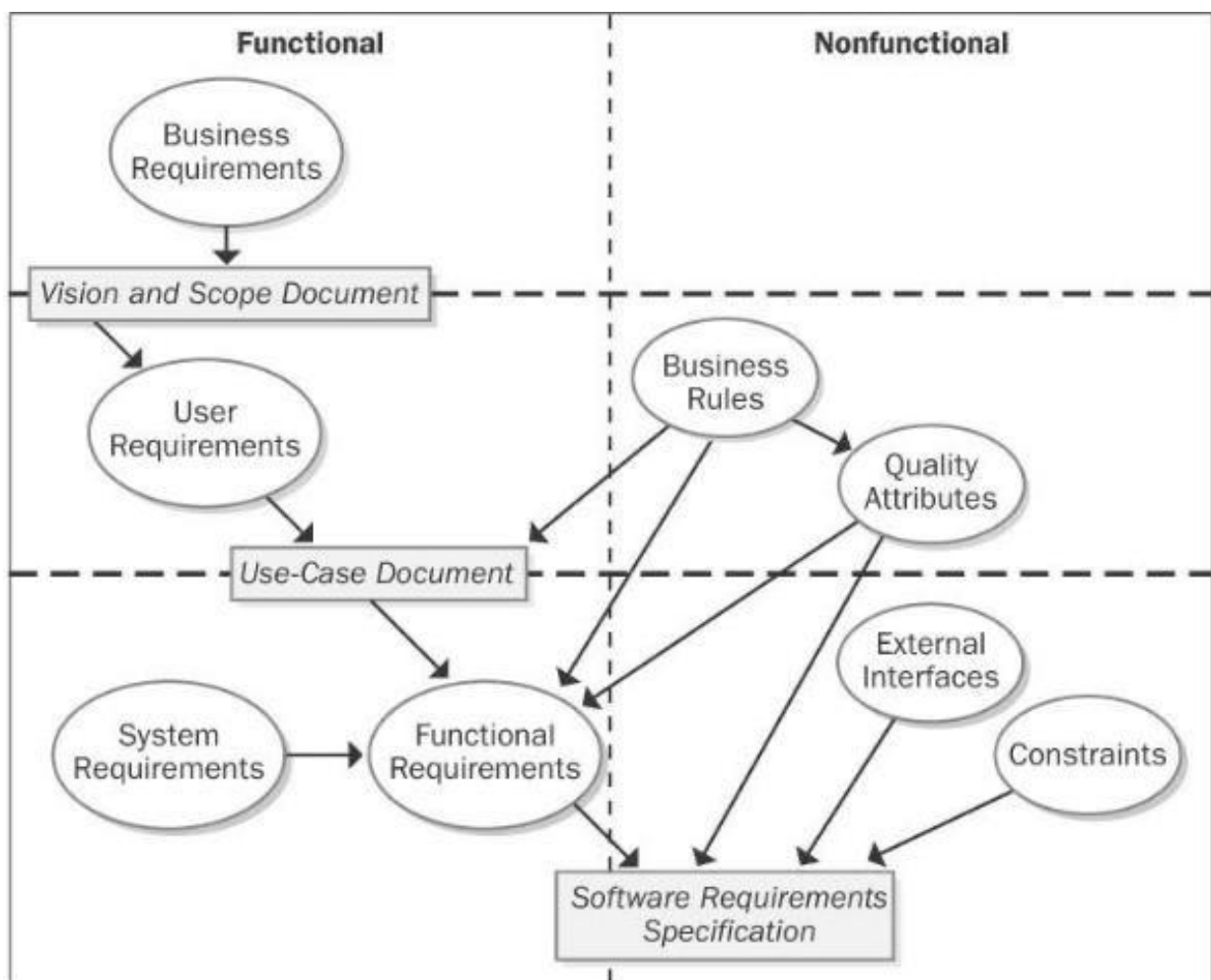
Básicamente, son los requisitos de calidad relacionados en el contrato del proyecto que debe satisfacer el sistema. En lugar de centrarse en la funcionalidad del sistema, los requisitos no funcionales definen cómo debe funcionar el sistema. La importancia de estos factores o su aplicación al proyecto determinarán cómo se priorizan y en qué

medida. También se conocen como requisitos no conductuales. (Saradhi & Jena, 2022)

La siguiente etapa del proceso de ingeniería de requerimientos es la obtención y análisis de requerimientos. En esta actividad, los ingenieros de software trabajan con los clientes y los usuarios finales del sistema para determinar el dominio de la aplicación, qué servicios debe proporcionar el sistema, el rendimiento requerido del sistema, las restricciones hardware, etcétera.

Figura 1

Etapas de los requerimientos funcionales y no funcionales



Nota. Se obtienen las etapas de requerimientos funcionales y no funcionales desde el libro *Software Requirements and Estimation: A Systematic Approach* (Saradhi & Jena, 2022, p. 10)

6.3. Elementos de viabilidad

Cuando un cliente se pone en contacto con una empresa para que le construyan un producto deseado, tiene una noción general de todas las tareas que debe realizar el software y todas las funcionalidades que quiere ver en él.

Los analistas realizan una información exhaustiva a partir de estos datos para determinar si es posible desarrollar el sistema requerido y su funcionalidad.

El objetivo de la organización es el tema principal de este estudio de viabilidad. Este análisis examina si el producto de software puede materializarse realmente en términos de ejecución, contribución del proyecto a la organización, restricciones financieras y de acuerdo con los valores y objetivos de la empresa. Investiga los aspectos técnicos del proyecto y del producto, como su usabilidad, mantenimiento, productividad y potencial de integración. (Saradhi & Jena, 2022)

El resultado de esta fase debe ser un informe de estudio de viabilidad que incluya comentarios y sugerencias suficientes para la dirección sobre la conveniencia o no de realizar el proyecto.

6.4. Diseño basado en UWE

UWE (Unified Web Engineering) representa una perspectiva integral en el desarrollo de sistemas web, destacando su enfoque meticuloso en cada fase del ciclo de vida del desarrollo.

Uno de los pilares fundamentales de UWE es el énfasis en el modelado preciso, donde cada aspecto, desde la arquitectura hasta la interfaz de usuario, se aborda mediante modelos detallados.

Esta aproximación no solo facilita la comprensión y toma de decisiones, sino que también establece una base sólida para el desarrollo y la gestión del proyecto.

Además, UWE promueve la modularidad y la reutilización de componentes, facilitando tanto la creación inicial como el mantenimiento a largo plazo de los sistemas web. La adaptabilidad es otro rasgo clave de UWE, ya que se diseñó para integrar tecnologías

emergentes, garantizando que los sistemas desarrollados sigan siendo relevantes y eficientes en un entorno tecnológico en constante evolución.

La filosofía de UWE va más allá de la mera implementación técnica, ya que aborda la importancia de la documentación exhaustiva para cada aspecto del desarrollo.

Este enfoque no solo actúa como una guía clara para los desarrolladores, sino que también contribuye a la calidad del producto final y simplifica el proceso de mantenimiento. La colaboración eficiente entre los miembros del equipo es un componente esencial de UWE, ya que adopta estándares compartidos y prácticas colaborativas.

En conjunto, UWE no solo aborda las necesidades inmediatas del desarrollo web, sino que también prepara a los profesionales para enfrentar los desafíos futuros, consolidándose como un enfoque integral que busca elevar el estándar general de la ingeniería web.

Las principales características de UWE son las diferentes vistas que se representan como modelos UML y el uso de un conjunto de estereotipos y etiquetas, que definen los elementos del modelo de UWE en su perfil UML. (Busch, 2016)

6.4.1. Modelo de requerimientos

Esta extensión propone el uso de un modelo de requisitos para el desarrollo de aplicaciones web. Este modelo se basa en el concepto de tarea que se ha reorientado para capturar no sólo los requisitos estructurales y de comportamiento (como ocurre en las aplicaciones no web), sino también requisitos de navegación.

6.4.1.1. Casos de uso

Los casos de uso y los objetos del proceso de evaluación son un punto de partida para identificar conceptos relevantes relacionados con la seguridad para utilizar y evaluar métodos, notaciones y herramientas durante el proceso de ingeniería de software. (Busch, 2016)

Los casos de uso se utilizan para visualizar las funcionalidades que proporcionará el sistema. El diagrama de casos de uso representa los casos de uso, los actores y las asociaciones, mostrando las funciones de los actores en la interacción con el sistema, por ejemplo, activando algunos casos de uso.

6.4.1.2. Diagrama de actividades UWE

En el marco de la metodología Unified Web Engineering (UWE), el Diagrama de Actividad emerge como una herramienta esencial para la representación gráfica del dinamismo en el desarrollo de aplicaciones web.

Como componente de la Unified Modeling Language (UML), este diagrama captura el flujo de actividades dentro del sistema, visualizando de manera clara las interacciones entre actores, que pueden ser usuarios o sistemas externos, así como las diversas acciones que se ejecutan en el sistema.

Cada actividad se representa como un nodo, y las transiciones entre ellos delinean la secuencia de eventos en el sistema web, proporcionando una valiosa comprensión del comportamiento dinámico.

La utilización del diagrama de actividad según la metodología UWE ofrece una visión detallada y estructurada de las interacciones dentro de un sistema web. Esta representación visual no solo facilita la comprensión del flujo de trabajo, sino que también sirve como una guía efectiva para diseñadores y desarrolladores, permitiendo una identificación más clara de áreas de mejora y optimización en el proceso de desarrollo de aplicaciones web eficientes y sólidas.

Los diagramas de actividad se construyen a partir de acciones del sistema y puntos de interacción, los cuales reflejan momentos clave en una tarea donde el sistema y el usuario intercambian información. Este intercambio es crucial, ya que permite capturar la semántica de navegación en el nivel de requisitos. (Busch, 2016)

6.4.2. Modelo de contenido

El modelo de contenido contiene la estructura de datos utilizada por la aplicación. «Estructura de datos» suele referirse a la estructura de clases importantes en una

aplicación orientada a objetos o a la estructura lógica de los datos en una base de datos y, en muchas ocasiones, ambas están estrechamente relacionadas. En el caso de lenguajes no orientados a objetos, las clases de Contenido pueden representar módulos principales o archivos de código y sus relaciones entre sí, en caso de que esto resulte útil para planificar la estructura de la aplicación o con fines de documentación. (Busch, 2016)

6.4.2.1. Diagrama de Clases (Clase – Atributos – Métodos – Relaciones)

El diagrama de clases se erige como una representación gráfica de las clases que conforman el sistema analizado, detallando sus atributos, métodos y las relaciones que las vinculan. Cada clase se presenta de manera estructurada, revelando las propiedades que la caracterizan, los comportamientos que ejecuta y cómo se entrelaza con otras clases. Este componente visual facilita la comprensión integral de la arquitectura del sistema, permitiendo identificar y analizar de manera efectiva las interdependencias entre los diversos elementos que lo componen.

6.4.2.2. Diagrama de estado

El diagrama de estado se erige como herramienta fundamental para ilustrar las fases o estados que puede experimentar un objeto en su ciclo de vida. Este modelo gráfico proporciona una representación clara de las transiciones entre estados, los eventos que las desencadenan y las acciones asociadas a cada cambio de estado. Este enfoque provee una visión dinámica del sistema, permitiendo comprender cómo las entidades evolucionan y responden a estímulos externos.

6.4.2.3. Modelo de datos (SQL o NoSQL)

El modelo de datos constituye la estructura subyacente que organiza y almacena la información relevante para el sistema. En este contexto, se considerarán las opciones de modelos de bases de datos, ya sea SQL (Structured Query Language) o NoSQL (Not Only SQL), en función de las necesidades específicas del proyecto. Este componente aborda la definición de tablas, relaciones, restricciones y otros elementos esenciales para la gestión eficiente de los datos, asegurando la integridad y coherencia de la información almacenada.

En síntesis, el Modelo de Contenido propuesto, compuesto por el Diagrama de Clases, el Diagrama de Estado y el Modelo de Datos, constituye el marco teórico que orienta la comprensión y análisis de la estructura, dinámica y gestión de la información dentro del sistema bajo estudio. Cada elemento se concibe como una pieza integral del rompecabezas, contribuyendo de manera conjunta a la construcción de un enfoque sólido y formal para la conceptualización del objeto de investigación.

6.4.3. Modelo navegacional

El modelo navegacional en la metodología UWE (Web Engineering) desempeña un papel crucial al especificar la forma en que los usuarios interactúan con el sistema web. Este modelo, inspirado en las pautas delineadas en el libro de referencia "Web Engineering: Modelling and Implementing Web Applications" (páginas 71-74), se desarrolla en dos fases distintas: "Authoring-in-the-Large" y "Authoring-in-the-Small".

6.4.3.1. Diagrama de navegación por roles

Dentro del modelo navegacional, el diagrama de navegación por roles destaca la adaptabilidad del sistema a diferentes tipos de usuarios. Cada rol específico tiene asignado un conjunto único de permisos y restricciones de navegación. Este diagrama se construye considerando los contextos y las Unidades de Información Abstracta (AIU) definidas en las fases anteriores, adaptando la navegación a las necesidades y privilegios de cada usuario o grupo de usuarios.

En síntesis, el modelo navegacional en la metodología UWE se estructura en dos fases, "Authoring-in-the-Large" y "Authoring-in-the-Small", las cuales culminan en un diagrama de navegación por roles que personaliza la experiencia de usuario según sus roles y responsabilidades en el sistema web.

- **Authoring-in-the-Large**

En esta fase, se establece una vista global de la navegación mediante un mapa navegacional. Este mapa, representado gráficamente como un grafo dirigido, incorpora contextos de navegación y subsistemas, delineando relaciones y rutas

válidas a través del sistema. Los contextos de navegación se categorizan en "Exploration" (alcance amplio y accesible desde cualquier contexto) y "Sequence" (acceso secuencial siguiendo una ruta predeterminada). Los enlaces de navegación pueden ser de tipo "Sequence" (enlaces contextuales) o "Exploration" (enlaces no contextuales), definiendo así las vías de navegación dentro del sistema web.

- **Authoring-in-the-Small**

En esta fase, se detallan los contextos de navegación previamente definidos, desglosándolos en Unidades de Información Abstracta (AIU). Estas AIUs se componen de clases de navegación, representando vistas sobre las clases definidas en el diagrama de clases. Cada AIU incluye una clase de navegación principal (clase manager) y, opcionalmente, clases complementarias para enriquecer la información recuperada. Las clases de navegación se relacionan mediante relaciones binarias unidireccionales llamadas relaciones de navegación, que pueden ser de dependencia de contexto o de relación de contexto.

6.4.4. Modelo de presentación

El modelo de presentación en UWE desempeña un papel esencial en el diseño de aplicaciones web, al delinear los elementos lógicos que conforman la interfaz de usuario. Utilizando diagramas de estructura compuesta, este modelo permite la representación de estas estructuras como clases y propiedades UML interrelacionadas. Además, se encarga de especificar los requisitos de presentación en consonancia con el modelo de navegación, apoyándose en los contextos de navegación para definir cómo se presentan las interfaces. (Busch, 2016)

En el contexto específico de la paginación de información, un patrón fundamental en el modelo de presentación, se aborda la necesidad de organizar y presentar grandes conjuntos de datos de manera manejable. Este patrón permite dividir las instancias recuperadas en bloques lógicos, controlando qué bloque es visible en un momento dado y proporcionando mecanismos para avanzar o retroceder en la información. La cardinalidad y el modo de acceso, como secuencial o aleatorio, son elementos

esenciales para configurar este patrón, asegurando una experiencia de usuario fluida y personalizada.

Los patrones de presentación en la construcción de interfaces web efectivas en combinación con las características de navegación, capturan los requisitos esenciales para diseñar interfaces web coherentes y centradas en el usuario. La paginación de información, junto con otros patrones como la circularidad y los criterios de ordenación, se presenta como una herramienta valiosa para abordar los desafíos específicos de presentación en el contexto de la ingeniería web.

6.4.5. Modelo de procesos

La creciente y extendida utilización de la tecnología de servicios web ha posicionado a Internet como la plataforma más adecuada para el desarrollo de aplicaciones empresariales, donde muchas empresas ya ofrecen servicios a terceros a través de esta tecnología. Sin embargo, el uso de estas aplicaciones empresariales presenta desafíos significativos. En ocasiones, la descripción de estas aplicaciones empresariales está estrechamente vinculada a la definición de un proceso empresarial (BP, por sus siglas en inglés), donde el objetivo no se limita a la gestión de información, sino que abarca también la gestión de procesos. Además, los procesos empresariales reales incluyen actividades automatizadas y participantes del sistema, y también participan participantes humanos que requieren una interfaz de usuario para interactuar con el proceso, así como actividades manuales no automatizadas, como llamar por teléfono o revisar un documento.

6.4.6. Modelo de despliegue y componentes

6.4.6.1. Modelo de despliegue en UWE

El Modelo de Despliegue en UWE se centra en la estructuración y organización física de los componentes de un sistema web. Este modelo considera la distribución de los recursos computacionales y la disposición de los componentes en entornos de

ejecución. La clave del éxito de este enfoque radica en garantizar una arquitectura de despliegue eficiente y escalable.

6.4.6.2. Componentes en UWE

- **Servidores web**

Los servidores web constituyen la infraestructura básica para alojar y servir contenido web. Se encargan de gestionar las solicitudes HTTP y proporcionar las respuestas adecuadas.

- **Servidores de aplicaciones**

Los servidores de aplicaciones son responsables de ejecutar y gestionar aplicaciones web complejas, proporcionando un entorno en el que pueden funcionar eficientemente.

- **Bases de datos**

La base de datos almacena y recupera datos necesarios para el funcionamiento del sistema. La elección de un sistema de gestión de bases de datos adecuado es esencial para garantizar la integridad y la eficiencia de la aplicación web.

- **Clientes web**

Los clientes web son los navegadores y dispositivos que acceden al sistema. La compatibilidad con múltiples navegadores y dispositivos es crucial para asegurar una experiencia del usuario consistente.

6.4.7. Consideraciones de diseño

El diseño del modelo de despliegue y componentes debe alinearse con los requisitos de usabilidad del sistema. Es fundamental considerar la escalabilidad, la redundancia y la seguridad para garantizar un funcionamiento óptimo del sistema web.

En síntesis, el modelo de despliegue y componentes en UWE proporciona una guía estructurada para implementar sistemas web usables y eficientes. La correcta disposición de los componentes y su interacción juegan un papel crucial en el logro de la usabilidad deseada.

6.5. Tecnologías

6.5.1. PHP

PHP (acrónimo recursivo de PHP: Hypertext Preprocessor) es un lenguaje de código abierto muy popular especialmente adecuado para el desarrollo web y que puede ser incrustado en HTML.

Lo que distingue a PHP de otro lenguaje en el cliente como Javascript, es que el código es ejecutado en el servidor, generando HTML y enviándolo al cliente. El cliente recibirá el resultado de ejecutar el script, aunque no se sabrá el código subyacente que era. El servidor web puede ser configurado incluso para que procese todos los ficheros HTML con PHP, por lo que no hay manera que los usuarios puedan saber qué se tiene por debajo.

PHP se destaca por su integración fácil con HTML y por ofrecer funciones específicas que aceleran el desarrollo web. Su versatilidad y robustez hacen de PHP una opción confiable para la construcción de sistemas web responsivos y seguros.

6.5.2. Doctrine

Doctrine es una biblioteca de mapeo objeto-relacional (ORM) para PHP que proporciona persistencia transparente para objetos PHP. Utiliza el patrón de diseño del Data Mapper en su núcleo, buscando una separación completa entre tu lógica de dominio/negocio y la persistencia en un sistema de gestión de bases de datos relacionales.

El beneficio de Doctrine para el programador es la capacidad de concentrarse en la lógica de negocio orientada a objetos y preocuparse por la persistencia solo como un

problema secundario. Esto no significa que Doctrine desconsidere la persistencia, pero creen que hay beneficios considerables para la programación orientada a objetos si la persistencia y las entidades se mantienen separadas.

6.5.3. CSS

Las hojas de estilo en cascada (CSS) juegan un papel crucial en la presentación visual de las aplicaciones web. Permiten la separación entre la estructura del contenido y su presentación, facilitando la creación de diseños responsivos y adaptativos. CSS3, la versión más reciente, introduce características avanzadas como transiciones y transformaciones, posibilitando animaciones suaves y efectos visuales atractivos. La modularidad de CSS, combinada con preprocesadores como Sass, agrega flexibilidad al manejo de proyectos grandes y complejos, proporcionando variables, anidamiento y mixins que facilitan la gestión y mantenimiento del código.

CSS, como lenguaje de estilos, es esencial para la creación de interfaces de usuario atractivas y responsivas. La combinación de las capacidades fundamentales de CSS con las características avanzadas de CSS3 ofrece herramientas poderosas para el desarrollo de sistemas web visualmente atractivos y eficientes.

6.5.4. Tailwind

Tailwind CSS se describe automáticamente como un primer marco CSS de utilidad. En lugar de centrarse en la funcionalidad del elemento al que se aplica estilo, Tailwind se centra en cómo se debe mostrar. Esto facilita al desarrollador probar nuevos estilos y cambiar el diseño.

Tailwind CSS se destaca por su enfoque utilitario, proporcionando una metodología eficiente y altamente personalizable para el diseño de interfaces de usuario. Su estructura modular lo convierte en una herramienta valiosa para desarrolladores que buscan eficiencia y flexibilidad en el diseño de sistemas web.

6.5.5. Javascript

Javascript es esencial para la creación de sistemas web interactivos y responsivos. Su capacidad para manipular el Document Object Model (DOM) facilita la creación de interfaces de usuario dinámicas y la gestión de eventos del usuario. La evolución de ECMAScript ha introducido nuevas características como `async/await`, mejorando la eficiencia al gestionar operaciones asíncronas de manera más legible y estructurada. La presencia de bibliotecas y frameworks populares como React, Vue y Angular amplía las posibilidades de desarrollo, permitiendo la construcción de aplicaciones web complejas y eficientes.

Javascript se ha convertido en un pilar fundamental para la interactividad en el desarrollo web. Su evolución constante y la presencia de frameworks modernos ofrecen a los desarrolladores las herramientas necesarias para construir sistemas web eficientes y altamente interactivos.

6.5.6. ReactJS

React (también llamada React.js o ReactJS) es una biblioteca Javascript de código abierto diseñada para crear interfaces de usuario con el objetivo de facilitar el desarrollo de aplicaciones en una sola página. Es mantenido por Facebook y la comunidad de software libre. En el proyecto hay más de mil desarrolladores libres.

React intenta ayudar a los desarrolladores a construir aplicaciones que usan datos que cambian todo el tiempo. Su objetivo es ser sencillo, declarativo y fácil de combinar. React sólo maneja la interfaz de usuario en una aplicación; React es la Vista en un contexto en el que se use el patrón MVC (Modelo-Vista-Controlador) o MVVM (Modelo-vista-modelo de vista). También puede ser utilizado con las extensiones de React-based que se encargan de las partes no-UI (que no forman parte de la interfaz de usuario) de una aplicación web.

6.5.7. MySql Server

MySQL, como sistema de gestión de bases de datos relacional, contribuye significativamente a la eficiencia y seguridad de los sistemas web. Su capacidad para manejar grandes volúmenes de datos de manera eficiente lo convierte en una elección ideal para aplicaciones escalables. MySQL ofrece funciones avanzadas como la replicación, el particionamiento y la indexación eficiente, lo que mejora el rendimiento de las consultas y la disponibilidad del sistema. Además, la autenticación y autorización basada en roles refuerzan la seguridad de la base de datos, proporcionando un entorno confiable para el almacenamiento y recuperación de datos críticos en aplicaciones web.

MySQL, como sistema de gestión de bases de datos, juega un papel crucial en la eficiencia y seguridad de los sistemas web. Su capacidad para gestionar grandes volúmenes de datos y sus características avanzadas lo convierten en una elección sólida para el desarrollo de aplicaciones web escalables y seguras.

CAPITULO 1. ANÁLISIS DEL CONTEXTO ORGANIZACIONAL Y REQUERIMIENTOS DEL SISTEMA

1.1. Entorno organizacional actual del departamento de evaluación de riesgos del CNR

El Consejo Nacional de Rectores (CNR) de Nicaragua, como órgano rector del sistema de educación superior pública del país, ha asumido un papel cada vez más proactivo en la gestión de riesgos y desastres, entendiendo que el conocimiento, la articulación interinstitucional y la responsabilidad social son pilares fundamentales en contextos de vulnerabilidad.

En este marco, se ha consolidado la Comisión Universitaria del CNR para la gestión integral de riesgos y desastres, una instancia técnica y operativa que coordina esfuerzos a nivel nacional para enfrentar los efectos de eventos adversos, tanto naturales como antrópicos. Esta comisión, vigente desde hace aproximadamente quince años, se ha constituido como un referente en la evaluación de daños post-desastre, movilizando recursos humanos especializados provenientes de múltiples universidades.

Dicho esfuerzo permite una respuesta rápida, organizada y con respaldo científico-técnico, involucrando a estudiantes y profesionales de diversas disciplinas en labores de campo orientadas a la recolección de información crítica en las zonas afectadas. El accionar de la comisión ha sido particularmente notorio durante eventos como los huracanes Eta e Iota, donde se desplegaron más de 1,500 evaluadores en territorio nacional.

1.2. Descripción de la situación actual

Durante el levantamiento de información en campo, los formatos impresos pueden deteriorarse por lluvia o humedad, así como extraviarse o presentar inconsistencias derivadas del llenado manual.

Asimismo, una vez concluida la fase de recolección, la información registrada en los formularios debe ser trasladada de forma manual a equipos de cómputo para su

consolidación y análisis, lo que incrementa el riesgo de errores, retrasa la disponibilidad de los datos y dificulta la toma de decisiones oportunas por parte de las autoridades competentes.

Los formularios utilizados en este proceso corresponden a los formatos EDAN oficiales empleados por la Comisión de Universidades del CNR para la evaluación de daños y análisis de necesidades en situaciones de emergencia.

Para efectos de referencia y documentación, los formatos EDAN actualmente utilizados en el proceso de evaluación se presentan en el Anexo O, donde se incluyen los formularios vigentes que sirven como base para la recolección de información en el contexto actual.

1.2.1. Misión de la institución

Coordinar y ejecutar procesos técnicos de evaluación de daños y análisis de necesidades en contextos de desastre, promoviendo una respuesta solidaria, eficiente y basada en evidencia, que contribuya a la gestión del riesgo en Nicaragua y fortalezca la formación integral de la comunidad universitaria.

1.2.2. Visión de la institución

Constituirse como una comisión universitaria líder en la detección y evaluación integral de riesgos en Nicaragua, reconocida por su capacidad técnica, su compromiso social y su aporte al desarrollo de sistemas resilientes ante desastres en el país.

1.3. Organigrama del departamento de evaluación de riesgos del CNR

La Comisión de Universidades del CNR para la gestión integral de riesgos y desastres del CNR opera bajo una estructura organizativa funcional, que articula los niveles de coordinación y ejecución de manera jerárquica. Aunque su organigrama puede variar según ajustes administrativos, en la práctica se mantiene un modelo operativo consolidado que integra a actores desde el nivel central hasta el territorial.

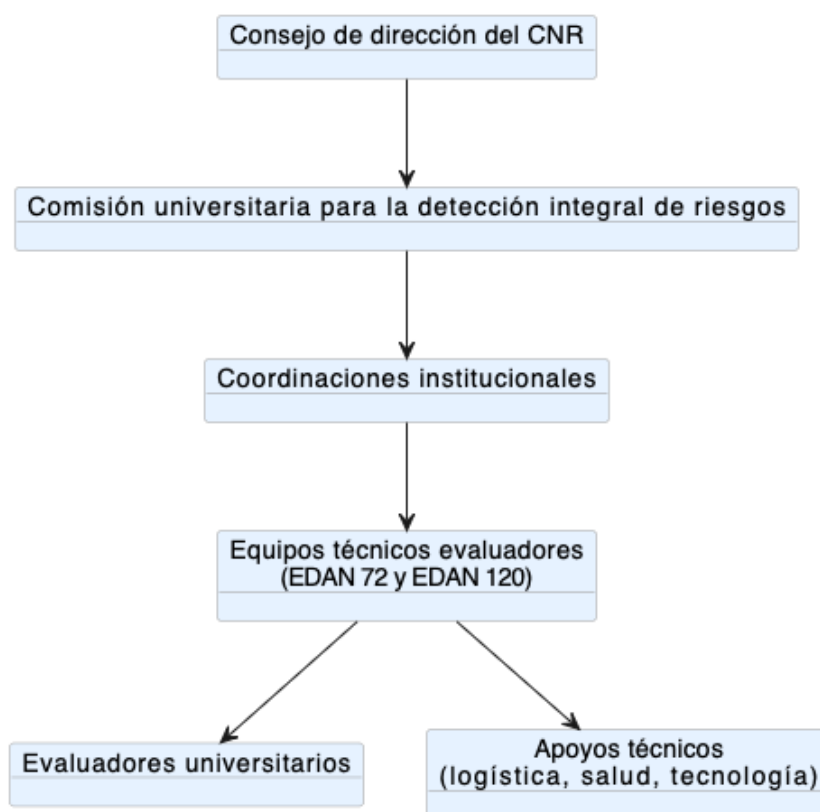
En la cúspide se encuentra el Consejo de Dirección del CNR, responsable de aprobar lineamientos y validar decisiones estratégicas. Este órgano delega la operatividad técnica en la comisión especializada, conformada por personal que coordina directamente con las universidades miembros.

En el nivel intermedio se ubican las coordinaciones universitarias, encargadas de organizar recursos y movilizar los equipos hacia las zonas afectadas. En la base del organigrama están los equipos técnicos evaluadores, integrados por estudiantes y personal académico capacitado. Este grupo, el más numeroso y activo, representa el componente esencial en el trabajo de campo.

A continuación, se presenta una representación gráfica de esta estructura jerárquica:

Figura 2

Organigrama de la Comisión de Universidades del CNR para la gestión integral de riesgos y desastres



Nota. Elaboración propia a partir de información proporcionada por el Dr. Armando Ugarte Solís, mediante comunicación personal (llamada telefónica, mayo de 2025).

1.4. Objetivos del departamento de evaluación de riesgos del CNR

Los objetivos fundamentales de la comisión son amplios y transversales, orientados a brindar una respuesta eficaz, formar talento humano comprometido, y enriquecer el conocimiento práctico institucional. Entre ellos destacan:

- Realizar evaluaciones técnicas en zonas afectadas por desastres, identificando los daños generados por eventos naturales, socio-naturales o provocados por el ser humano, incluyendo daños a infraestructura, servicios, producción, medio ambiente y salud.
- Utilizar herramientas tecnológicas que fortalezcan el proceso de recolección y análisis de datos durante las evaluaciones de campo, contribuyendo a mejorar la eficiencia y calidad de la información recopilada.
- Proporcionar información técnica oportuna al CNR y al SINAPRED, con el propósito de apoyar la toma de decisiones inmediatas en contextos de emergencia.
- Asegurar que los procesos de evaluación puedan desarrollarse eficazmente en zonas rurales o con limitaciones tecnológicas, mediante el uso de soluciones que garanticen la captura y posterior transmisión de datos.
- Fomentar la participación activa de estudiantes universitarios en las evaluaciones de daños, como medio para fortalecer su formación profesional y su compromiso con la realidad social del país.
- Aportar insumos prácticos al desarrollo y actualización de los planes de estudio universitarios, integrando aprendizajes adquiridos en el contexto de evaluaciones reales de desastres.
- Promover el trabajo multidisciplinario en los procesos de evaluación de daños, integrando profesionales y estudiantes de distintas áreas del conocimiento.

1.5. Modelado del negocio

El presente apartado tiene como objetivo describir el funcionamiento actual del proceso de levantamiento y gestión de información ante desastres naturales dentro del entorno universitario, previo a la implementación del sistema propuesto. Este modelado permite entender las actividades clave, los actores involucrados y las principales limitaciones que justifican la necesidad de una solución tecnológica.

En la actualidad, el proceso de evaluación de daños en las universidades ante eventos adversos (como terremotos, inundaciones u otros desastres naturales) se realiza de forma manual. Personal designado en cada institución utiliza los formularios EDAN 72 y EDAN 120 para recolectar información en campo sobre los daños estructurales, afectaciones a bienes, impactos humanos y condiciones generales del entorno. Estos formularios se llenan en formato físico o mediante plantillas digitales dispersas, y posteriormente son enviados al personal encargado para ser analizados y consolidados en reportes.

Este flujo de trabajo presenta múltiples desafíos, entre los que destacan:

- Retrasos en la consolidación de la información, debido al uso de medios manuales o poco estandarizados.
- Falta de trazabilidad y control, ya que no se puede monitorear en tiempo real el estado de los formularios ni quién los ha completado.
- Errores frecuentes en la transcripción de datos, lo que impacta la calidad de los reportes.
- Dificultad para acceder a información histórica o comparativa, debido a la dispersión de los documentos.
- Falta de control centralizado sobre quién puede acceder, editar o eliminar los registros recopilados.

El proceso involucra varios actores clave, entre ellos:

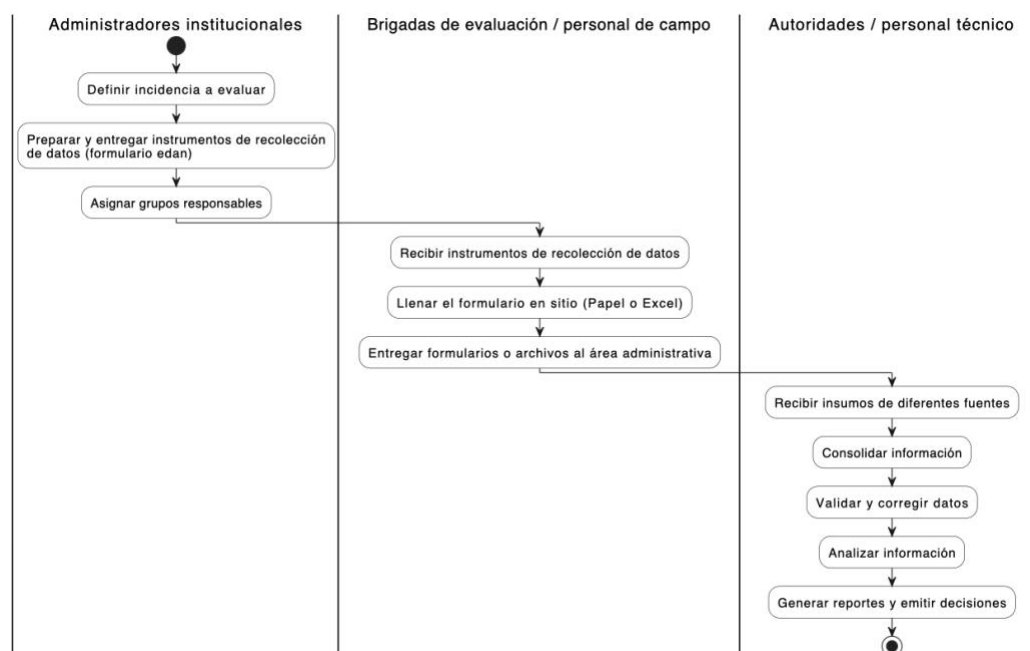
- Administradores institucionales, encargados de asignar formularios y recibir los resultados.

- Brigadas de evaluación o personal de campo, que llenan los formularios EDAN en las zonas afectadas.
- Autoridades o personal técnico, que analizan los datos para tomar decisiones o emitir reportes.

Con el fin de representar gráficamente la secuencia de actividades y la interacción entre los actores involucrados, en la siguiente figura se presenta el diagrama de modelado del negocio correspondiente a la situación actual.

Figura 3

Diagrama del modelado del negocio (Situación actual)



Este panorama evidencia la necesidad de un sistema informático que agilice, centralice y optimice la gestión de las incidencias mediante formularios EDAN. El modelado del negocio sirve, así como punto de partida para el análisis de requerimientos que serán abordados en los apartados siguientes.

1.6. Actores del sistema

Los actores a identificar comprenden todas las entidades que interactúan con el sistema y requieren alguna funcionalidad específica. A continuación, se presentan en detalle.

Tabla 1

Actores del sistema

Actor	Descripción
Administrador	Usuario con privilegios completos sobre el sistema. Puede crear y eliminar incidencias, gestionar usuarios, asignar grupos, gestionar catálogos, y acceder a los reportes.
Evaluador	Usuario asignado a uno o más grupos responsables del llenado de formularios EDAN (72 o 120) para cada incidencia. Tiene acceso limitado a las funciones del sistema según los permisos otorgados.
Consultor	Persona asignada a uno o más grupos responsables del análisis de formularios EDAN (72 o 120) completados o en proceso de completar, y de los reportes que arroje el sistema.

1.7. Descripción de los procesos del negocio

El proceso de recolección de información relacionada con afectaciones en las instituciones de educación superior, producto de eventos naturales adversos, es una labor crítica para la planificación, rehabilitación y toma de decisiones del Consejo Nacional de Rectores (CNR). Actualmente, esta actividad se realiza de forma manual o semiautomatizada, utilizando formularios impresos (EDAN 72 y EDAN 120) o archivos digitales dispersos, los cuales son completados por equipos de campo y luego enviados por correo electrónico u otros medios informales para su procesamiento.

Dichos procesos involucran múltiples etapas: desde la creación y asignación de formularios a grupos de trabajo, el llenado de la información en sitio, la revisión,

consolidación y análisis de los datos. Estas tareas son propensas a errores, demoras y pérdida de información, especialmente en situaciones donde el acceso a equipos de cómputo es limitado o cuando se trabaja bajo presión en entornos afectados por desastres.

Con la implementación del sistema web propuesto, se busca optimizar y centralizar la gestión de incidencias relacionadas a estos eventos, permitiendo:

- A los administradores del sistema, crear incidencias (formularios) y asignarlas a los grupos responsables.
- A los evaluadores, llenar los formularios de forma digital desde computadoras o dispositivos móviles, incluso desde zonas remotas con conectividad limitada.
- A los consultores, evaluar los datos recopilados y generar reportes para apoyar las decisiones estratégicas.

Esta transformación digital permitirá agilizar los procesos actuales, reducir errores en la captura de datos y mejorar significativamente el control, trazabilidad y acceso a la información en tiempo real.

1.8. Requerimientos del sistema

En este apartado se describen los requerimientos fundamentales que debe cumplir el sistema para satisfacer las necesidades identificadas. Se incluyen tanto los objetivos generales que guían el desarrollo del sistema como los requerimientos funcionales y no funcionales que delimitan su comportamiento y características técnicas. Los objetivos representan las metas principales que el sistema debe alcanzar, mientras que los requerimientos definen con mayor detalle las funcionalidades esperadas y las condiciones bajo las cuales debe operar. A continuación, se detallan los objetivos del sistema:

Tabla 2

OBJ-001 - Automatizar formularios EDAN 71 y EDAN 120

OBJ-001	Automatizar formularios EDAN 72 y EDAN 120
Versión	1.0 (13/05/2025)
Autores	- José Ángel Guerrero - José Manuel Dubón - Marcos Antonio Zúniga
Fuente	Dr. Armando Ugarte Solís
Descripción	Implementar una plataforma digital que permita apoyar a completar los formularios EDAN 72 y EDAN 120, optimizando la recolección de datos en situaciones de emergencia y garantizando su almacenamiento estructurado para un análisis posterior eficiente.
Subobjetivos	Ninguno
Importancia	Vital
Urgencia	Inmediata
Estado	Validado
Estabilidad	Alta
Comentarios	Ninguno

Tabla 3

OBJ-002 - Gestionar el acceso al sistema según jerarquías de roles y permisos definidos

OBJ-002	Controlar acceso al sistema
Versión	1.0(13/05/2025)
Autores	- José Ángel Guerrero - José Manuel Dubón - Marcos Antonio Zúniga
Fuente	Dr. Armando Ugarte Solís
Descripción	El sistema deberá permitir autenticar a los usuarios y autorizar su acceso a las diferentes funcionalidades, según los roles y permisos asignados, garantizando una navegación y operación diferenciada para cada tipo de usuario.
Subobjetivos	Ninguno
Importancia	Alta
Urgencia	Inmediata
Estado	Validado
Estabilidad	Alta
Comentarios	Ninguno

Tabla 4

OBJ-003 - Permitir a los administradores crear incidencias y asignarlas a los grupos encargados de recopilar la información

OBJ-003	Permitir a los administradores crear incidencias y asignarlas a los grupos encargados de recopilar la información
Versión	1.0 (13/05/2025)
Autores	- José Ángel Guerrero - José Manuel Dubón - Marcos Antonio Zúniga
Fuente	Dr. Armando Ugarte Solís
Descripción	Facilitar a los administradores del sistema la creación de nuevos registros de formularios (incidencias) y la asignación de dichos formularios a grupos de usuarios responsables del llenado, garantizando una correcta organización y distribución del trabajo.
Subobjetivos	Ninguno
Importancia	Alta
Urgencia	Inmediata
Estado	Validado
Estabilidad	Alta
Comentarios	Ninguno

Tabla 5

OBJ-004 - Facilitar la gestión de catálogos internos del sistema

OBJ-004	Facilitar la gestión de catálogos internos del sistema
Versión	1.0 (13/05/2025)
Autores	- José Ángel Guerrero - José Manuel Dubón - Marcos Antonio Zúniga
Fuente	Dr. Armando Ugarte Solís
Descripción	Proporcionar herramientas que permitan mantener actualizados y organizados los catálogos de datos utilizados por el sistema, tales como universidades, municipios y departamentos, asegurando que la información seleccionada en los formularios sea precisa y estandarizada.
Subobjetivos	Ninguno
Importancia	Media
Urgencia	Normal
Estado	Validado
Estabilidad	Alta
Comentarios	Ninguno

Tabla 6

OBJ-005 - Ofrecer funciones de consulta y reportes para apoyar en la toma de decisiones

OBJ-005	Ofrecer funciones de consulta y reportes para apoyar en la toma de decisiones
Versión	1.0 (13/05/2025)
Autores	- José Ángel Guerrero - José Manuel Dubón - Marcos Antonio Zúniga
Fuente	Dr. Armando Ugarte Solís
Descripción	Incorporar funcionalidades que permitan a los usuarios consultar registros históricos y generar reportes filtrados por diferentes criterios, con el fin de respaldar la evaluación, planeación y respuesta ante eventos reportados.
Subobjetivos	Ninguno
Importancia	Alta
Urgencia	Alta
Estado	Validado
Estabilidad	Alta
Comentarios	Ninguno

Tabla 7

OBJ-006 - Centralizar la configuración de usuarios, roles y permisos para facilitar su gestión y mantenimiento

OBJ-006	Gestión centralizada de usuarios, roles y permisos
Versión	1.0 (13/05/2025)
Autores	- José Ángel Guerrero - José Manuel Dubón - Marcos Antonio Zúniga
Fuente	Dr. Armando Ugarte Solís
Descripción	El sistema deberá proporcionar un módulo único para la gestión de usuarios, roles y permisos, permitiendo la creación, edición, asignación y la desactivación (dar de baja) de perfiles de usuario, así como la administración de los accesos asociados a cada rol dentro del sistema web EDAN.
Subobjetivos	Ninguno
Importancia	Alta
Urgencia	Alta
Estado	Validado
Estabilidad	Alta
Comentarios	Ninguno

1.8.1. Requerimientos funcionales

Los requerimientos funcionales detallan las funciones específicas que el sistema debe ser capaz de realizar. Estas funcionalidades responden directamente a las necesidades de los usuarios y actores del sistema, y están alineadas con los objetivos previamente establecidos.

A través de estos requerimientos se define el comportamiento esperado del sistema en distintos escenarios de uso.

Tabla 8

FRQ-001 - El sistema debe permitir a los administradores crear incidencias en base a los formularios EDAN 72 o EDAN 120

FRQ-001	El sistema debe permitir a los administradores crear incidencias en base a los formularios EDAN 72 o EDAN 120
Versión	1.0 (17/05/2025)
Autores	- José Ángel Guerrero - José Manuel Dubón - Marcos Antonio Zúniga
Fuentes	Dr. Armando Ugarte Solís
Dependencias	OBJ-001
Descripción	El sistema deberá permitir a los administradores registrar nuevas incidencias utilizando los formularios EDAN 72 y EDAN 120 de forma automatizada.
Importancia	alta
Urgencia	inmediata
Estado	en construcción
Estabilidad	media
Comentarios	Ninguno

Tabla 9

FRQ-002 - El sistema debe permitir a los administradores asignar un grupo de usuarios a cada incidencia

FRQ-002	El sistema debe permitir a los administradores asignar un grupo de usuarios a cada incidencia
Versión	1.0 (17/05/2025)
Autores	- José Ángel Guerrero - José Manuel Dubón - Marcos Antonio Zúniga
Fuentes	Dr. Armando Ugarte Solís
Dependencias	OBJ-003
Descripción	Cada vez que se cree una incidencia, el sistema deberá ofrecer la opción de seleccionar el grupo responsable de recopilar la información.
Importancia	alta
Urgencia	inmediata
Estado	en construcción
Estabilidad	media
Comentarios	Ninguno

Tabla 10

FRQ-003 - El sistema debe permitir a los usuarios autorizados llenar los formularios EDAN asignados

FRQ-003	El sistema debe permitir a los usuarios autorizados llenar los formularios EDAN asignados
Versión	1.0 (17/05/2025)
Autores	- José Ángel Guerrero - José Manuel Dubón - Marcos Antonio Zúniga
Fuentes	Dr. Armando Ugarte Solís
Dependencias	OBJ-001
Descripción	Los usuarios con permisos adecuados podrán acceder al formulario que se les haya asignado y completarlo digitalmente.
Importancia	alta
Urgencia	inmediata
Estado	en construcción
Estabilidad	media
Comentarios	Ninguno

Tabla 11

FRQ-004 - El sistema debe permitir a los administradores eliminar incidencias existentes

FRQ-004	El sistema debe permitir a los administradores desactivar incidencias existentes
Versión	1.0 (17/05/2025)
Autores	- José Ángel Guerrero - José Manuel Dubón - Marcos Antonio Zúniga
Fuentes	Dr. Armando Ugarte Solís
Dependencias	OBJ-003
Descripción	Los administradores puedan desactivar registros de formularios (incidencias) cuando sea necesario.
Importancia	alta
Urgencia	inmediata
Estado	en construcción
Estabilidad	media
Comentarios	Ninguno

Tabla 12

FRQ-005 - El sistema debe permitir a los administradores crear, editar y eliminar usuarios

FRQ-005	El sistema debe permitir asignación de roles a usuarios
Versión	1.0 (17/05/2025)
Autores	- José Ángel Guerrero - José Manuel Dubón - Marcos Antonio Zúniga
Fuentes	Dr. Armando Ugarte Solís
Dependencias	OBJ-006
Descripción	El sistema deberá permitir que los administradores asignen y modifiquen los roles de los usuarios, determinando los permisos y accesos disponibles dentro del sistema web EDAN.
Importancia	alta
Urgencia	inmediata
Estado	en construcción
Estabilidad	media
Comentarios	Ninguno

Tabla 13

FRQ-006 - El sistema debe permitir asignar usuarios a uno o más grupos

FRQ-006	El sistema debe permitir asignar usuarios a uno o más grupos
Versión	1.0 (17/05/2025)
Autores	- José Ángel Guerrero - José Manuel Dubón - Marcos Antonio Zúniga
Fuentes	Dr. Armando Ugarte Solís
Dependencias	OBJ-006
Descripción	El sistema deberá permitir la asignación de usuarios a uno o más grupos, facilitando la organización de los responsables por cada incidencia.
Importancia	alta
Urgencia	inmediata
Estado	en construcción
Estabilidad	media
Comentarios	Ninguno

Tabla 14

FRQ-007 - El sistema debe permitir gestionar roles y permisos

FRQ-007	El sistema debe permitir gestión de roles y permisos
Versión	1.0 (17/05/2025)
Autores	- José Ángel Guerrero - José Manuel Dubón - Marcos Antonio Zúniga
Fuentes	Dr. Armando Ugarte Solís
Dependencias	OBJ-002
Descripción	El sistema deberá permitir que los administradores creen, editen y desactiven roles, así como asignen y modifiquen los permisos asociados a cada rol dentro del sistema.
Importancia	alta
Urgencia	inmediata
Estado	en construcción
Estabilidad	media
Comentarios	Ninguno

Tabla 15

FRQ-008 - El sistema debe permitir la administración de catálogos internos

FRQ-008	El sistema debe permitir la administración de catálogos internos
Versión	1.0 (17/05/2025)
Autores	- José Ángel Guerrero - José Manuel Dubón - Marcos Antonio Zúniga
Fuentes	Dr. Armando Ugarte Solís
Dependencias	OBJ-004
Descripción	El sistema deberá incluir un módulo para modificar los catálogos usados en los formularios como universidades, municipios, departamentos, etc.
Importancia	alta
Urgencia	inmediata
Estado	en construcción
Estabilidad	media
Comentarios	Ninguno

Tabla 16

FRQ-009 - El sistema debe validar que los formularios usen únicamente valores activos en los catálogos

FRQ-009	El sistema debe validar que los formularios usen únicamente valores activos en los catálogos
Versión	1.0 (17/05/2025)
Autores	- José Ángel Guerrero - José Manuel Dubón - Marcos Antonio Zúniga
Fuentes	Dr. Armando Ugarte Solís
Dependencias	OBJ-004
Descripción	El sistema deberá validar que, al momento de llenar un formulario, los datos ingresados correspondan a elementos activos del catálogo.
Importancia	alta
Urgencia	inmediata
Estado	en construcción
Estabilidad	media
Comentarios	Ninguno

Tabla 17

FRQ-010 - El sistema debe generar reportes de incidencias filtradas por fecha, ubicación, tipo de evento o universidad

FRQ-010	El sistema debe generar reportes de incidencias filtradas por fecha, ubicación, tipo de evento o universidad
Versión	1.0 (17/05/2025)
Autores	- José Ángel Guerrero - José Manuel Dubón - Marcos Antonio Zúniga
Fuentes	Dr. Armando Ugarte Solís
Dependencias	OBJ-005
Descripción	El sistema debe contar con un módulo de reportes que permita aplicar filtros a los datos de las incidencias almacenadas.
Importancia	alta
Urgencia	inmediata
Estado	en construcción
Estabilidad	media
Comentarios	Ninguno

Tabla 18

FRQ-011 - El sistema debe permitir exportar reportes en formato PDF o Excel

FRQ-011	El sistema debe permitir exportar reportes
Versión	1.0 (17/05/2025)
Autores	- José Ángel Guerrero - José Manuel Dubón - Marcos Antonio Zúniga
Fuentes	Dr. Armando Ugarte Solís
Dependencias	OBJ-005
Descripción	El sistema debe permitir exportar cualquier reporte generado a formato PDF y a formato Excel. La funcionalidad estará accesible desde la interfaz de visualización de reportes.
Importancia	alta
Urgencia	inmediata
Estado	en construcción
Estabilidad	media
Comentarios	Ninguno

1.8.2.Requerimientos no funcionales

Los requerimientos no funcionales especifican las condiciones y restricciones que debe cumplir el sistema en términos de calidad, seguridad, rendimiento, escalabilidad y usabilidad, entre otros aspectos. Aunque no describen funciones directas del sistema, son esenciales para garantizar un funcionamiento eficiente, seguro y confiable en su entorno operativo.

Tabla 19

NFR-001 - Compatibilidad con navegadores modernos

NFR-001	Compatibilidad con navegadores modernos
Versión	1.0 (17/05/2025)
Autores	- José Ángel Guerrero - José Manuel Dubón - Marcos Antonio Zúniga
Fuentes	Dr. Armando Ugarte Solís
Dependencias	OBJ-001
Descripción	El sistema debe ser accesible desde navegadores modernos como Google Chrome, Mozilla Firefox y Microsoft Edge, asegurando su correcto funcionamiento y visualización.
Importancia	Alta
Urgencia	Inmediata
Estado	En construcción
Estabilidad	Alta
Comentarios	Ninguno

Tabla 20

NFR-002 - Autenticación segura

NFR-002	Autenticación segura
Versión	1.0 (17/05/2025)
Autores	- José Ángel Guerrero - José Manuel Dubón - Marcos Antonio Zúniga
Fuentes	Dr. Armando Ugarte Solís
Dependencias	OBJ-002
Descripción	El sistema debe proteger el acceso mediante un mecanismo de autenticación con usuario y contraseña para garantizar la seguridad de la información.
Importancia	Alta
Urgencia	Inmediata
Estado	En construcción
Estabilidad	Alta
Comentarios	Ninguno

Tabla 21

NFR-003 - Control de acceso basado en roles

NFR-003	Control de acceso basado en roles
Versión	1.0 (17/05/2025)
Autores	- José Ángel Guerrero - José Manuel Dubón - Marcos Antonio Zúniga
Fuentes	Dr. Armando Ugarte Solís
Dependencias	OBJ-002
Descripción	El sistema debe implementar un control de acceso que restrinja las funcionalidades según el rol asignado a cada usuario.
Importancia	Alta
Urgencia	Inmediata
Estado	En construcción
Estabilidad	Alta
Comentarios	Ninguno

Tabla 22

NFR-004 - Protección contra ataques comunes

NFR-004	Protección contra ataques comunes
Versión	1.0 (17/05/2025)
Autores	- José Ángel Guerrero - José Manuel Dubón - Marcos Antonio Zúniga
Fuentes	Dr. Armando Ugarte Solís
Dependencias	OBJ-002
Descripción	El sistema debe estar protegido contra ataques comunes de ciberseguridad como inyección SQL, XSS y CSRF para salvaguardar la integridad de los datos.
Importancia	Alta
Urgencia	Inmediata
Estado	En construcción
Estabilidad	Alta
Comentarios	Ninguno

Tabla 23

NFR-005 - Base de datos segura con respaldo

NFR-005	Base de datos segura con respaldo
Versión	1.0 (17/05/2025)
Autores	- José Ángel Guerrero - José Manuel Dubón - Marcos Antonio Zúniga
Fuentes	Dr. Armando Ugarte Solís
Dependencias	OBJ-006
Descripción	La información del sistema debe almacenarse en una base de datos segura que cuente con respaldo periódico para evitar pérdida de datos.
Importancia	Alta
Urgencia	Inmediata
Estado	En construcción
Estabilidad	Alta
Comentarios	Ninguno

Tabla 24

NFR-006 - Registro de acciones en log de auditoría

NFR-006	Registro de acciones en log de auditoría
Versión	1.0 (17/05/2025)
Autores	- José Ángel Guerrero - José Manuel Dubón - Marcos Antonio Zúniga
Fuentes	Dr. Armando Ugarte Solís
Dependencias	OBJ-002
Descripción	El sistema debe registrar en un log de auditoría todas las acciones relevantes como creación, edición y eliminación de datos para fines de trazabilidad.
Importancia	Alta
Urgencia	Inmediata
Estado	En construcción
Estabilidad	Alta
Comentarios	Ninguno

Tabla 25

NFR-007 - Modularidad y escalabilidad

NFR-007	Modularidad y escalabilidad
Versión	1.0 (17/05/2025)
Autores	- José Ángel Guerrero - José Manuel Dubón - Marcos Antonio Zúniga
Fuentes	Dr. Armando Ugarte Solís
Dependencias	OBJ-001
Descripción	El sistema debe ser diseñado de manera modular y escalable para facilitar la integración de futuros módulos o formularios.
Importancia	Alta
Urgencia	Inmediata
Estado	En construcción
Estabilidad	Alta
Comentarios	Ninguno

Tabla 26

NFR-008 - Adaptabilidad a dispositivos

NFR-008	Adaptabilidad a dispositivos
Versión	1.0 (17/05/2025)
Autores	- José Ángel Guerrero - José Manuel Dubón - Marcos Antonio Zúniga
Fuentes	Dr. Armando Ugarte Solís
Dependencias	OBJ-001
Descripción	El sistema debe poder adaptarse a diferentes tamaños de pantalla y dispositivos para ofrecer una experiencia de usuario adecuada en computadoras, tablets y smartphones.
Importancia	Alta
Urgencia	Inmediata
Estado	En construcción
Estabilidad	Alta
Comentarios	Ninguno

CAPITULO 2. ANÁLISIS DEL SISTEMA

2.1. Descripción de los procesos principales

Esta sección detalla los procesos clave del sistema web para la Comisión Universitaria del CNR, abarcando desde la asignación de roles y gestión de equipos hasta la digitalización de formularios EDAN 72 y 120.

2.1.1. Diagrama de actividades

Se presentan diagramas generales para facilitar la comprensión lógica del sistema.

2.1.1.1. Asignación de un rol

La asignación inicial de roles implica distribuir las responsabilidades de evaluación y apoyo técnico a los distintos equipos, considerando la especialidad de cada participante y las zonas que deben evaluarse.

Figura 4

Diagrama de actividad del proceso principal de asignar roles

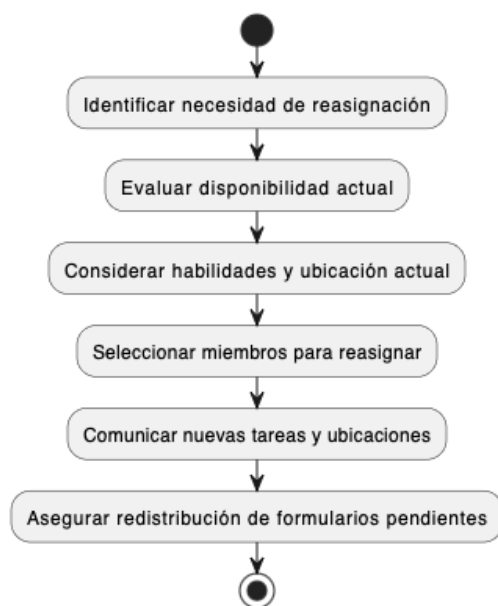


2.1.1.2. Reasignación de roles

Este proceso cubre la necesidad de adaptar y reorganizar los equipos ante cambios imprevistos en campo, asegurando siempre la cobertura completa y eficiente de las zonas afectadas.

Figura 5

Diagrama de actividad del proceso principal de reasignar roles

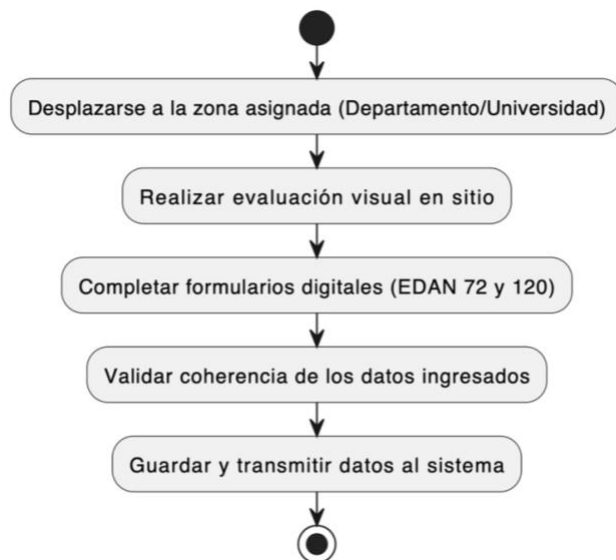


2.1.1.3. Recolección de datos

Este diagrama representa la actividad crítica de recopilación de datos en terreno, asegurando información detallada y precisa para su posterior análisis.

Figura 6

Diagrama de actividad del proceso principal de recolectar los datos

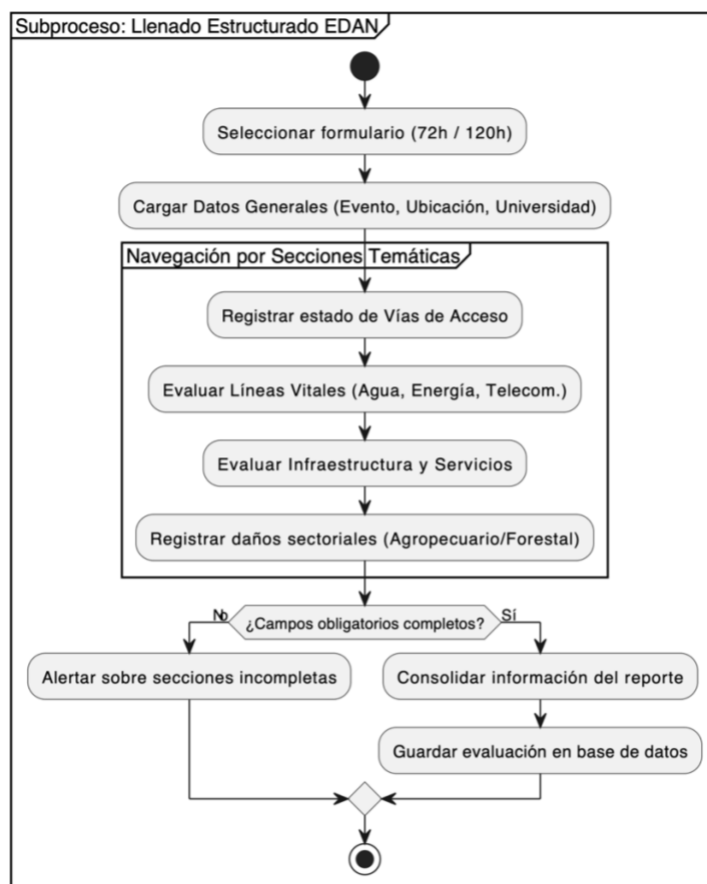


En respuesta a la complejidad y extensión de los formatos originales (EDAN 72 y 120 horas), se ha modelado el subproceso de llenado no como un formulario plano, sino como un flujo estructurado por secciones temáticas. Esto permite al evaluador gestionar la gran cantidad de variables (desde líneas vitales hasta daños agropecuarios) de manera organizada.

La figura 6.1 detalla este subproceso, mostrando cómo el sistema desglosa la estructura de los formatos físicos en una secuencia lógica de captura de datos, asegurando que la información crítica del departamento y la universidad sea registrada antes de proceder a los detalles técnicos de los daños.

Figura 6.1

Diagrama de actividad del subproceso de llenado de formularios EDAN



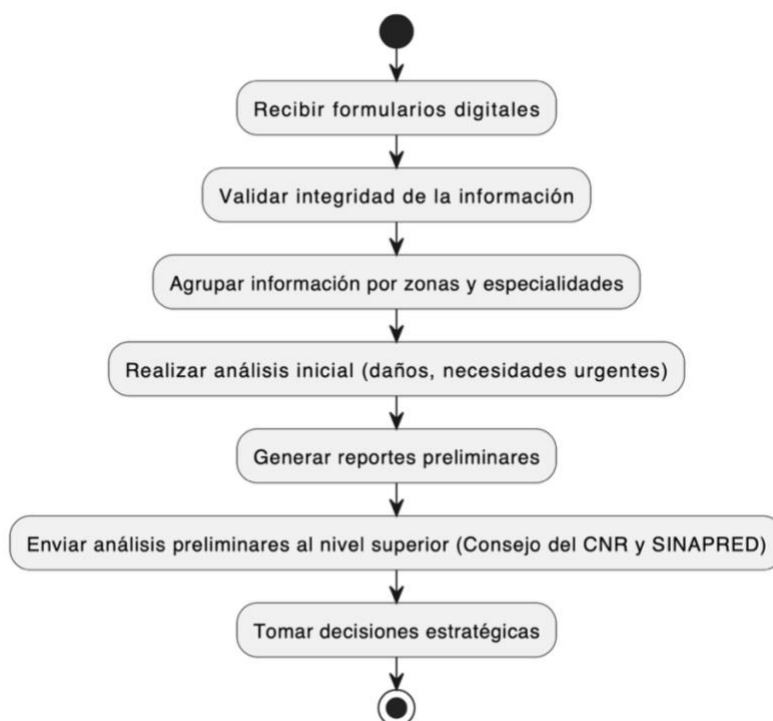
2.1.1.4. Análisis de la información recolectada en los formatos EDAN

Esta actividad detalla el proceso de análisis inicial y validación de la información recogida, para posteriormente derivar conclusiones que faciliten la toma de decisiones estratégicas a nivel institucional. Una vez que los datos llegan desde las zonas afectadas, son organizados, revisados y filtrados por los niveles intermedios de la comisión, quienes se encargan de generar reportes preliminares y trasladar la información más relevante a las instancias superiores como el Consejo del CNR y el SINAPRED.

Como parte de esta monografía, se propone la incorporación de un módulo específico dentro del sistema web que facilite este proceso de análisis. Dicho módulo permitirá validar automáticamente los datos recibidos, generar visualizaciones interactivas, elaborar reportes agrupados por zonas y sectores afectados, y entregar insumos clave para la toma de decisiones. Esta funcionalidad será esencial para transformar la información recolectada en campo en conocimiento útil, oportuno y accionable.

Figura 7

Diagrama de actividad del proceso principal del análisis de la información



2.2. Factibilidad técnica

Este apartado analiza la factibilidad técnica para la implementación del sistema web destinado a la creación, llenado y gestión de formularios EDAN (Evaluación de Daños y Análisis de Necesidades) en las universidades públicas de Nicaragua, bajo la administración del Consejo Nacional de Rectores (CNR). Se detallan los recursos tecnológicos actuales, los requerimientos necesarios, así como aspectos de software, hardware, conectividad, seguridad, compatibilidad y mantenimiento.

2.2.1. Documentación técnica

La presente monografía incorpora documentación técnica fundamental para la comprensión, mantenimiento y escalabilidad del sistema web propuesto. Entre los elementos desarrollados se incluyen:

- Diagramas UWE que modelan los requerimientos, la navegación, el contenido y los procesos del sistema. Estos se desarrollan más a detalle en el Capítulo IV del presente documento.
- Diagrama Entidad-Relación (DER) que representa la estructura de la base de datos y las relaciones entre sus entidades.
- Documentación de la API implementada con Swagger, que describe de forma estandarizada los endpoints, parámetros, respuestas y códigos de estado, facilitando la integración y el mantenimiento.

Estos componentes proveen una visión integral de la arquitectura y funcionamiento del sistema, sirviendo como referencia para el personal técnico encargado de su soporte y evolución.

Si bien se ha documentado ampliamente la estructura y lógica interna, se reconoce que la elaboración de manuales de usuario y de instalación sería un complemento necesario para garantizar la adopción eficiente del sistema por parte de los usuarios finales y el equipo de TI. La preparación de estos materiales se recomienda como una etapa posterior a la entrega del presente proyecto.

2.2.2. Infraestructura tecnológica actual, requerimientos de hardware y evaluación de compatibilidad

Con el objetivo de garantizar la correcta implementación y operación del sistema web propuesto, se realizó un análisis de la infraestructura tecnológica disponible en el Consejo Nacional de Rectores (CNR), junto con la definición de los requerimientos mínimos y recomendados necesarios para su futuro despliegue en el entorno de producción. Este análisis permitió evaluar la compatibilidad entre los recursos existentes y las necesidades técnicas del sistema.

Dado que el sistema adopta una arquitectura web cliente–servidor, la evaluación se centró principalmente en la infraestructura del servidor, responsable de ejecutar el backend, gestionar la base de datos y atender las solicitudes de los usuarios. No se requiere la instalación de componentes adicionales en los equipos de los usuarios finales, ya que el acceso se realiza mediante un navegador web, lo que simplifica su adopción y uso a nivel institucional.

A partir del análisis técnico se establecieron tres escenarios de comparación: la infraestructura actual del CNR, los requerimientos mínimos para el funcionamiento adecuado del sistema y los requerimientos recomendados orientados a un mejor rendimiento y escalabilidad. La Tabla 27 presenta este cuadro comparativo, evidenciando que el servidor del CNR cumple en su totalidad con los requerimientos mínimos y con la mayoría de los recomendados, especialmente en aspectos como memoria RAM, almacenamiento, sistema operativo, servidor web, base de datos, conectividad y mecanismos de respaldo.

La infraestructura tecnológica actual del CNR resulta adecuada para alojar el sistema web propuesto, permitiendo su despliegue inmediato sin necesidad de inversiones significativas en hardware adicional. La presencia de elementos como redundancia energética, enlaces de red dedicados y copias de seguridad periódicas refuerza la confiabilidad del entorno de operación, respaldando la factibilidad técnica del proyecto y garantizando condiciones apropiadas de rendimiento, disponibilidad y seguridad.

Tabla 27

Infraestructura actual, requerimientos mínimos y recomendados

Característica	Servidor Actual CNR	Requerimiento Mínimo	Requerimiento recomendado	Cumple Mínimo	Cumple Recomendado
Procesador	8 núcleos, 2.10 GHz	4 núcleos, 2.4 GHz	8 núcleos a 3.0 GHz o superior	✓	✗
Memoria RAM	16 GB DDR4 ECC	8 GB	16 GB o más	✓	✓
Almacenamiento	1 TB SSD NVMe	100 GB SSD	500 GB SSD NVMe	✓	✓
Sistema Operativo	Ubuntu Server 22.04 LTS	Ubuntu 20.04+	Ubuntu Server 22.04 LTS / Windows Server 2022	✓	✓
Servidor Web	Nginx 1.24 con PHP-FPM 8.2	Nginx 1.20+	Nginx 1.24+ con PHP-FPM 8.2+	✓	✓
Base de Datos	MySQL 8.0	MySQL 5.7+	MySQL 8.0	✓	✓
Conectividad	200 Mbps dedicado	50 Mbps dedicado	Enlace dedicado 100 Mbps o superior	✓	✓
Redundancia Energética	UPS Online 3 kVA	UPS 1.5 kVA	UPS 1.5 kVA	✓	✓
Copias de Seguridad	Incremental diario + Respaldo semanal completo semanal		Respaldo semanal	✓	✓

2.2.3. Software

El sistema será implementado como una aplicación web, lo cual facilita su acceso desde distintos dispositivos conectados a la red institucional y reduce la necesidad de instalaciones locales en las estaciones de trabajo de los usuarios. Esta arquitectura permite un mantenimiento centralizado, actualizaciones eficientes y una experiencia homogénea para todos los usuarios autorizados.

Para su desarrollo, despliegue y operación, se contemplan las siguientes herramientas y tecnologías de software:

- **Backend:** Laravel (framework PHP moderno y robusto, orientado al desarrollo de aplicaciones web estructuradas y seguras).
- **Frontend:** React con TypeScript (biblioteca JavaScript orientada a interfaces dinámicas, con tipado estático para mayor seguridad y mantenibilidad del código).
- **Base de datos:** MySQL (sistema de gestión de bases de datos relacional ampliamente adoptado por su fiabilidad, escalabilidad y compatibilidad con múltiples entornos).
- **Servidor web:** Nginx (servidor web de alto rendimiento, eficiente en el manejo de conexiones concurrentes, y óptimo para aplicaciones modernas gracias a su integración con PHP-FPM).
- **Lenguajes de programación:** PHP, JavaScript (con TypeScript), siendo estos lenguajes base del backend (Laravel) y frontend (React), respectivamente.

Estas herramientas han sido seleccionadas por su madurez, amplio soporte en la comunidad de desarrollo y compatibilidad con infraestructuras actuales del entorno institucional. Además, su uso asegura la escalabilidad y mantenibilidad del sistema en el mediano y largo plazo.

2.2.4. Requisitos de red y conectividad

El correcto funcionamiento del sistema depende de una conectividad estable a internet. Tanto usuarios finales como servidores deberán contar con conexiones adecuadas que permitan cargas y descargas de datos de manera eficiente.

Requisitos mínimos:

- Usuarios: 5 Mbps de descarga y 1 Mbps de subida.
- Servidor: conexión simétrica de 10 Mbps o superior.

Se recomienda priorizar redes cableadas (Ethernet) o redes Wi-Fi de alto desempeño para minimizar interrupciones.

2.2.5. Seguridad técnica

La seguridad de la información es un pilar fundamental, especialmente considerando la naturaleza sensible de los formularios EDAN. El sistema implementará diversas estrategias para mitigar riesgos de acceso no autorizado, pérdida de información o vulnerabilidades:

- Uso de protocolo HTTPS en todas las comunicaciones.
- Políticas de contraseñas seguras y autenticación de usuarios.
- Respaldo periódico de bases de datos.
- Actualización constante de parches de seguridad.
- Monitoreo de actividad sospechosa.

2.2.6. Compatibilidad y requisitos de integración

El sistema propuesto está diseñado para operar en entornos web modernos, lo que asegura su compatibilidad con los navegadores más utilizados actualmente, tales como Google Chrome, Mozilla Firefox, Microsoft Edge y Safari en macOS. Esta compatibilidad garantiza un acceso fluido a las funcionalidades del sistema desde

equipos de escritorio o portátiles que cuenten con sistemas operativos como Windows, Linux o macOS, sin necesidad de configuraciones especiales.

Además, el sistema cuenta con un diseño adaptable (responsive), lo que permite su correcta visualización y uso en dispositivos con distintas resoluciones de pantalla, incluyendo tablets o pantallas proyectadas en sesiones institucionales, aunque su enfoque principal está orientado a equipos de escritorio por el volumen de información a procesar.

En cuanto a los requisitos de integración, se prevé que el sistema pueda conectarse, si es necesario, a otros sistemas internos del Consejo Nacional de Rectores mediante mecanismos como servicios web o API RESTful. Esta integración permitiría, por ejemplo, validar información institucional, sincronizar catálogos o compartir datos de incidencias con otras plataformas analíticas o de respuesta ante emergencias. Asimismo, el sistema permite la exportación de datos en formatos ampliamente utilizados como PDF y Excel, lo que facilita su utilización en herramientas de análisis, presentación o documentación externa.

Estas capacidades aseguran que el sistema no solo sea compatible con el entorno tecnológico actual del CNR, sino que también esté preparado para integrarse con futuras soluciones tecnológicas, favoreciendo la escalabilidad y la interoperabilidad institucional.

2.2.7. Plan de mantenimiento y soporte técnico

Un sistema web exitoso requiere no solo un buen diseño inicial, sino también un plan constante de mantenimiento y soporte. Se establece un esquema de mantenimiento dividido en:

- Mantenimiento preventivo: aplicación de actualizaciones, monitoreo de servidores y optimización de base de datos.
- Mantenimiento correctivo: solución de errores técnicos detectados por los usuarios o durante el monitoreo.

- Mantenimiento evolutivo: incorporación de nuevas funcionalidades de acuerdo con las necesidades emergentes.
- Soporte técnico: canal de atención para resolver dudas y problemas operativos, inicialmente vía correo electrónico y asistencia remota.

2.3. Factibilidad operacional

2.3.1. Necesidades de personal

Para garantizar el funcionamiento óptimo del sistema, se requiere personal especializado en diferentes áreas:

- I. 1-2 desarrolladores (para mantenimiento y actualizaciones del sistema).
- II. 1 administrador de bases de datos (gestión y seguridad de la información).
- III. Personal de campo (encargados de llenar formularios EDAN en las universidades).
- IV. 1-2 capacitadores (para entrenar a usuarios finales).
- V. 1 responsable de soporte técnico (para resolver incidencias y brindar asistencia continua).

La institución cuenta con instancias técnicas que pueden servir como soporte para la gestión del sistema. No obstante, la operación integral del mismo requeriría la designación formal de un área responsable, encargada de su administración, mantenimiento y seguimiento.

El esquema de personal podrá definirse de acuerdo con la capacidad operativa existente. En caso de ser necesario, se prevé la asignación de personal adicional o la contratación de servicios especializados, conforme a las disposiciones administrativas vigentes.

2.3.2. Adiestramiento del personal

El éxito del sistema depende de la capacidad del personal para utilizarlo correctamente. Se deben considerar:

- Talleres presenciales: Para personal del CNR y encargados regionales.
- Materiales digitales: Guías interactivas y videotutoriales para universidades en zonas remotas.
- Sesiones prácticas: Simulaciones de llenado de formularios EDAN en escenarios reales.
- Una capacitación adecuada reduce la resistencia al cambio y asegura la adopción efectiva de la herramienta.

2.3.3. Detalle del costo de utilización de medios técnicos en el adiestramiento del personal

Para garantizar una estimación económica precisa y ajustada a la realidad institucional, los costos asociados a los medios técnicos y humanos se determinaron mediante la evaluación comparativa de diversas propuestas laborales y cotizaciones de equipos, las cuales se detallan íntegramente en el Anexo A.

Tabla 28

Adiestramiento del personal

Medio técnico	Descripción / Cantidad	Costo
Video conferencia (Zoom-Google Meet)	Licencias para sesiones remotas	C\$ 0.00
Recurso humano para creación de videos tutoriales	2 especialistas encargados de la producción audiovisual y diseño instruccional, según los perfiles definidos en la sección 3.3.1	C\$ 6,000.00
Una laptop (computadora portátil)	1 unidad para edición y gestión de materiales de capacitación	C\$ 18,000.00
Total		C\$ 24,000.00

La asignación de recursos financieros para el adiestramiento del personal responde a una lógica de optimización y pertinencia institucional. A continuación, se exponen los criterios técnicos, operativos y económicos que validan las partidas presupuestarias establecidas, asegurando que cada inversión contribuya directamente al éxito del proceso de transferencia de conocimientos y a la sostenibilidad técnica del sistema.

Justificación de las partidas presupuestarias:

- Optimización de recursos (C\$ 0.00): Se ha determinado la viabilidad del uso de la plataforma Google Meet en su versión gratuita. Esta decisión estratégica permite la ejecución de sesiones sincrónicas con los usuarios de las distintas universidades sin comprometer la calidad del adiestramiento ni requerir la adquisición de licencias comerciales, ajustándose rigurosamente al alcance operativo del proyecto.
- Dimensionamiento del Recurso Humano: En estricta correspondencia con los perfiles de competencia definidos en la sección 3.3.1 (Necesidades de personal), se cuantifica la participación de dos especialistas. Este personal es el responsable directo de la transposición didáctica del sistema a formato audiovisual. El costo asignado deriva del análisis de las propuestas laborales presentadas en las figuras 37, 38 y 39.
- Sustentación del Equipamiento: La adquisición de una computadora portátil es indispensable para centralizar la edición de contenidos y la administración del sistema durante la fase de despliegue. Dicha inversión se fundamenta en el análisis de idoneidad técnica y económica de las cotizaciones recopiladas en las figuras 40, 41 y 42.

2.3.4. Plan de capacitación

El plan inicia con una Fase Piloto de 2 semanas, centrada en entrenar al personal técnico del CNR y realizar pruebas controladas en tres universidades clave. Posteriormente, se ejecuta la Fase de Escalamiento durante 1 mes, impartiendo talleres regionales a los encargados de gestión de riesgo. Finalmente, se asegura la

sostenibilidad mediante Soporte Continuo, que incluye actualizaciones vía webinars trimestrales y atención de dudas por WhatsApp y correo.

2.3.5. Eficiencia en el uso de los recursos

El sistema optimizará recursos al:

- Eliminar el uso de papel: Formularios 100% digitales.
- Reducción de hasta 2 días en los tiempos de procesamiento de la información, de acuerdo con el EDAN aplicado.
- Minimizar errores: Validación automática de datos.

2.3.6. Impacto en la productividad

En la siguiente tabla se identifica el impacto positivo en la productividad de los procesos:

Tabla 29

Impacto en la productividad

Mejora	Situación actual	Beneficio de sistema
Rapidez en la recopilación de información	Se debe esperar un periodo de tiempo extenso para poder visualizar la información.	Aumenta la rapidez de visualizar la información una vez el usuario la ingresa.
Precisión de los datos	Ocurren errores en el tipo de datos a ingresar. Se debe realizar esfuerzo humano para organizar la información por EDAN.	Los datos registrados serán precisos con respecto a cada EDAN. Un conjunto de validaciones hace que la información sea correcta.
Toma de decisiones	Se debe realizar un análisis más a profundidad para poder evaluar la información de cada EDAN.	En gran ayuda por parte del Dashboard se pueden tomar decisiones más asertivas.

2.3.7. Ahorro en costos operacionales

El sistema generará ahorros en:

La implementación del sistema web EDAN generará reducciones significativas en los gastos operativos que actualmente afectan al CNR y las universidades, gracias a la automatización de procesos que hoy se realizan de forma manual o semi digitalizada.

El sistema elimina la carga financiera de la documentación física actual, caracterizada por altos costos de impresión y necesidades de almacenamiento (archiveros y espacio). Al digitalizar el proceso, se erradican estos gastos de insumos y logística, generando un ahorro mensual tangible detallado en la tabla financiera.

Tabla 30

Ahorro en materiales y servicios

Material	Precio local	Precio con sistema
Papel	- Caja papel carta: C\$ 1,400 * 4 unidades - Caja papel legal: C\$ 1,700 * 3 unidades	- Caja papel carta: C\$ 1,400 - Caja papel legal: C\$ 1,700
Tinta de impresora	Tóner para 1000 paginas C\$ 1,600 * 2 unidades	Tóner para 1000 paginas C\$ 1,600
Mantenimiento de impresoras	Impresora laser C\$ 3,800	Impresora laser C\$ 3,800
Mantenimiento espacios de almacenamiento (mensual)	C\$ 3,500	C\$ 0
Total	C\$ 21,200	C\$ 8,500

Nota. Respecto a “Mantenimiento espacios de almacenamiento (mensual)”, este costo corresponde a un gasto operativo mensual asociado al resguardo físico de los formularios EDAN, e incluye labores de limpieza, organización, control de humedad y preservación del archivo, así como el uso de áreas físicas dentro de las instalaciones del CNR. Con la digitalización del proceso, este gasto se elimina completamente al trasladar el almacenamiento de la información a medios digitales.

- Eliminación de riesgos por pérdida o deterioro de documentos.

2. Reducción de gastos logísticos

- Situación actual:
 - Transporte de formularios físicos desde universidades regionales hacia el CNR (transporte, combustible y tiempo del personal).
 - Digitación manual de datos en hojas de cálculo centralizadas.
- Ahorro con el sistema digital:
 - La digitalización del proceso evitará los costos derivados del transporte y manejo físico de documentos entre las universidades y el CNR.
 - Eliminación de retrasos por traslado de información.

3. Optimización de recursos humanos

- Situación actual:
 - El personal actualmente destina una parte significativa de su jornada laboral a tareas repetitivas de digitación, verificación y consolidación manual de información.
 - Errores recurrentes que requieren retrabajos.
- Ahorro de recurso humano con el sistema digital:
 - El sistema transforma la dinámica laboral, permitiendo que el personal dedique sus esfuerzos a tareas estratégicas en lugar de procesos manuales repetitivos.
 - Se minimizan los errores en la captura de información, reduciendo los casos que requieren rectificación y reprocesamiento.

2.4. Factibilidad económica

La factibilidad económica permite evaluar el desarrollo del sistema propuesto es viable en términos financieros, considerando la relación entre el esfuerzo necesario, los recursos disponibles y los beneficios esperados. Para ello, se utilizan herramientas de

estimación reconocidas como puntos de función y el modelo de COCOMO II, que ayudan a calcular el costo total del proyecto con base en métricas objetivas.

2.4.1. Puntos de función

La técnica de análisis de puntos de función se utiliza para medir el tamaño funcional del sistema de forma independiente del lenguaje de programación o la tecnología utilizada. A través de esta metodología, es posible estimar la complejidad del sistema y calcular una proyección inicial de líneas de código necesarias para su desarrollo.

2.4.1.1. Puntos de función sin ajustar

En esta etapa se identifican y clasifican las funcionalidades del sistema en cinco tipos:

- Entradas externas
- Salidas externas
- Consultas externas
- Archivos lógicos internos
- Archivos de interfaz externo

Cada función se evalúa según complejidad y se asigna un peso (Anexo B), lo que permite obtener un valor inicial de puntos de función sin considerar factores adicionales de influencia.

Tabla 31

Puntos de función sin ajustar

Computación de métrica de puntos de función							
Descripción	Complejidad						
	Baja		Media		Alta		Total
Entradas	111	3	7	4	2	6	373
Salidas	49	4		5	6	7	238
Consultas	39	3		4		6	117
Archivos		7	110	10	2	15	1130
Interfaces de Programa	79	5	3	7	3	10	446
Total de puntos de función sin ajustar							2304

2.4.1.2. Características generales del sistema

Se consideran 14 características generales que influyen en el esfuerzo de desarrollo, tales como la confiabilidad del sistema, complejidad de las transacciones o flexibilidad de los requerimientos. Cada característica se califica y su sumatoria permite calcular el nivel de influencia del sistema, lo que posteriormente se usará para ajustar los puntos de función.

Tabla 32

Características generales del sistema

Características generales del sistema	
Preguntas	Rango
1. Copias de seguridad y de recuperación fiables	5
2. Comunicación de datos	0
3. Funciones de procesamiento distribuido	2
4. Rendimiento crítico	2
5. Entorno operativo existente y fuertemente utilizado	5
6. Entrada de datos interactiva	4
7. Transacciones sobre múltiples pantallas	2
8. Actualización interactiva de archivos maestros	4
9. Entradas, salidas, archivos o peticiones complejas	4
10. Procesamiento interno complejo	2
11. Código Reutilizable	5
12. Conversión e instalación	0
13. Múltiples instalaciones en diferentes organizaciones	0
14. Facilitar cambios y ser fácilmente reutilizadas	3
Nivel de Influencia ΣFi	38

2.4.1.3. Factor de ajuste

El factor de ajuste se calcula con base en la suma de los valores asignados a las 14 características generales del sistema. Este valor se incorpora a la fórmula que ajusta el total de puntos de función, haciendo que el resultado sea más representativo de la realidad del proyecto y su contexto técnico.

Tabla 33

Factor de ajuste

Factor de ajuste	
Fórmula	$FA = [0.65 + 0.01 * \sum Fi]$
Valor	1.03

2.4.1.4. Puntos de función ajustados

Los puntos de función ajustados resultan de aplicar el factor de ajuste al valor inicial de puntos de función. Este valor refleja de forma más precisa el esfuerzo estimado en función del contexto técnico y organizacional del sistema, y se utiliza como base para cálculos posteriores en la estimación del desarrollo.

Tabla 34

Puntos de función ajustados

Puntos de función ajustados	
Fórmula	$PFA = PFSA * FA$
Valor	2373.12

2.4.1.5. Estimación de líneas de código

A partir de los puntos de función ajustados y del multiplicador correspondiente al lenguaje de programación empleado (Anexo C), se estima la cantidad total de líneas de código que serán necesarias para desarrollar el sistema. Esta métrica será usada posteriormente como insumo para el modelo de esfuerzo COCOMO.

Tabla 35

Estimación total de líneas de código

Estimación del total de líneas de código	
Fórmula	$TLDC = \text{Lenguaje} * PFA$
Valor	47462.4 líneas de código

2.4.2. COCOMO

COCOMO II (Constructive Cost Model) es un modelo de estimación de costos de software que permite proyectar el esfuerzo, tiempo y recursos necesarios para el desarrollo de un sistema. (Panda, Sharma, & Roy, 2018)

Se basa en parámetros como el tamaño del software (medido en miles de líneas de código), el nivel de complejidad y otros factores técnicos organizacionales.

2.4.2.1. Cálculo del esfuerzo

La estimación del esfuerzo se realizó utilizando el modelo COCOMO II, con base en el tamaño del sistema expresado en miles de líneas de código y considerando factores clave del contexto técnico y organizativo del proyecto. Para este cálculo, fue necesario analizar los factores de escala, que representan aspectos como la precedencia del proyecto, flexibilidad, resolución de arquitectura, cohesión del equipo y madurez del proceso.

Además, se estimó el valor del exponente de escala y el producto de los multiplicadores de esfuerzo derivados de los indicadores del producto, como la fiabilidad del software, el volumen del esfuerzo total en personas / mes que permitirá dimensionar adecuadamente los recursos necesarios para el desarrollo del sistema.

Las tablas con los valores asignados y los resultados parciales utilizados en esta estimación se encuentran disponibles en los anexos D, E, F, G y H.

Tabla 36

Estimación del esfuerzo

Estimación del esfuerzo	
Fórmula	$E = A * TLDC^B * \pi EM_i$
Valor	57.17147696 persona - mes

Este valor representa el volumen total de trabajo acumulado necesario para completar el proyecto. En términos prácticos, significa que si una sola persona trabajara a tiempo completo, necesitaría alrededor de 57 meses para finalizarlo. En cambio, si el equipo de desarrollo estuviera conformado por cinco personas, el tiempo estimado se reduciría a aproximadamente 11.43 meses, considerando que la relación entre personal y tiempo no es estrictamente proporcional debido a los costos de coordinación, comunicación y otros factores que pueden influir.

Este esfuerzo calculado no debe interpretarse como tiempo de desarrollo final, sino como una métrica base que, combinada con la estimación de tiempo de desarrollo

(Tdes), permitirá definir la cantidad óptima de personal para completar el proyecto en el plazo planificado.

2.4.2.2. Cálculo del tiempo de desarrollo

El tiempo de desarrollo hace referencia a la duración estimada en meses que se necesitará para completar el proyecto. Este cálculo se realiza en función del esfuerzo estimado y el nivel de paralelismo o productividad que pueda alcanzarse en el equipo de desarrollo.

Tabla 37

Calculo del tiempo de desarrollo

Calculo del tiempo de desarrollo	
Fórmula	$Tdes = 3.67 * ((E)^{(0.28 + (0.002 * \sum Fi))})$
Valor	13.10931919 meses

Con base en el esfuerzo previamente calculado y aplicando la fórmula correspondiente del modelo COCOMO II, se obtuvo un tiempo estimado de desarrollo de 13.11 meses. Este valor representa la duración total prevista para el proyecto desde su inicio hasta la entrega final, considerando todas las fases.

Es importante destacar que este tiempo se estima bajo condiciones óptimas de trabajo y con el equipo necesario para cumplir con el esfuerzo calculado. Factores externos como cambios en los requerimientos, disponibilidad de personal o ajustes en la planificación podrían afectar este estimado. No obstante, este cálculo constituye una base sólida para la planificación y asignación de recursos del proyecto, permitiendo establecer cronogramas realistas y metas alcanzables.

2.4.2.3. Estimación de personal

Se estima el número de personas necesarias para completar el proyecto dentro del tiempo previsto, considerando el esfuerzo total calculado en personas / mes. Esta estimación permite proyectar la carga de trabajo por persona y planificar la asignación de recursos humanos.

Tabla 38

Estimación de personal

Estimación de personal	
Fórmula	$CH = E / Tdes$
Valor	4.361132422 ≈ 5 personas

Tabla 39: Estimación de personal

Este valor sirve como referencia para la conformación del equipo de trabajo, considerando que, dependiendo de la especialización de los roles (desarrolladores backend, desarrolladores frontend, diseñadores, analistas y personal de pruebas), el número de integrantes podría variar ligeramente. La estimación busca optimizar la carga de trabajo y los recursos humanos, asegurando un equilibrio entre productividad, calidad y plazos de entrega.

2.4.3. Análisis costo / beneficio

El análisis costo/beneficio del presente proyecto no se interpreta bajo un esquema de rentabilidad financiera tradicional, debido a su carácter social y a que está orientado a fortalecer la gestión institucional del CNR ante emergencias. En su lugar, se evalúa la eficiencia del gasto, considerando los costos de desarrollo e implementación frente a los ahorros operativos recurrentes, la reducción de tareas manuales y la mejora en la oportunidad de la información para la toma de decisiones.

2.4.3.1. Esfuerzo, tiempo de desarrollo y cantidad de personas por etapas

Aquí se desglosa el esfuerzo estimado, el tiempo de desarrollo y el personal requerido en cada una de las etapas del ciclo de vida del proyecto, lo cual permite una planificación más precisa del trabajo.

Tabla 40

Esfuerzo, tiempo de desarrollo y cantidad de personas por etapa

Etapas	ESF%	ESF	TDES%	TDES	CH(E/TDES)
Estudio preliminar	7%	4.0020	20.32%	2.6641	2
Análisis	17%	9.7192	26.16%	3.4295	3
Diseño y desarrollo	57.52%	32.8832	47.36%	6.2080	5
Prueba e implementación	25.48%	14.5691	26.48%	3.4718	5

2.4.3.2. Distribución del costo de fuerza de trabajo y consumo de energía por etapa

Se detallan los costos asociados a la contratación del personal técnico y al consumo de recursos como energía eléctrica, agrupados por fase del desarrollo (análisis, diseño, programación, pruebas, etc.). Los costos de consumo de energía se detallan más en el Anexo I.

Tabla 41

Distribución del costo de fuerza de trabajo y consumo de energía por etapa

Distribución del costo de fuerza de trabajo y consumo de energia por etapa					
ETAPA	Cantidad	Salario	Tiempo de desarrollo	Total mano de obra	Costo de cada etapa por consumo de energia
Estudio preliminar	2	C\$ 15,000.00	2.6641	C\$ 79,922.80	C\$ 836.50
Análisis	3	C\$ 15,000.00	3.4295	C\$ 154,329.20	C\$ 1,615.25
Diseño y desarrollo	5	C\$ 15,000.00	6.2080	C\$ 465,601.07	C\$ 4,873.12
Prueba e implementación	5	C\$ 15,000.00	3.4718	C\$ 260,382.54	C\$ 2,725.24
Total de inversion de mano de obra + Costos de energia				C\$ 960,235.61	C\$ 10,050.11

2.4.3.3. Costo de materiales

Incluye los gastos relacionados con materiales necesarios para el desarrollo del sistema, tales como dispositivos, impresiones, entre otros insumos físicos o digitales.

Tabla 42

Costo de materiales

Cálculo del costo abastecimiento técnico de materiales.			
Cantidad	Descripción	Precio Unitario	Costo Total
1	Resma de papel	C\$ 170.00	C\$170.00
5	Lapiceros	C\$ 20.00	C\$100.00
5	Cuaderno universitario	C\$ 130.00	C\$ 650.00
1	Caja folder	C\$ 150.00	C\$ 150.00
5	Lapiz mecanico 0.5 mm	C\$ 25.00	C\$ 125.00
10	Mina 0.5 mm	C\$ 10.00	C\$ 100.00
5	Borradores	C\$ 10.00	C\$ 50.00
5	Engrapadora	C\$ 150.00	C\$ 750.00
3	Caja de grapas	C\$ 30.00	C\$ 90.00
1	Tóner o tinta para impresora	C\$ 850.00	C\$ 850.00
2	Memoria USB 32GB	C\$ 250.00	C\$ 500.00
1	Disco duro externo 1TB	C\$ 1,500.00	C\$ 1,500.00
2	Protector de voltaje	C\$ 350.00	C\$ 700.00
2	Cable de red Cat6 (3m)	C\$ 120.00	C\$ 240.00
1	Paños de microfibra (paquete)	C\$ 90.00	C\$ 90.00
1	Limpiador de pantallas	C\$ 150.00	C\$ 150.00
1	Etiquetas adhesivas	C\$ 80.00	C\$ 80.00
5	Carpetas para manuales	C\$ 45.00	C\$ 225.00
Sub Total			C\$ 6,520.00
Iva			C\$ 978.00
Total			C\$ 7,498.00

2.4.3.4. Costo directo

El costo directo contempla todos aquellos gastos que están directamente asociados con el desarrollo del sistema, como salarios del equipo técnico, herramientas de desarrollo o infraestructura. En este caso es la sumatoria de los costos de fuerza de trabajo y consumo de energía por etapa (3.4.3.2.) y los costos de materiales (3.4.3.3.).

Tabla 43

Costos directos

Costos directos	
Formula	$CD = CFT + Cce + CMAT$
Valor	C\$ 977,783.71

2.4.3.5. Costos indirectos

Incluye aquellos gastos que no se relacionan directamente con la programación del sistema pero que forman parte del contexto del proyecto, como servicios administrativos, soporte logístico, etc.

Tabla 44

Costos indirectos

Costos indirectos	
Formula	$CI = CD + 15\%$
Valor	C\$ 146,667.56

2.4.3.6. Costo total de inversión del proyecto

En esta sección se suman todos los costos calculados en apartados anteriores para obtener el monto total de inversión necesario para el desarrollo completo del sistema.

Tabla 45

Costo total de inversión del proyecto

Costo Total de inversión del proyecto	
Formula	$CTP = CD + CI$
Valor Cordobas	C\$ 1,124,451.27
Valor Dolares	USD \$ 30,702.33
Taza de cambio del dolar	36.6243

2.4.3.6.1. Interpretación del costo total de inversión

El costo total calculado corresponde a la inversión inicial necesaria para implementar el sistema web EDAN, considerando los costos directos e indirectos definidos en los apartados previos. Esta inversión habilita la automatización de procesos que actualmente se ejecutan de forma manual o semi digitalizada dentro del flujo institucional, permitiendo una gestión centralizada de la información asociada a los formularios EDAN.

2.4.3.6.2. Análisis costo / beneficio y viabilidad institucional

El análisis costo–beneficio se sustenta en la evidencia presentada sobre la reducción de gastos operativos y la mejora del proceso mediante la automatización de actividades que actualmente se realizan de forma manual o semi digitalizada.

En este marco, la viabilidad se interpreta desde un criterio de costo-efectividad, considerando los beneficios cuantificados y descritos en el documento.

a) Eficiencia en el gasto operativo recurrente (materiales y servicios).

El informe establece un ahorro mensual en materiales y servicios al comparar el esquema tradicional con el sistema digitalizado: el gasto mensual estimado pasa de C\$ 21,200 a C\$ 8,500, según la Tabla 32.

Este resultado se vincula con la eliminación o reducción de rubros asociados a la gestión documental física, tales como impresión, fotocopiado y almacenamiento de formularios, así como con la disminución de riesgos por pérdida o deterioro de documentos.

b) Aprovechamiento de infraestructura disponible y requerimientos de implementación.

La viabilidad técnica se apoya en las especificaciones del servidor institucional (Tabla 27) y en los requerimientos mínimos y recomendados para la implementación (Tabla 28), los cuales en conjunto evidencian la disponibilidad de una plataforma apta para hospedar el sistema.

La comparativa de cumplimiento (Tabla 29) concluye que el servidor actual supera los requerimientos mínimos y cumple con casi todos los valores recomendados, señalando como excepción la velocidad por núcleo del procesador, aspecto que no representa una limitación inmediata y que podría evaluarse a futuro si la demanda de procesamiento se incrementa significativamente.

Asimismo, la revisión técnica indica que el servidor cuenta con capacidades suficientes para hospedar aplicaciones web modernas, recomendando monitoreo posterior a la implementación para proyectar posibles ampliaciones.

c) Reducción de gastos logísticos y del manejo físico de documentación.

En el apartado de ahorro operacional se documenta que, en la situación actual, existe transporte de formularios físicos desde universidades regionales hacia el CNR, con costos asociados a transporte, combustible y tiempo del personal, además de digitación manual en hojas de cálculo centralizadas.

Con la digitalización, se establece que se evitarán los costos derivados del transporte y manejo físico de documentos entre universidades y el CNR, así como la eliminación de retrasos por traslado de información.

d) Impacto en productividad y optimización del recurso humano.

El documento identifica mejoras en productividad asociadas a la rapidez de visualización de la información una vez ingresada, mayor precisión por validaciones y apoyo a la toma de decisiones mediante el dashboard, conforme a la Tabla 31.

De forma complementaria, se describe que, en la situación actual, el personal destina una parte significativa de su jornada a tareas repetitivas de digitación, verificación y consolidación manual, además de retrabajos por errores recurrentes; y que con el sistema se transforma la dinámica laboral hacia tareas estratégicas y se minimizan errores y reprocesamientos.

e) Sostenibilidad operativa por arquitectura web y mantenimiento centralizado.

Se establece que, al tratarse de una plataforma web, el sistema no requiere instalación directa en los dispositivos de los usuarios finales y el proceso de instalación y configuración se concentra en el servidor.

Además, se indica que esta arquitectura reduce la necesidad de instalaciones locales, permitiendo mantenimiento centralizado y actualizaciones eficientes.

CAPITULO 3. DISEÑO BASADO EN UWE

El presente capítulo desarrolla el diseño del sistema a partir de los resultados obtenidos en la fase de análisis, organizando su descripción de manera estructurada y coherente para garantizar la trazabilidad entre los objetivos planteados y la solución propuesta. El contenido se presenta de forma progresiva, abarcando desde la definición de las funcionalidades y la organización de la información hasta la representación de los flujos de interacción, la disposición de la interfaz, los procesos internos y la arquitectura técnica sobre la cual se implementará el sistema.

Para lograr una representación integral y estandarizada, se adopta la metodología UWE (UML-based Web Engineering), la cual extiende el lenguaje de modelado UML con elementos específicos para el desarrollo de aplicaciones web. En el marco de este proyecto, el diseño se organiza en un conjunto de modelos que permiten cubrir distintas perspectivas del sistema, entre los que se incluyen: el modelo de requerimientos, enfocado en identificar actores, casos de uso y funcionales; el modelo de contenido, que describe las entidades de información, sus atributos, relaciones y posibles estados; el modelo de navegación, que define la estructura de acceso y recorrido a través de la aplicación; el modelo de presentación, que especifica la disposición y comportamiento de la interfaz de usuario; el modelo de procesos, que representa la lógica y secuencia de las operaciones; y el modelo de despliegue y componentes, que documenta la arquitectura lógica y física del sistema.

En las siguientes secciones se desarrolla cada uno de estos modelos, incorporando diagramas y descripciones que permiten comprender de manera integral la estructura y el funcionamiento del sistema, asegurando la coherencia entre el diseño propuesto y los requerimientos definidos en capítulos anteriores.

3.1. Modelado de requerimientos

La fase de modelado de requerimientos tiene como objetivo capturar y estructurar las funcionalidades necesarias del sistema, a través de la identificación de actores, casos de uso y sus respectivas interacciones. En este proyecto, se ha seguido la metodología

basada en UWE (UML-based Web Engineering), complementada con el uso de la plantilla Coleman para la especificación detallada de cada requerimiento funcional.

Para estructurar los requerimientos funcionales del sistema de manera clara y trazable, se utilizó dicha plantilla como base. Esta herramienta facilita la identificación y documentación de los elementos clave de cada funcionalidad, incluyendo objetivos, actores, flujos de eventos, reglas de negocio y requerimientos relacionados. Su uso permite una transición más fluida entre la etapa de análisis y el diseño del sistema, sirviendo como puente entre los requerimientos expresados en lenguaje natural y su modelado formal a través de diagramas basados en UML, siguiendo las extensiones definidas por la metodología UWE.

El sistema web está orientado a la recolección estructurada de información a través de formularios diferenciados, enmarcados en el contexto de evaluación de daños y análisis de necesidades. Para apoyar este proceso, el sistema incluye subsistemas complementarios que permiten gestionar catálogos, incidencias, usuarios, roles y permisos.

A través de las plantillas de Coleman se describen los distintos escenarios de los casos de uso del sistema, ofreciendo un panorama detallado de cada funcionalidad implementada. Estas plantillas permiten una documentación estructurada y homogénea de los requerimientos funcionales, facilitando su posterior modelado en diagramas UML bajo la metodología UWE.

Las especificaciones completas de cada funcionalidad documentada mediante la plantilla Coleman se presentan en el Anexo J, clasificadas por módulos funcionales.

3.1.1. Casos de uso

3.1.1.1. Incidencias

El núcleo operativo del sistema gira en torno al registro y seguimiento de incidencias. Estas representan situaciones que deben ser evaluadas con base en criterios técnicos y logísticos, utilizando formatos estandarizados. El sistema permite que cada incidencia esté asociada a uno o más evaluadores, los cuales serán responsables de llenar la información solicitada.

Actualmente las incidencias se enfocan en dos formatos EDAN 120 y EDAN 72, cada uno correspondiente a distintos niveles de detalle y enfoque para la recolección de información. La plantilla seleccionada para una incidencia determina qué estructura y campos se deben completar durante la evaluación.

El sistema proporciona funcionalidades para crear una nueva incidencia, buscar incidencias existentes, ver su contenido, editar información relacionada o eliminar registros en caso necesario dicha gestión tiene su propio caso de uso que se puede encontrar en el Anexo L.

3.1.1.2. Catálogos

Uno de los componentes fundamentales del sistema es la administración de catálogos. Estos catálogos contienen información que alimenta distintos campos dentro de los formularios de evaluación, permitiendo su llenado de forma más ágil y precisa. Para evitar la dependencia de servicios externos y asegurar que los datos estén disponibles y actualizados según los criterios internos, el sistema permite a los administradores registrar, modificar y dar de baja elementos del catálogo directamente en el Anexo L se podrá encontrar un diagrama de caso relacionado con la gestión del catálogo de universidades, pero cuya mecánica es la misma que siguen los demás catálogos del sistema.

3.1.1.3. Usuarios roles y permisos

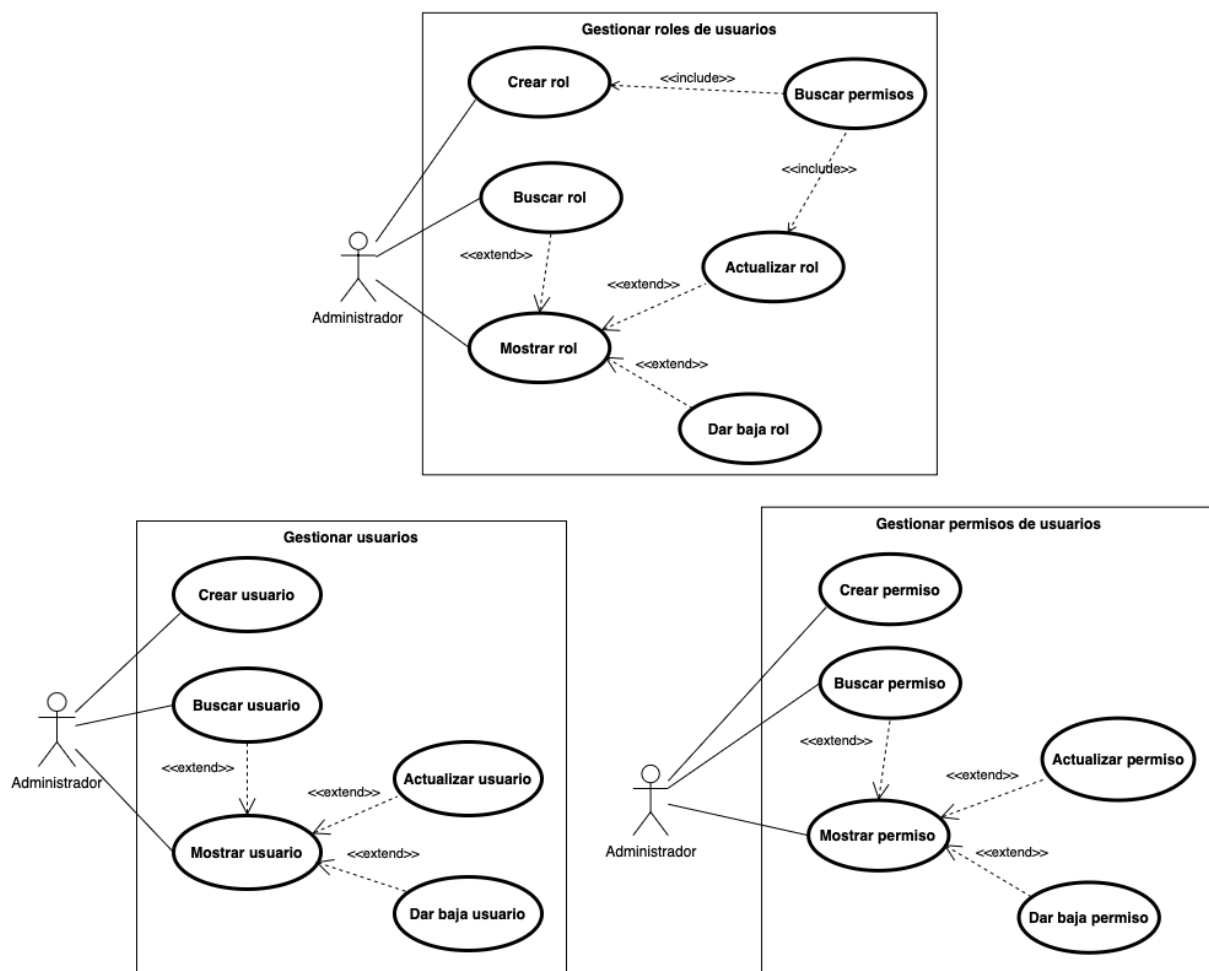
Como parte de los elementos transversales al sistema, se incluye una gestión integral de usuarios y sus privilegios. Se han definido casos de uso clásicos para la administración de roles, permisos y usuarios, siguiendo buenas prácticas en el control de acceso y seguridad.

Los administradores pueden crear nuevos usuarios, asignarles roles, gestionar los permisos asociados a cada rol, y deshabilitar usuarios que ya no requieran acceso al sistema.

Estas funciones permiten mantener una operación segura, ordenada y alineada con la estructura organizativa de la institución que utilizará el sistema.

Figura 8

Caso de uso: Gestión de usuarios, roles y permisos



3.1.2. Diagrama de actividades UWE

Bajo el estándar de la metodología UWE, se presentan los diagramas de actividad que detallan la lógica de los procesos del sistema cada diagrama se abarca dentro del Anexo L.

3.1.2.1. Incidencias

El módulo de incidencias actúa como eje central, documentando mediante diagramas de actividad el flujo operativo desde la creación (vía EDAN 72/120) hasta la búsqueda, edición y eliminación de registros. Cada proceso describe las interacciones del usuario y las validaciones del sistema necesarias para una gestión de datos precisa y segura.

3.1.2.2. Catálogos

El sistema incluye un subsistema de gestión de catálogos, fundamental para alimentar dinámicamente los formularios de evaluación. Se han elaborado diagramas de actividad para las principales operaciones sobre catálogos, centradas inicialmente en el catálogo de universidades. Estos diagramas son:

- **Buscar universidad:** Ilustra el proceso de localización de una universidad específica dentro del catálogo.
- **Actualizar universidades:** Modela la modificación de registros existentes para reflejar cambios en su información.
- **Dar de baja universidades:** Describe el procedimiento para desactivar registros que ya no deben estar disponibles para selección.

Estas actividades aseguran la integridad y vigencia de los datos utilizados por los evaluadores en los formularios.

3.1.2.3. Usuarios, roles y permisos

El control de acceso al sistema se gestiona mediante un conjunto de funcionalidades relacionadas con usuarios, roles y permisos, para los cuales se han modelado los respectivos diagramas de actividades que se podrán encontrar en el Anexo L.

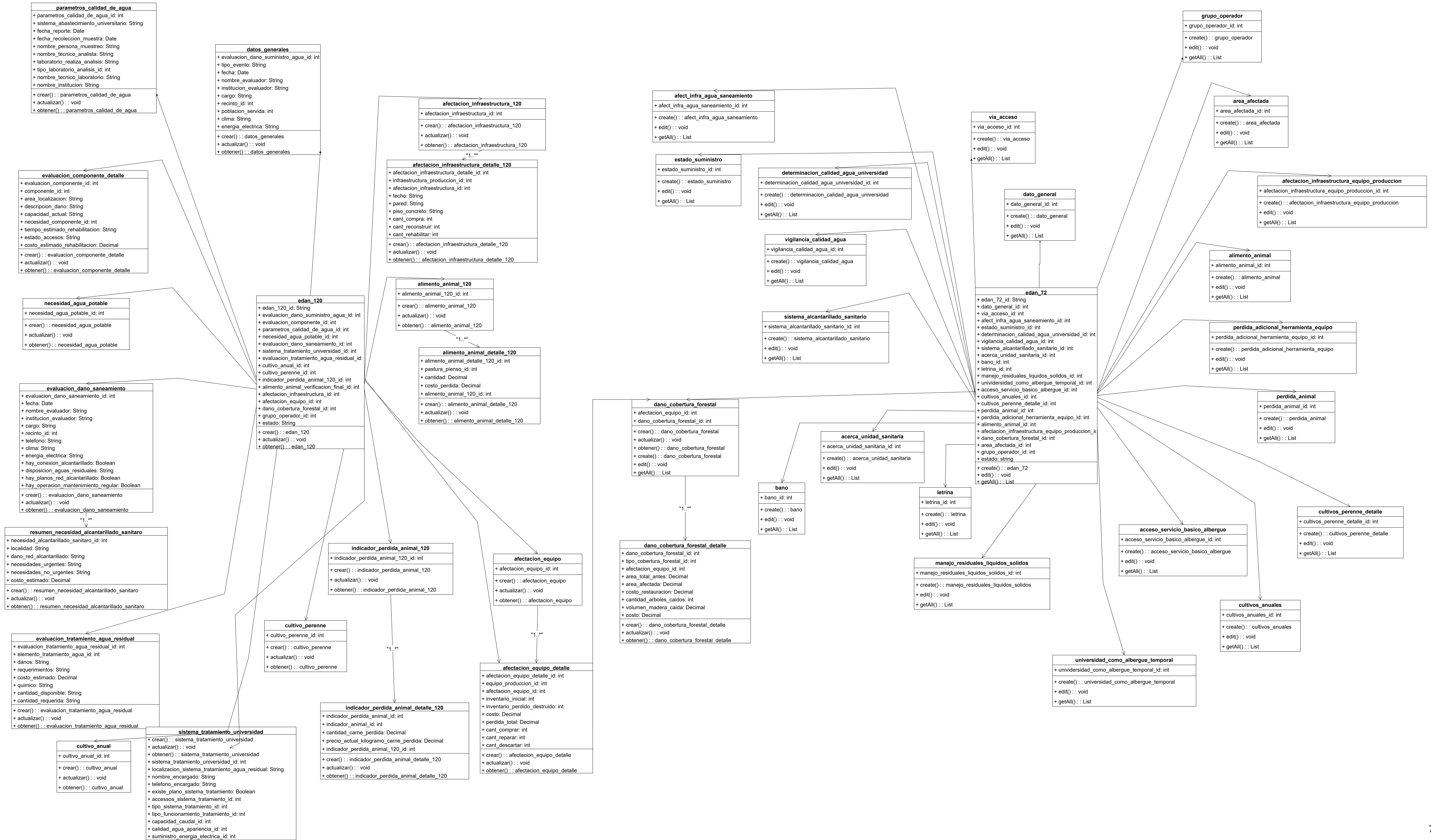
3.2. Modelo de contenido

Basado en la metodología UWE, el modelo de contenido define las estructuras de información y las relaciones del sistema. Esta sección integra el diagrama de clases (organización lógica), el diagrama de estados (comportamiento dinámico de entidades) y el modelo de datos (visión relacional), garantizando una transición precisa desde el concepto hasta la implementación funcional.

3.2.1. Diagrama de clases

Figura 9

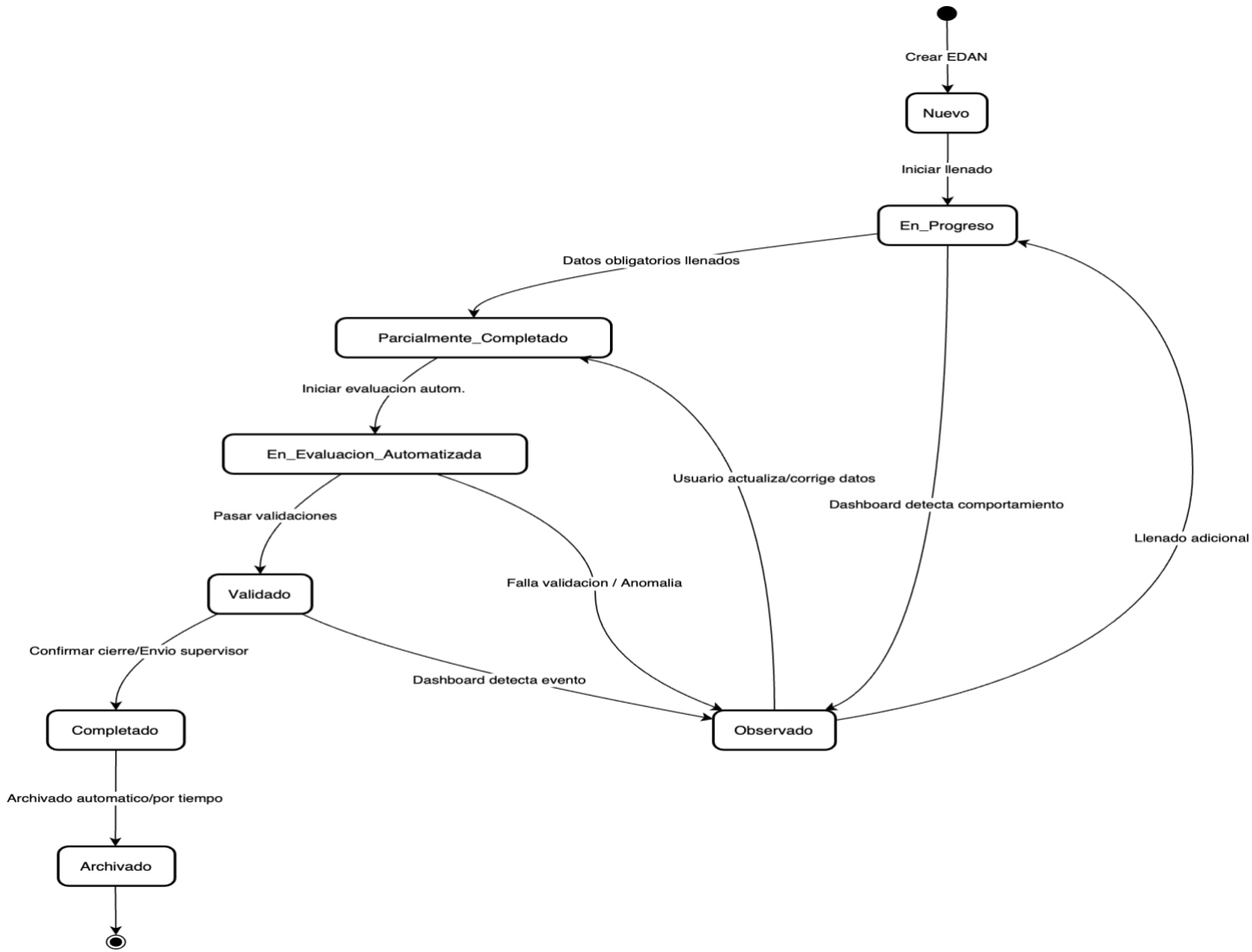
Diagrama de clases



3.2.2. Diagrama de estados

Figura 10

Diagrama de estados



3.2.3. Modelo de datos

El modelo de datos define la estructura relacional que respalda el almacenamiento persistente de la información en el sistema. Para este proyecto se ha utilizado una base de datos MySQL, cuya estructura fue diseñada a partir del modelo de contenido y los requerimientos funcionales identificados.

El sistema organiza su información en torno a dos entidades principales: EDAN120 y EDAN72, que representan los formularios de evaluación utilizados para registrar incidencias. Ambas se relacionan con múltiples catálogos (como universidades, municipios, departamentos, entre otros), lo que garantiza un llenado estandarizado y controlado de los datos. A su vez, cada evaluación se asocia a uno o más usuarios evaluadores, y está sujeta a estados que reflejan su progreso a lo largo del proceso de revisión.

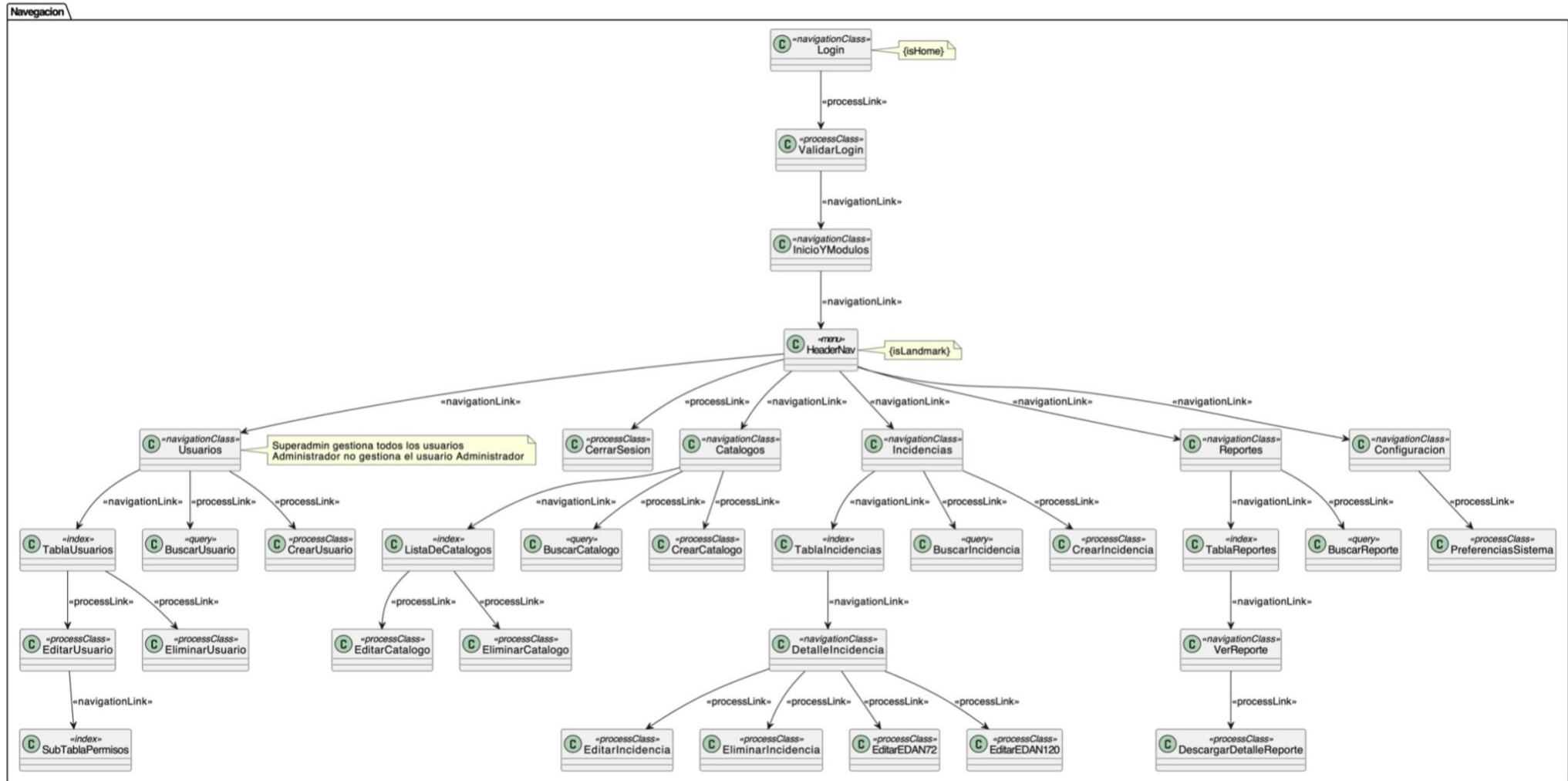
El modelo también incorpora un subsistema de gestión de autenticación, compuesto por tablas para usuarios, roles y permisos, que define qué operaciones puede ejecutar cada tipo de usuario dentro del sistema. Esta estructura permite un control de acceso granular, alineado con la organización y los flujos definidos por la plataforma.

Por criterios de organización y claridad expositiva, el diagrama completo del modelo de datos se ha ubicado en el Anexo K donde se presentan todas las entidades, relaciones, claves primarias y foráneas que componen la estructura lógica del sistema. Esta disposición permite mantener la continuidad del documento y, al mismo tiempo, ofrecer una representación integral de la base de datos para consulta detallada. El modelo garantiza la integridad referencial de los datos, facilitando su validación, trazabilidad y mantenimiento.

3.3. Modelo de navegación

Figura 11

Modelo de navegación por roles (Superadministrador y Administrador)



En la figura anterior se presenta el modelo de navegación del sistema web para los roles de Superadministrador y Administrador, elaborado conforme a la metodología UWE (Unified Web Engineering). Ambos roles comparten la misma estructura de navegación, ya que acceden a los mismos módulos del sistema: Catálogos, Incidencias, Reportes, Usuarios y Configuración.

La principal diferencia entre estos dos roles radica en el control sobre la gestión de usuarios: el Administrador no tiene permisos para gestionar usuarios con rol de Superadministrador, mientras que el Superadministrador posee control total sobre todos los usuarios del sistema. Esta diferencia se indica mediante una nota dentro del propio diagrama de navegación.

El flujo de navegación inicia con el proceso de autenticación (Login), seguido de la validación de credenciales, que permite el acceso a la vista principal de módulos. Desde esta vista, el usuario puede desplazarse hacia los distintos módulos disponibles a través del menú de navegación. Cada módulo incorpora sus respectivos componentes de interacción, tales como:

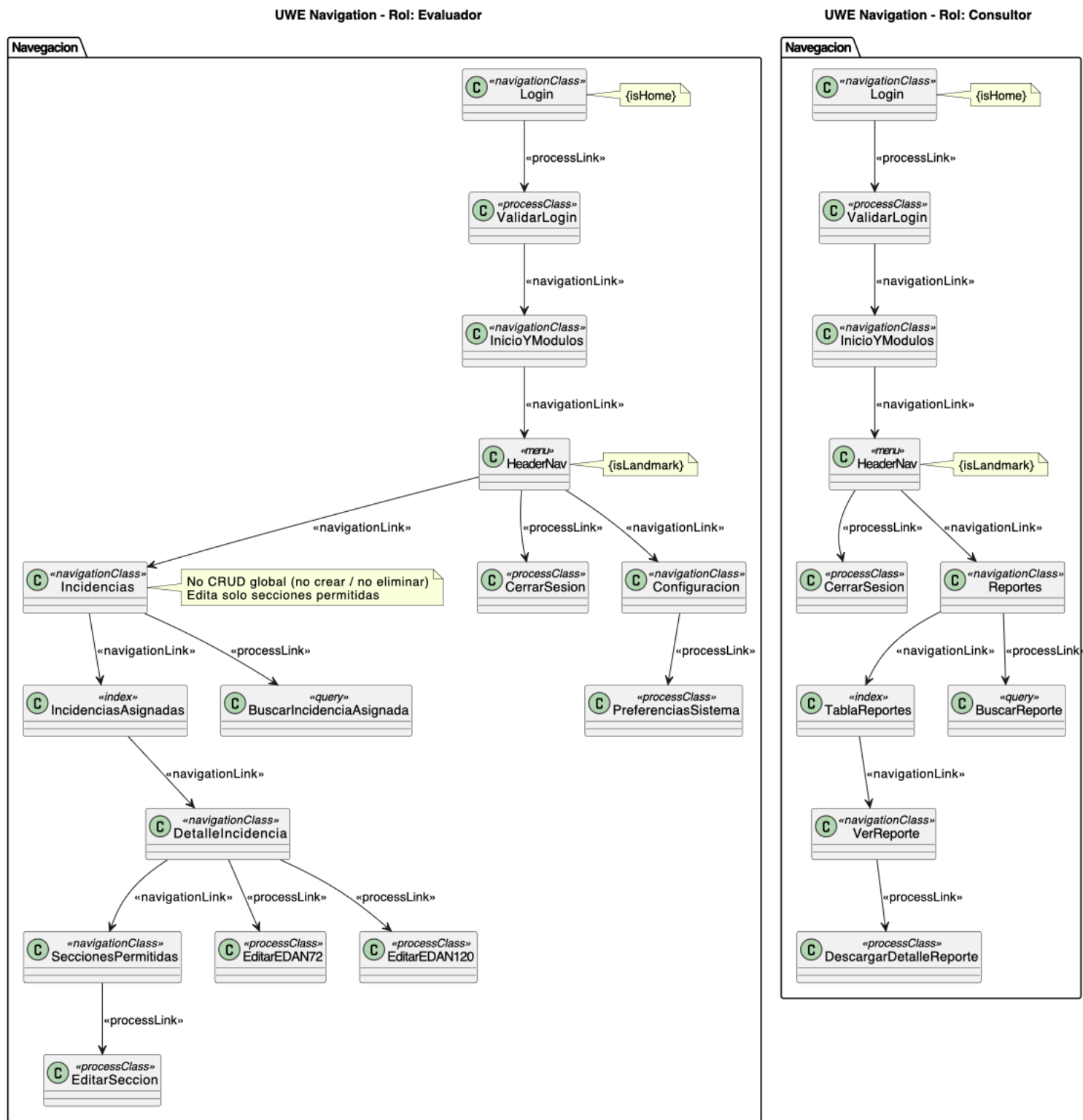
- Listas de elementos (<<index>>), para la visualización de registros.
- Consultas (<<query>>), para la búsqueda y filtrado de información.
- Procesos (<<processClass>>), para acciones como crear, editar o eliminar datos.

Este modelo refleja un enfoque de navegación por roles, asegurando que los usuarios accedan únicamente a las funcionalidades que corresponden a sus permisos dentro del sistema, manteniendo así la jerarquía de control y la seguridad de la información.

Los modelos de navegación correspondientes a los roles de Evaluador y Consultor se presentan en la figura 12, donde se detallan sus flujos de navegación específicos y las restricciones de acceso asociadas a cada uno de estos perfiles.

Figura 12

Modelo de navegación por roles (Evaluador y Consultor)



3.4. Modelo de presentación

Figura 13

Modelo de presentación Login

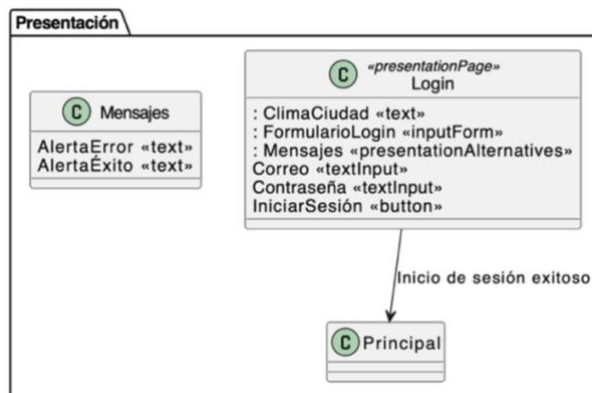


Figura 14

Modelo de presentación Vista de inicio

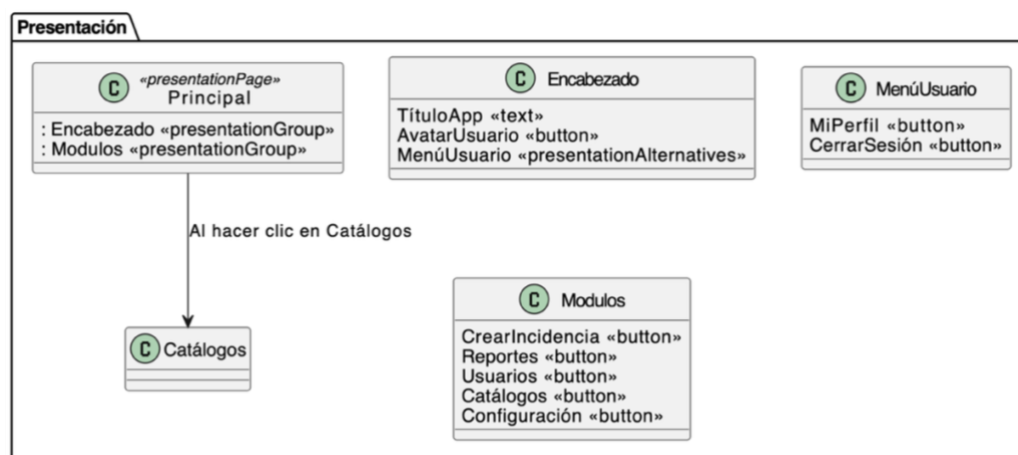
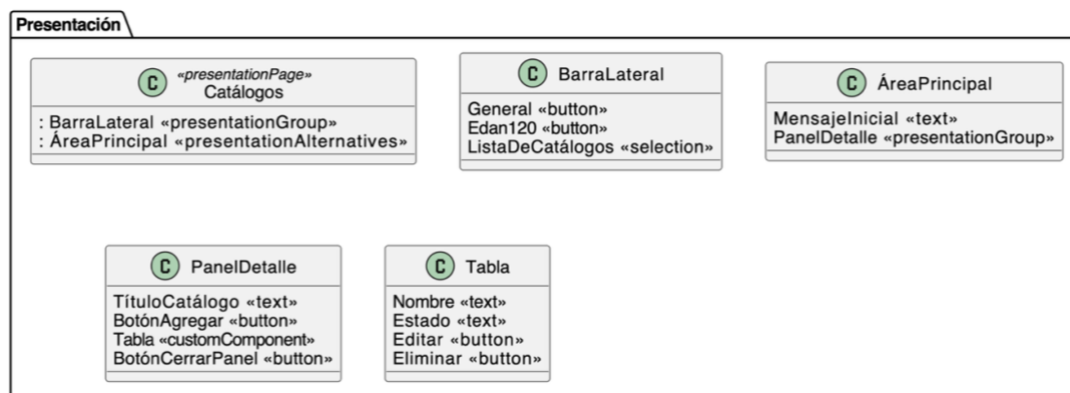


Figura 15

Modelo de presentación Módulo de catálogos



La interfaz final del sistema refleja los elementos definidos en el modelo de presentación. En el Anexo M se incluyen las capturas de pantalla correspondientes a las vistas principales de la plataforma.

3.5. Modelo de procesos

Basado en la metodología UWE, el modelo de procesos detalla la lógica de negocio y la interacción entre actores y sistema. Esta sección se centra en el módulo de incidencias, presentando los diagramas de flujo para las operaciones de búsqueda, creación, edición y eliminación, destacando los puntos de decisión y validación críticos para su ejecución.

Figura 16

Diagrama de proceso - Buscar incidencia

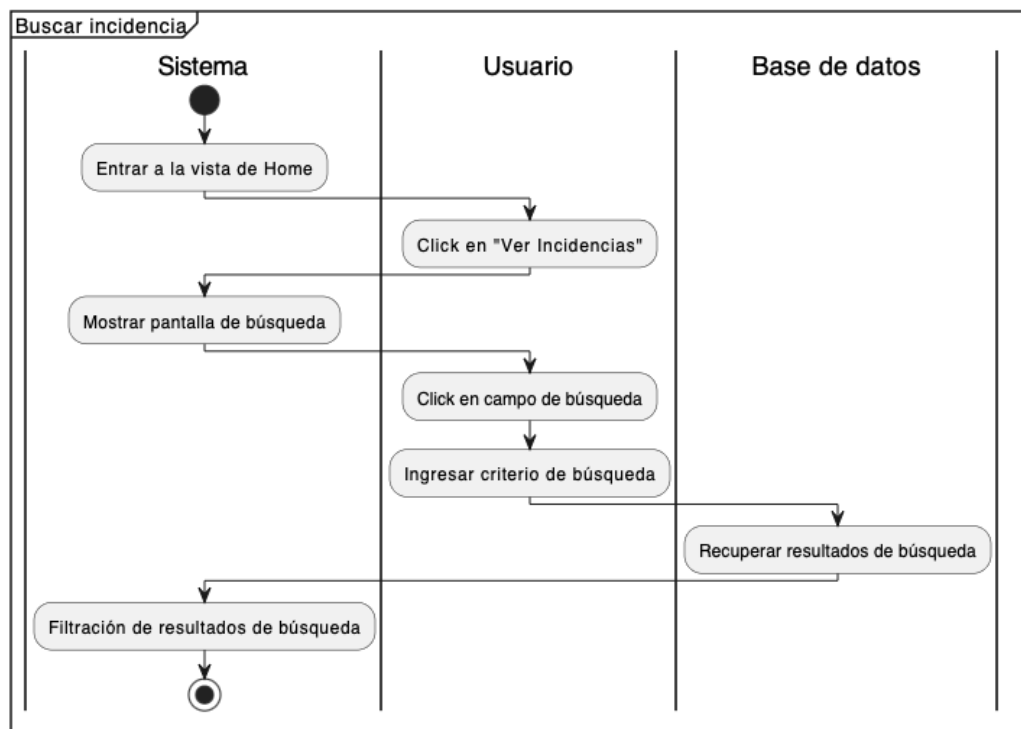


Figura 17

Diagrama de proceso - Crear incidencia

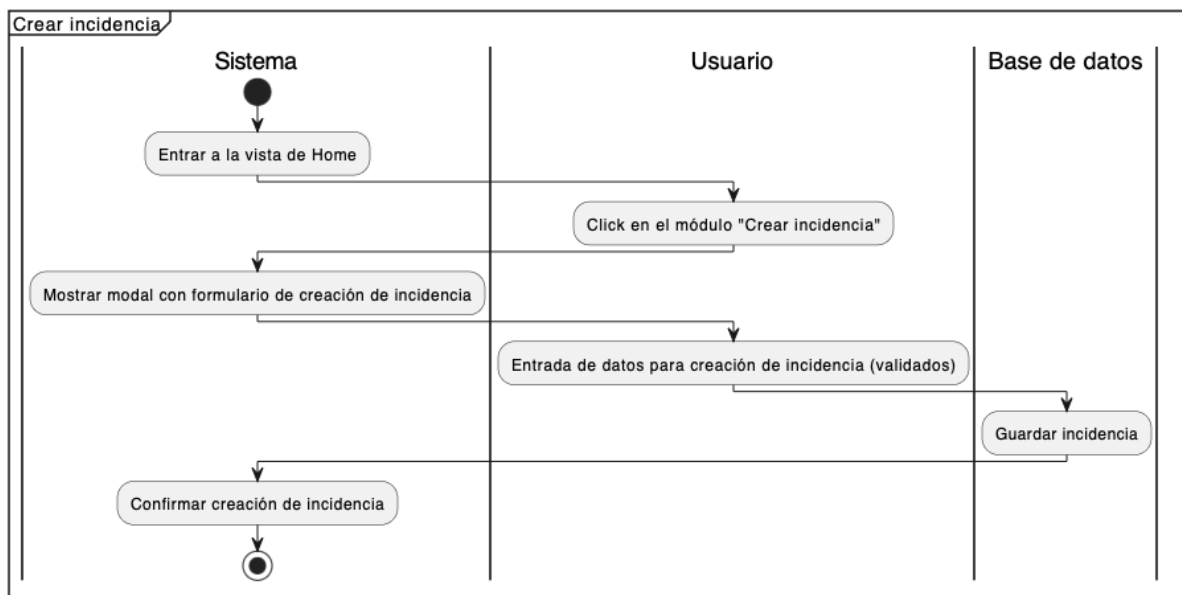


Figura 18

Diagrama de proceso - Editar incidencia

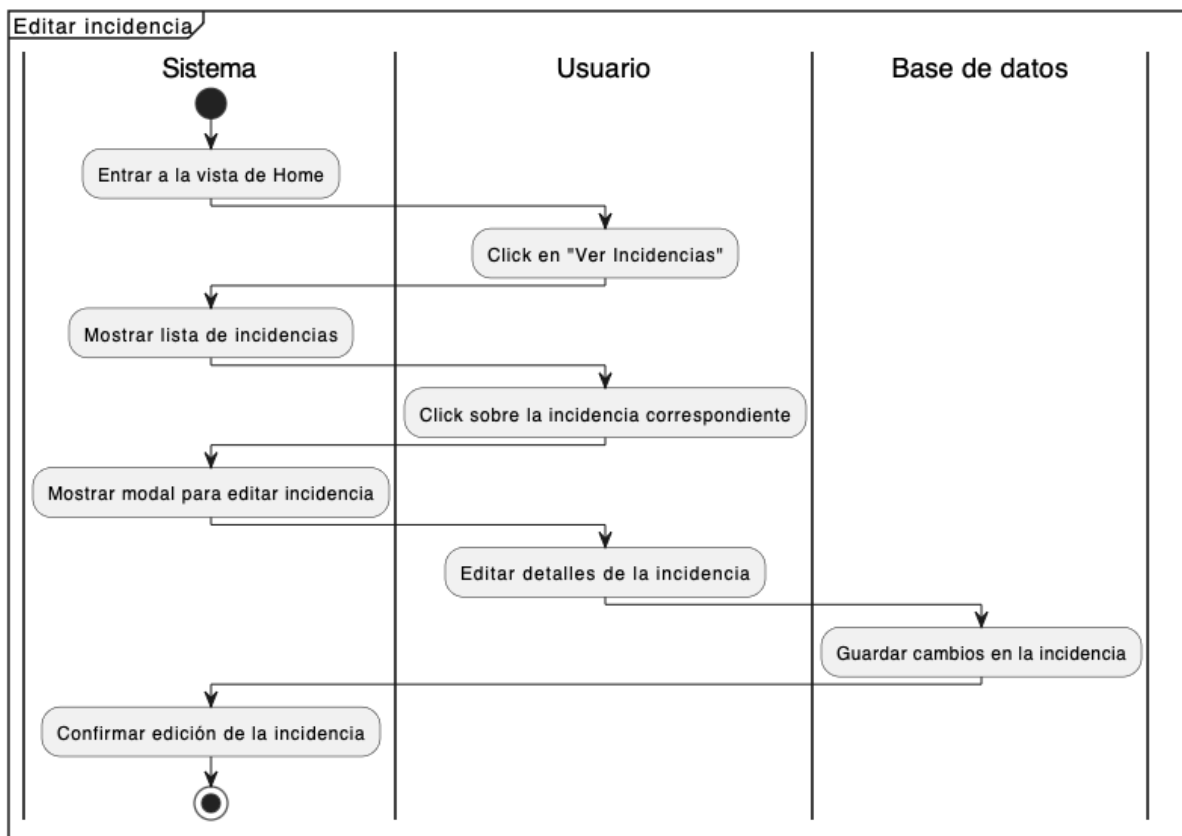
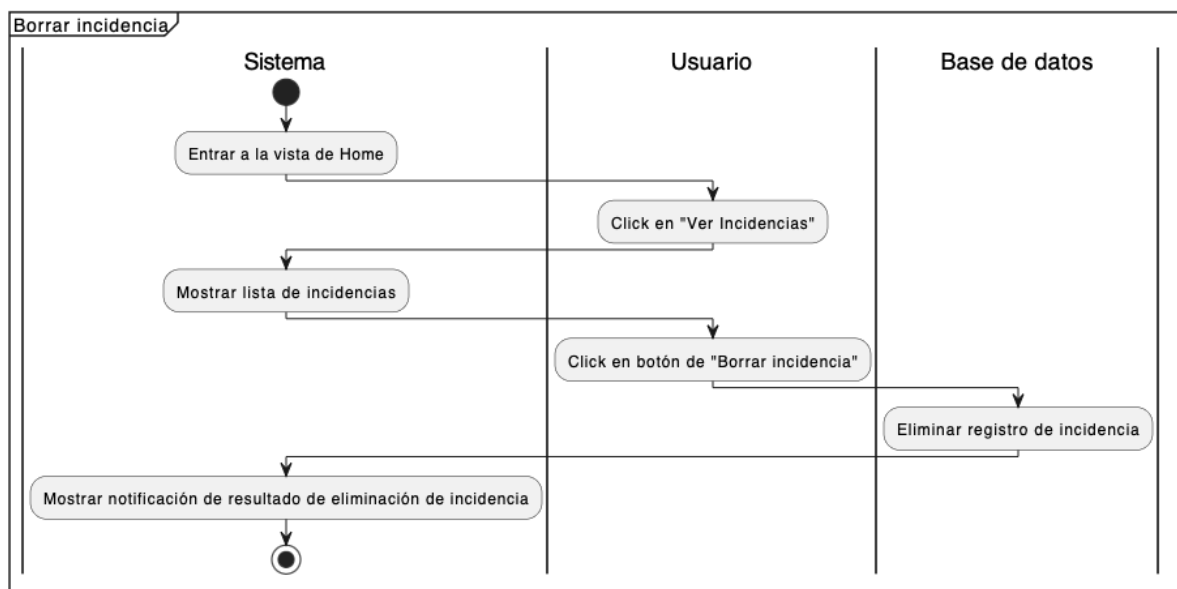


Figura 19

Diagrama de proceso - Eliminar incidencia



Para mantener la claridad expositiva y evitar interrupciones en el desarrollo del capítulo, el resto de los diagramas de procesos correspondientes a los demás módulos funcionales del sistema —catálogos, usuarios, roles y permisos— se han incluido en el Anexo L. Estos diagramas complementan los modelos presentados en los apartados anteriores, documentando de forma precisa los flujos que permiten administrar los elementos transversales del sistema.

3.6. Modelo de despliegue y componentes

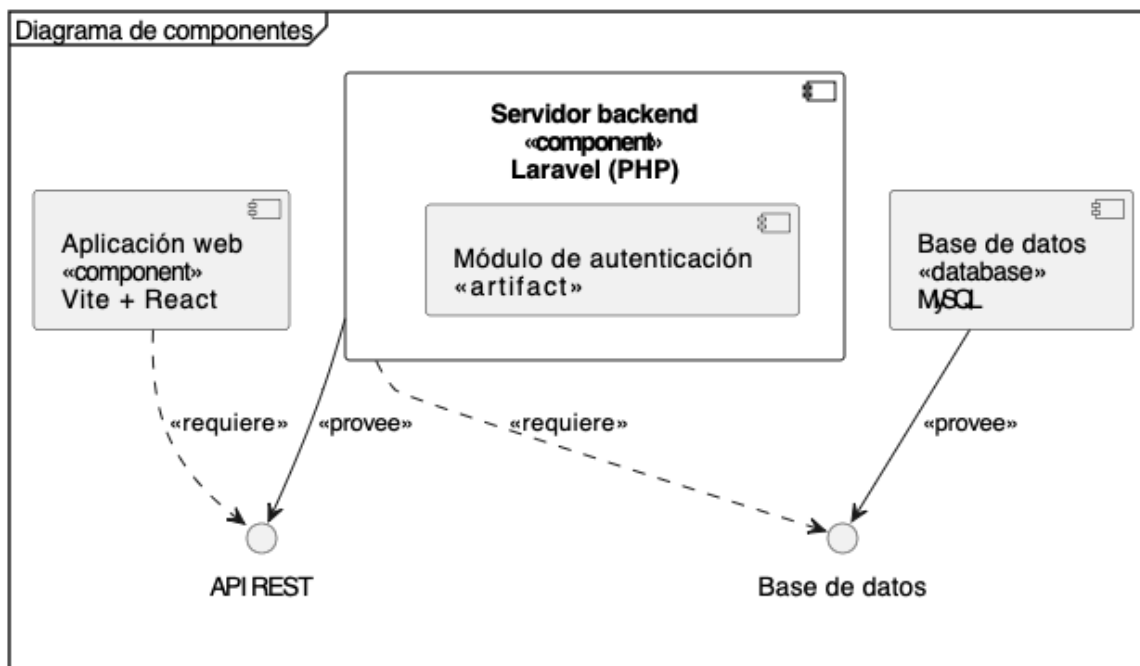
Esta sección integra los modelos que representan la arquitectura lógica y física del sistema, siguiendo las directrices de la metodología UWE. Ambos modelos permiten visualizar cómo se organizan y comunican los diferentes elementos técnicos que conforman la solución web.

3.6.1. Diagrama de componentes

El diagrama de componentes describe la arquitectura lógica de alto nivel, estructurada en tres niveles: un frontend en React (Vite), un backend en Laravel (PHP) y una base de datos MySQL. La integración se realiza mediante una API REST para la comunicación entre cliente y servidor, asegurando la persistencia de datos a través de la conexión directa del backend con MySQL.

Figura 20

Diagrama de componentes del sistema



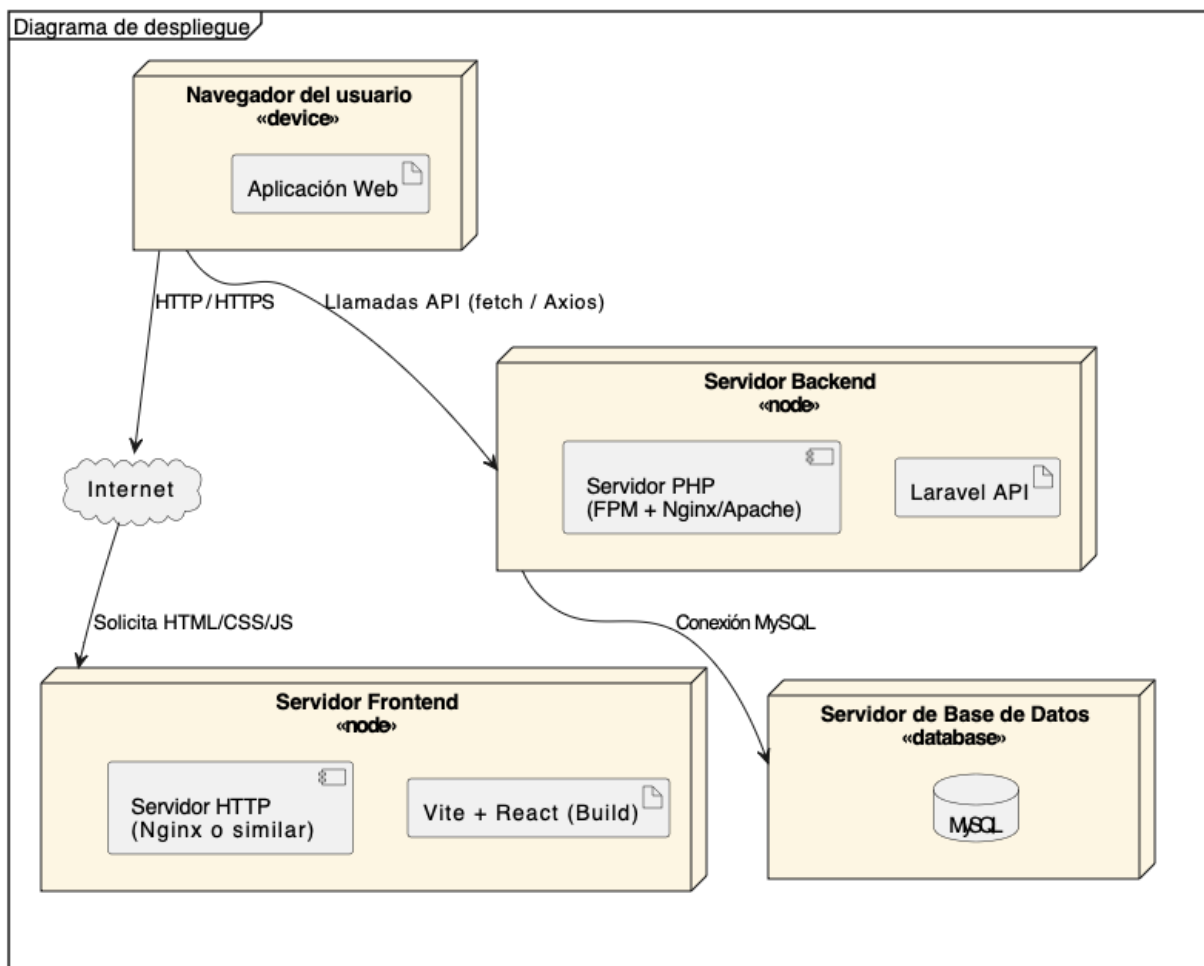
3.6.2. Diagrama de despliegue

El diagrama de despliegue representa la distribución física de los componentes descritos anteriormente sobre la infraestructura utilizada. Este modelo detalla la disposición de nodos físicos o virtuales, la ubicación del código y las conexiones entre los diferentes servidores.

El sistema se ejecuta desde el navegador del usuario, el cual accede a un servidor de frontend que entrega los recursos estáticos de la aplicación web. Las interacciones del usuario que requieren lógica de negocio o acceso a datos se canalizan hacia el servidor backend mediante llamadas HTTP, el cual a su vez establece la conexión con el servidor de base de datos MySQL.

Figura 21

Diagrama de despliegue del sistema



CAPITULO 4. ARQUITECTURA Y DESARROLLO DEL SISTEMA

4.1. Visión general de la arquitectura del sistema

El sistema desarrollado adopta una arquitectura cliente – servidor, en la cual el frontend y el backend se encuentran completamente desacoplados y se comunican a través de servicios web basados en una API REST. Esta decisión arquitectónica permite una mayor flexibilidad, escalabilidad y mantenibilidad, además de facilitar la evolución futura del sistema sin afectar a los demás componentes.

El backend es responsable de la lógica de negocio, la gestión de datos, la seguridad y la exposición de servicios, mientras que el frontend se encarga de la interacción con el usuario, la presentación de la información y el consumo de los servicios ofrecidos por la API. Esta separación de responsabilidades garantiza una clara división entre capas, alineándose con buenas prácticas de ingeniería de software y favoreciendo el trabajo colaborativo.

4.2. Arquitectura del backend

El backend del sistema fue desarrollado utilizando el framework Laravel, el cual sigue un enfoque basado en el patrón Modelo-Vista-Controlador (MVC). Este patrón permite organizar el código de manera estructurada, separando la representación de los datos, la lógica de negocio y el control de las peticiones.

Dentro de la arquitectura del backend se identifican las siguientes capas y componentes principales (Anexo N):

- **Controladores (Controllers):** encargados de recibir las solicitudes HTTP, validar el flujo general y delegar la lógica de negocio a los servicios correspondientes.
- **Modelos (Models):** representan las entidades del sistema y gestionan la interacción con la base de datos.
- **Servicios (Services):** concentran la lógica de negocio, permitiendo mantener controladores livianos y reutilizables.

- **Requests y Responses:** utilizados para la validación de datos de entrada y la estandarización de las respuestas del sistema.
- **Middleware:** implementa mecanismos de seguridad, autenticación y control de acceso según roles y permisos.
- **Enums y Traits:** facilitan la reutilización de código y la consistencia en valores constantes y comportamientos comunes.

Esta organización permite un backend modular, mantenible y preparado para escalar conforme se incorporen nuevas funcionalidades.

4.2.1. Base de datos y persistencia de la información

El sistema utiliza MySQL como sistema de gestión de base de datos relacional, seleccionado por su estabilidad, rendimiento, escalabilidad y amplia compatibilidad con entornos web. Esta base de datos es responsable de almacenar de forma estructurada toda la información relacionada con las incidencias EDAN, usuarios, roles, catálogos y registros históricos del sistema.

La persistencia de datos se gestiona a través del ORM Eloquent, propio de Laravel, el cual permite mapear las tablas de la base de datos a modelos de dominio del sistema. Este enfoque facilita la manipulación de la información mediante una abstracción orientada a objetos, reduciendo la complejidad del acceso a datos y mejorando la legibilidad del código.

La estructura de la base de datos se define y mantiene mediante el uso de migraciones, lo que permite versionar los cambios en el esquema, asegurar la consistencia entre entornos de desarrollo y producción, y facilitar la evolución del modelo de datos conforme se incorporan nuevas funcionalidades. Asimismo, se utilizan seeders y factories para la generación controlada de datos iniciales y de prueba, lo cual resulta fundamental durante las fases de desarrollo, pruebas y validación del sistema.

El diseño de la base de datos sigue principios de normalización, garantizando la integridad referencial y evitando la redundancia de información. Además, se

implementan relaciones entre entidades clave como usuarios, grupos, incidencias y formularios EDAN, lo que permite una gestión coherente y trazable de los datos.

Este enfoque asegura que la base de datos no solo funcione como un repositorio de información, sino como un componente activo del sistema, alineado con la lógica de negocio y preparado para soportar consultas, reportes y análisis posteriores de forma eficiente y segura.

4.2.2. Gestión de la lógica de negocio

La lógica de negocio del sistema se encuentra organizada en una **capa de servicios**, cuyo objetivo es centralizar las reglas y procesos que rigen el funcionamiento de la aplicación. Este enfoque permite desacoplar la lógica principal de los controladores, los cuales se limitan a gestionar las solicitudes entrantes y coordinar la respuesta adecuada.

La utilización de servicios favorece la reutilización de código, mejora la mantenibilidad del sistema y facilita la incorporación de nuevas funcionalidades sin afectar de manera directa a otros componentes. Asimismo, esta estructura permite un mayor control sobre los procesos internos y contribuye a un diseño más limpio y ordenado del backend.

4.2.3. Validación y manejo de datos

El sistema implementa mecanismos de validación a nivel de backend con el fin de garantizar la integridad, consistencia y confiabilidad de la información almacenada. Estos mecanismos permiten verificar que los datos recibidos cumplan con los formatos, tipos y rangos esperados antes de ser procesados o persistidos en la base de datos.

La validación se realiza de manera centralizada, reduciendo el riesgo de errores y evitando el ingreso de información incompleta o inconsistente. Además, el sistema maneja de forma controlada los errores derivados de datos inválidos, proporcionando respuestas claras al usuario final sin exponer detalles técnicos internos.

4.2.4. Gestión de seguridad y control de acceso

La arquitectura del backend contempla mecanismos de seguridad orientados a proteger la información y restringir el acceso a las funcionalidades del sistema según el perfil del usuario. Para ello, se emplea un esquema de autenticación basado en tokens y un modelo de autorización sustentado en roles y permisos.

Estos mecanismos permiten definir distintos niveles de acceso y asegurar que cada usuario solo pueda interactuar con los recursos que le han sido asignados. La aplicación de políticas de seguridad se realiza mediante componentes intermedios que controlan las solicitudes antes de su procesamiento, garantizando así un entorno controlado y alineado con los requerimientos de confidencialidad y protección de datos propios de un sistema gubernamental.

4.2.5. Manejo de errores y respuestas del sistema

El sistema incorpora un manejo estructurado de errores que permite identificar, registrar y responder adecuadamente ante situaciones inesperadas. Las respuestas generadas siguen convenciones estándar, lo que facilita su interpretación tanto por el frontend como por otros posibles consumidores de la API.

Este enfoque contribuye a una experiencia de usuario más clara y controlada, ya que los mensajes de error se presentan de forma comprensible, evitando la exposición de información técnica sensible. Internamente, el sistema cuenta con mecanismos de registro que apoyan las tareas de diagnóstico, mantenimiento y mejora continua.

4.2.6. Organización y estructura del proyecto

El proyecto backend se encuentra organizado siguiendo una estructura modular que agrupa los componentes del sistema según su responsabilidad. Esta organización facilita la comprensión del código, mejora la mantenibilidad y permite escalar el sistema de manera ordenada.

La separación en directorios específicos para modelos, servicios, validaciones, configuraciones y otros componentes contribuye a un diseño claro y coherente. Esta

estructura, alineada con las convenciones del framework utilizado, favorece el trabajo colaborativo y la incorporación de nuevos desarrolladores al proyecto, reduciendo la curva de aprendizaje y el riesgo de errores.

4.3. Documentación del backend

La documentación de la API fue implementada utilizando Swagger (OpenAPI), lo cual permite describir de forma clara y estructurada los endpoints disponibles, los métodos HTTP, los parámetros de entrada y las respuestas del sistema.

Esta documentación facilita:

- La comprensión del funcionamiento de la API.
- La realización de pruebas sin necesidad de acceder directamente al código.
- La integración futura con otros sistemas o aplicaciones.
- El mantenimiento y la evolución del backend por parte de otros desarrolladores.

El uso de Swagger contribuye significativamente a la calidad técnica del proyecto, al promover transparencia y estandarización en la comunicación entre componentes.

4.4. Arquitectura del frontend

El frontend fue desarrollado utilizando React con TypeScript, aplicando principios de Clean Architecture. Este enfoque arquitectónico promueve la separación clara entre las capas del sistema, asegurando que la lógica de negocio no dependa directamente de frameworks, librerías externas o detalles de infraestructura.

La estructura del frontend se organiza de la siguiente manera (Anexo N):

- **Dominio (domain):** contiene las reglas del negocio y los modelos principales.
- **Infraestructura (infrastructure / data):** maneja la comunicación con la API y el acceso a servicios externos.
- **Presentación (presentation / components / pages):** gestiona la interfaz de usuario y la experiencia visual.

- **Features:** agrupa los módulos funcionales del sistema, como EDAN 72, EDAN 120, usuarios, roles, catálogos y reportes.
- **State y hooks:** gestionan el estado global y la lógica reutilizable de la aplicación.

Esta arquitectura facilita la escalabilidad del frontend, mejora la legibilidad del código y reduce el acoplamiento entre componentes.

4.4.1. Enfoque arquitectónico general

El frontend del sistema fue desarrollado utilizando React con TypeScript, adoptando principios de Clean Architecture con el objetivo de lograr una aplicación modular, escalable y de fácil mantenimiento. Este enfoque permite separar claramente las responsabilidades del sistema, evitando dependencias directas entre la lógica de negocio y los detalles de implementación, tales como la interfaz de usuario o la comunicación con servicios externos.

La arquitectura del frontend se organiza en capas bien definidas, lo que facilita la evolución del sistema, el trabajo colaborativo y la incorporación de nuevos módulos funcionales sin afectar el comportamiento global de la aplicación.

4.4.2. Capa central y recursos comunes (core)

La carpeta core concentra los elementos transversales utilizados por toda la aplicación. En esta capa se incluyen recursos compartidos como:

- Assets, que contienen recursos gráficos y estáticos.
- Componentes reutilizables, empleados en múltiples vistas del sistema.
- Configuraciones generales, utilizadas para estandarizar comportamientos comunes.
- Utilidades y hooks genéricos, que encapsulan lógica reutilizable.
- Gestión de estado global, cuando aplica.

Esta organización evita la duplicación de código y promueve la reutilización de componentes, mejorando la coherencia visual y funcional del sistema.

4.4.3. Organización por funcionalidades (features)

El sistema adopta un enfoque modular basado en features, donde cada módulo funcional agrupa todos los elementos necesarios para una funcionalidad específica del sistema, tales como autenticación, gestión de incidencias, formularios EDAN 72 y EDAN 120, reportes, usuarios, roles y catálogos.

Cada feature funciona como un módulo independiente, lo que permite aislar cambios, mejorar la escalabilidad y facilitar el mantenimiento. Esta organización favorece el desarrollo incremental y el trabajo en paralelo por parte del equipo.

4.4.4. Separación interna de capas dentro de cada feature

Dentro de cada feature se implementa una separación clara de capas, alineada con los principios de Clean Architecture:

- Capa de dominio (domain):

Contiene las entidades, casos de uso y contratos que representan las reglas del negocio. Esta capa es independiente de frameworks y tecnologías externas, lo que garantiza que la lógica central del sistema permanezca estable ante cambios de infraestructura.

- Capa de datos (data):

Incluye las fuentes de datos, modelos y repositorios encargados de interactuar con servicios externos, como la API del backend. Esta capa actúa como intermediaria entre el dominio y la infraestructura.

- Capa de presentación (presentation):

Gestiona los componentes visuales, páginas y lógica relacionada con la interacción del usuario. Aquí se implementa la interfaz gráfica y se conectan los casos de uso con la experiencia de usuario.

Esta separación refuerza el desacoplamiento entre capas y mejora la claridad del diseño.

4.4.5. Inyección de dependencias y control de acoplamiento

El frontend incorpora mecanismos de inyección de dependencias, lo que permite desacoplar la creación de objetos de su uso. Este enfoque facilita la sustitución de implementaciones concretas, mejora la testabilidad y refuerza la independencia entre capas.

Gracias a esta estrategia, los módulos de dominio no dependen directamente de detalles de infraestructura, lo que contribuye a un diseño más flexible y mantenible.

4.4.6. Comunicación con el backend e integración de servicios

La comunicación con el backend se realiza mediante clientes HTTP centralizados, lo que permite estandarizar el manejo de solicitudes, respuestas y errores. Esta integración se encuentra encapsulada dentro de la capa de infraestructura, evitando que los componentes de presentación conozcan detalles específicos de la comunicación externa.

Este diseño asegura una interacción controlada y consistente entre frontend y backend, facilitando futuras modificaciones en los mecanismos de comunicación sin afectar la lógica de negocio ni la interfaz de usuario.

4.4.7. Gestión del estado y flujo de información

El sistema implementa mecanismos de gestión de estado que permiten mantener sincronizada la información entre distintas vistas y componentes. Esta gestión se apoya en herramientas modernas y en el uso de **custom hooks**, que encapsulan lógica reutilizable y reducen la complejidad de los componentes visuales.

El control del estado favorece una experiencia de usuario fluida y predecible, especialmente en flujos complejos como el llenado de formularios EDAN y la generación de reportes.

4.4.8. Interfaz de usuario y experiencia de uso

La interfaz del sistema fue desarrollada utilizando herramientas modernas de diseño basadas en utilidades CSS, lo que permitió construir una experiencia visual consistente, adaptable y responsiva. Este enfoque garantiza que la aplicación pueda ser utilizada tanto en equipos de escritorio como en dispositivos móviles, aspecto fundamental para los usuarios que operan en campo.

La reutilización de componentes visuales y estilos contribuye a una experiencia uniforme y facilita futuras mejoras en la presentación del sistema.

4.4.9. Herramientas de construcción y calidad del código

Para el desarrollo del frontend se emplearon herramientas modernas orientadas a la calidad y eficiencia del proceso de desarrollo, tales como:

- Un entorno de construcción y desarrollo rápido.
- Herramientas de análisis estático y formateo automático del código.
- Tipado estático mediante TypeScript, que reduce errores y mejora la mantenibilidad.

Estas herramientas fortalecen el proceso de desarrollo y aseguran un código más confiable y consistente a lo largo del tiempo.

4.5. Herramientas y entorno de desarrollo

El desarrollo del sistema se apoyó en herramientas modernas como:

- Git para control de versiones.
- Prettier y ESLint para calidad y consistencia del código.
- Variables de entorno para configuración segura de ambientes.
- PHPUnit y Faker para pruebas y generación de datos.

Estas herramientas garantizan un proceso de desarrollo ordenado, reproducible y alineado con estándares profesionales.

VII. CONCLUSIONES

- El análisis de los requerimientos técnicos y funcionales permitió identificar de manera precisa las necesidades operativas del CNR y las condiciones reales de trabajo de los agentes de campo, estableciendo una base sólida para la automatización de los formularios EDAN 72 y EDAN 120 y para la definición de un sistema web acorde al contexto institucional.
- El diseño de los modelos conceptuales, estructurales y navegacionales mediante la metodología UWE proporcionó una visión integral del sistema, asegurando coherencia entre los procesos de negocio, la estructura de la información y la interacción con el usuario, lo que facilitó un diseño ordenado y alineado con los objetivos del proyecto.
- El desarrollo de la plataforma web demostró la viabilidad de implementar una solución adaptable a distintos dispositivos, capaz de gestionar incidencias, recolectar información en campo y generar reportes automatizados, contribuyendo a una mejora significativa en la disponibilidad y uso de la información para la toma de decisiones institucionales.
- La arquitectura tecnológica basada en Laravel, React y MySQL resultó adecuada para las capacidades actuales del CNR, ofreciendo una solución robusta, modular y escalable, preparada para futuras ampliaciones o integraciones del sistema.
- La documentación técnica generada, apoyada en modelos UWE, diagramas de base de datos y documentación de la API, constituye un soporte fundamental para el mantenimiento, comprensión y evolución futura del sistema.
- El análisis de factibilidad técnica confirmó que la infraestructura tecnológica del CNR cumple con los requerimientos mínimos y la mayoría de los recomendados, permitiendo el despliegue del sistema sin requerir inversiones inmediatas en equipamiento adicional.
- Las estimaciones realizadas mediante Puntos de Función y el modelo COCOMO evidenciaron que el proyecto es realizable en términos de esfuerzo, tiempo y recursos humanos, proporcionando una proyección clara para su planificación e implementación.

VIII. RECOMENDACIONES

1. Implementación gradual y capacitación del personal: Se recomienda implementar el sistema en fases, comenzando por pilotos controlados en algunos recintos universitarios. Paralelamente, se debe desarrollar un programa de formación para los usuarios finales, enfocado en agentes de campo y personal administrativo, con el fin de asegurar una correcta adopción del sistema.
2. Evolución continua del sistema: Una vez desplegado, es recomendable realizar mantenimientos periódicos y recoger retroalimentación de los usuarios para identificar mejoras. Esto incluye la incorporación de nuevos módulos o funcionalidades, como alertas tempranas, integración con bases de datos nacionales o capacidades offline.
3. Integración con instituciones afines: Considerando la naturaleza de la gestión de riesgos, se recomienda establecer vínculos de interoperabilidad con otras instituciones clave como SINAPRED, permitiendo compartir información de forma segura y estandarizada durante emergencias.
4. Fortalecimiento de la seguridad de la información: A medida que el sistema evolucione, se sugiere realizar auditorías periódicas y adoptar políticas de seguridad avanzadas que garanticen la integridad, confidencialidad y disponibilidad de la información crítica recopilada.
5. Evaluación del impacto institucional: Finalmente, es recomendable que el CNR establezca indicadores de evaluación para medir el impacto del sistema en los procesos de respuesta ante desastres. Esta medición permitirá justificar futuras inversiones y mejorar continuamente las capacidades institucionales ante eventos adversos.
6. Elaborar manuales complementarios, incluyendo un manual de usuario con instrucciones paso a paso sobre la operación del sistema y un manual de instalación/configuración para el despliegue y mantenimiento en servidores. Estos documentos, junto con la documentación técnica existente, garantizarán una transición fluida hacia la operación productiva del sistema.

IX. Bibliografía

- Baena Paz, G. M. (2017). *Metodología de la investigación*. México: Grupo Editorial Patria.
- Busch, M. (2016). *Evaluating & Engineering: an Approach for the Development of Secure Web Applications*. Ludwig Maximilian University of Munich.
- Connolly, T., & Begg, C. (2014). *Database Systems: A Practical Approach to Design, Implementation, and Management*. Pearson.
- Flanagan, D. (2020). *Javascript: The Definitive Guide*. O'Reilly Media, Inc.
- Meyer, E., & Weyl, E. (2023). *CSS: The Definitive Guide*. O'Reilly Media. O'Reilly Media, Inc.
- Molina, J., & Pedreira, M. (2019). *SWIRL, metodología para el diseño y desarrollo de aplicaciones web* (Edición 1ª ed.).
- Morales, R. (2016). *Fundamentos de BPMN*. Ediciones Rainer, S.L.
- Panda, B., Sharma, S., & Roy, N. R. (2018). *Data Science and Analytics*. Singapore: Springer Singapore.
- Reyes Hernández, F., Hernández Cárdenas, L., Moreno Montes de Oca, I., Rodríguez Morffi, A., & Moreno Rodríguez, R. (2019). *Aplicación de directrices de calidad a modelos de procesos de negocio a través de archivos de salida con extensión BPMN y XPD L*. Editorial Feijóo.
- Saradhi, V., & Jena, S. (2022). *Software Requirements and Estimation: A Systematic Approach*. Lambert Academic Publishing.
- Sawyer McFarland, D. (2014). *JavaScript & jQuery: The Missing Manual*. O'Reilly Media, Inc.

X. ANEXOS

Anexo A. Gastos de capacitación

Para garantizar que la estimación de costos por personal para la creación de videos sea lo más cercana a la realidad, se evaluaron tres propuestas laborales de diferentes empresas dispuestas a contratar el recurso humano requerido. En los siguientes anexos se presentan dichas propuestas, especificando los montos ofertados, condiciones y otros aspectos relevantes para el análisis.

Figura 22

Oferta ofrecida en el mercado laboral



Figura 23

Oferta ofrecida en el mercado laboral



Figura 24

Oferta ofrecida en el mercado laboral

[← Regresar a la búsqueda](#)[Compartir esta oferta](#)



ACEITERA EL REAL
<http://www.tecoloco.com>

Trabajos en área Publicidad | Comunicaciones | Servicios

Trabajos en rubro Manufacturas Varias

[Ver más empleos de esta empresa →](#)

[ENVÍAME OFERTAS COMO ÉSTAS ▼](#)

PASANTE PARA MANAGUA (ESTUDIANTE DE DISEÑO GRAFICO)

ACEITERA EL REAL

⌚ Expira 27/08/2025

📍 Managua, Nicaragua

[☆ AGREGAR A FAVORITOS](#) [APLICAR](#)

DETALLE DE LA OFERTA

Área de la Empresa	Manufacturas Varias
Cargo Solicitado	Diseñador Gráfico
Puestos Vacantes	1
Tipo de Contratación	Pasantía
Nivel de Experiencia	Sin experiencia
Salario máximo (USD)	110
Salario mínimo (USD)	100
Vehículo	Indiferente
País	Nicaragua
Departamento	Managua

DESCRIPCIÓN DE LA OFERTA

Asimismo, se efectuó un análisis del costo correspondiente al recurso faltante para la ejecución del trabajo de capacitación, consistente en una computadora portátil que será asignada al profesional encargado de dicha labor.

Figura 25

Oferta de computadora portátil

SEVASA
tecnología a tu alcance

Buscar

INICIO CATALOGO TIENDA OFERTAS CONTÁCTANOS POLÍTICAS EMPLEOS

COMPUTADORAS LAPTOP CORE I3

LAPTOP DELL 3520 SPA CI3-1215U 8GB 512SSD 15.
Código: 4174
Marca: DELL
Imágenes con fines ilustrativos
€918,275.53
C\$16,297.82 +IVA
En existencia

Cantidad: 1 **Agregar al carrito**

Descripción
COMPUTADORA LAPTOP DELL INSPIRON 15 3520 Procesador Intel Corei i3-1215U has (1/2) Almacenamiento SSD 512GB Tres puertos USB HDMI WIFI Lector de tarjetas Cárr SL

Compartir

TU TRANSACCIÓN ES SEGURA



Figura 26

Oferta de computadora portátil

LAPTOPS

LAPTOP LENOVO IDEAPAD 3 15IAU7, PANTALLA: 15.6" TOUCH, PROCESADOR: INTEL CORE I3-1215U, MEMORIA: 8GB, ALMACENAMIENTO: 256GB SSD, SISTEMA OPERATIVO: WINDOWS 11, IDIOMA: INGLÉS, COLOR: GRIS

Disponible: 1 disponibles

C\$15,388.86 + IVA

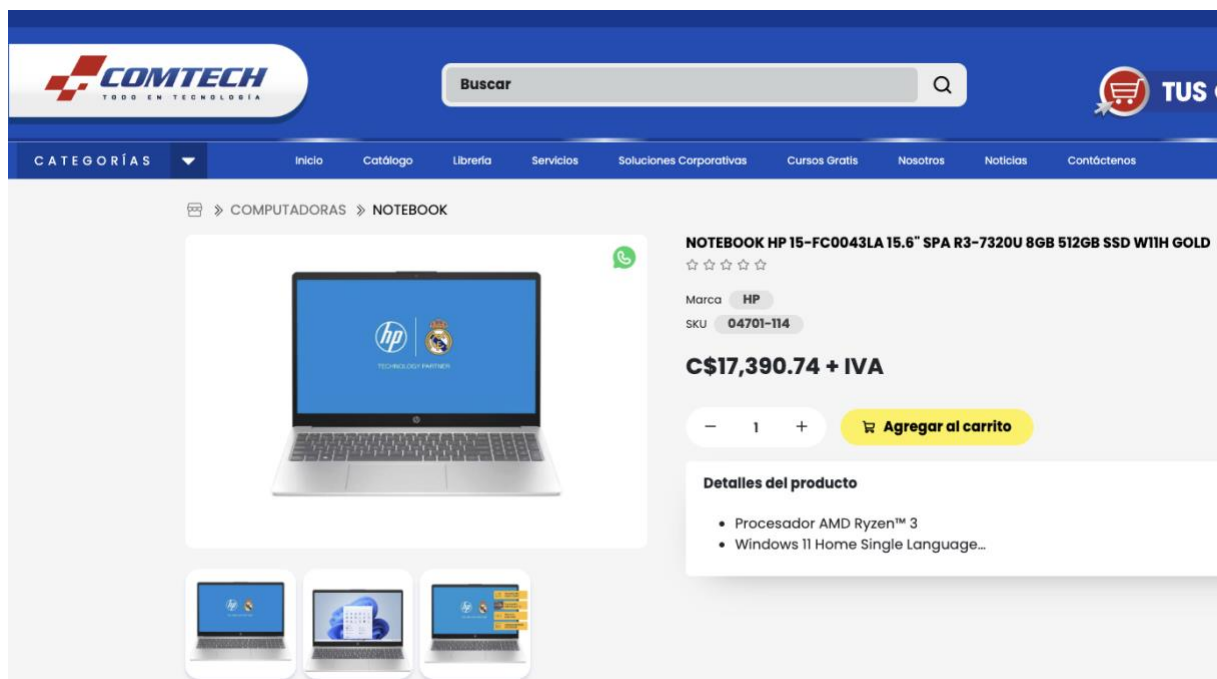
Añadir al carrito



SICSA

Figura 27

Oferta de computadora portátil



Anexo B. Pesos de complejidad de los componentes de los puntos de función

Figura 28

Pesos de complejidad de los componentes de los puntos de función (Panda, Sharma, & Roy, 2018)

Function type	Complexity		
	Low	Average	High
External input	3	4	6
External output	4	5	7
External inquiry	3	4	6
Internal logical file	7	10	15
External interface file	5	7	10

Anexo C. Multiplicador de lenguaje de programación

La conversión de Puntos de Función Ajustados a líneas de código se realizó utilizando el multiplicador del modelo COCOMO II para lenguajes de cuarta generación. Este factor se seleccionó por la alta capacidad de abstracción y productividad del stack tecnológico elegido: JavaScript (React) y PHP (Laravel).

Tabla 46

Multiplicador de lenguaje de programación

Lenguaje de programación	
Ensamblador	320
C	128
Cobol	105
Fortran	105
Pascal	90
ADA	70
Lenguajes orientados a objetos	30
Lenguajes de cuarta generación	20
Generadores de código	15
Hojas de cálculo	6
Íconos	4

Anexo D. Valores de referencia Factores de escala y Factores del producto

Figura 29

Valores de referencia Factores de escala (Panda, Sharma, & Roy, 2018)

Scale factors (<i>Wi</i>)	Extra high	Very high	High	Nominal	Low	Very low
PREC	Completely familiar	Very familiar	Quite familiar	Quite unprecedented	Very unprecedented	Never developed before
SFj:	0	1.24	2.48	3.72	4.96	6.2
FLEX	General goals	Some conformity	General conformity	Some relaxation	Occasional relaxation	Rigorous
SFj:	0	1.01	2.03	3.04	4.05	5.07
RESL	Full (100%)	Mostly (90%)	Generally (75%)	Often (60%)	Some (40%)	Little (20%)
SFj:	0	1.41	2.83	4.24	5.65	7.07
TEAM	Seamless interactions	Very cooperative	Quite	Generally cooperative interactions	Difficulty in interactions	Very difficult interactions
SFj:	0	1.1	2.19	3.29	4.38	5.48
PMAT	CMM Level 5	CMM Level 4	CMM Level 3	CMM Level 2	CMM Level 1 (Upper half)	CMM Level 1 (Lower half)
SFj:	0	1.56	3.12	4.68	6.24	7.8

Figura 30

Valores de referencia Factores de producto (Panda, Sharma, & Roy, 2018)

Post-architecture cost drivers		XH	VH	H	N	L	VL
Product factors	Required Reliability (RELY)	n/a	1.26	1.10	1	0.92	0.82
	Database Size (DATA)	n/a	1.28	1.14	1	0.90	n/a
	Product Complexity (CPLX)	1.74	1.34	1.17	1	0.87	0.73
	Developed for Reusability (RUSE)	1.24	1.15	1.07	1	0.95	n/a
	Documentation Match to Life-Cycle Needs (DOCU)	n/a	1.23	1.11	1	0.91	0.81
Platform factors	Execution Time Constraints (TIME)	1.63	1.29	1.11	1	n/a	n/a
	Main Storage Constraint (STOR)	1.46	1.17	1.05	1	n/a	n/a
	Platform Volatility (PVOL)	n/a	1.30	1.15	1	0.87	n/a
Personnel factors	Analyst Capability (ACAP)	n/a	0.71	0.85	1	1.19	1.42
	Programmer Capability (PCAP)	n/a	0.76	0.88	1	1.15	1.34
	Personnel Continuity (PCON)	n/a	0.81	0.90	1	1.12	1.29
	Applications Experience (APEX)	n/a	0.81	0.88	1	1.10	1.22
	Platform Experience (PLEX)	n/a	0.85	0.91	1	1.09	1.19
	Language and tool Experience (LTEX)	n/a	0.84	0.91	1	1.09	1.20
Project factors	Use of Software Tools (TOOL)	n/a	0.78	0.90	1	1.09	1.17
	Multi-site Development (SITE)	0.80	0.86	0.93	1	1.09	1.22
	Required Development Schedule (SCED)	n/a	1.00	1.00	1	1.14	1.43

Anexo E. Factores de escala

Durante el análisis de factibilidad económica y la aplicación del modelo COCOMO II, se asignaron valores a los cinco factores de escala del modelo, con base en una evaluación conjunta del equipo de trabajo. A continuación, se detalla la justificación correspondiente para cada uno de ellos:

- PREC (Precedencia del proyecto)

Nivel: Nominal

El equipo ha trabajado anteriormente en proyectos similares, lo que brinda cierta experiencia. Sin embargo, no se cuenta con cronogramas de ejecución formalizados ni con un liderazgo técnico definido, lo cual limita el grado de madurez en la gestión del proyecto.

- FLEX (Flexibilidad en el desarrollo)

Nivel: Alto

Aunque los requerimientos funcionales fueron establecidos por el consultor del proyecto, se nos otorgó libertad total para tomar decisiones técnicas relacionadas con arquitectura, herramientas y lenguajes, permitiendo una implementación adaptativa y orientada al contexto del equipo.

- RESL (Resolución de arquitectura y riesgos)

Nivel: Nominal

Se definió una arquitectura base para el sistema, la cual orienta la organización de los componentes principales. Sin embargo, no se realizaron pruebas de validación temprana ni un análisis formal de mitigación de riesgos técnicos.

- TEAM (Cohesión del equipo)

Nivel: Muy alto

El equipo de desarrollo mantiene una comunicación fluida y una colaboración activa. Existe una cultura de trabajo sólida, con un alto nivel de entendimiento

mutuo, lo cual ha favorecido una dinámica eficiente y sin fricciones durante el avance del proyecto.

- PMAT (Madurez del proceso)

Nivel: Bajo

A pesar de contar con una planificación básica y una estructura de desarrollo organizada, no se han implementado procesos estandarizados ni métricas de seguimiento formal. La documentación se mantiene funcional, pero no alcanza un nivel de madurez avanzado.

Tabla 47

Factores de escala calculados

Factores de escala		
Indicador	Nivel	Valor
PREC	Nominal	3.72
FLEX	Alto	2.03
RESL	Nominal	4.24
TEAM	Muy alto	1.1
PMAT	Bajo	6.24
ΣSFi		17.33

Anexo F. Factores del producto

Para el cálculo del producto de multiplicadores de esfuerzo (πEM_i) dentro del modelo COCOMO II, se evaluaron los cinco indicadores asociados al producto: fiabilidad, tamaño de la base de datos, complejidad del producto, reusabilidad y documentación. A continuación, se detalla la justificación para cada uno, según el análisis realizado por el equipo de desarrollo:

- RELY (Fiabilidad del software)

Nivel: Nominal

El sistema requiere mantener una alta disponibilidad, especialmente en contextos de emergencia, sin embargo, no se considera de misión crítica en cuanto a fallas que puedan comprometer decisiones inmediatas. Por tanto, se clasificó con un nivel nominal de fiabilidad.

- DATA (Tamaño de la base de datos)

Nivel: Alto

El sistema gestionará formularios extensos (EDAN 72 y EDAN 120), los cuales contienen múltiples secciones, subregistros y relaciones. Esto implica una base de datos robusta, con varias entidades y alto volumen de información estructurada.

- CPLX (Complejidad del producto)

Nivel: Nominal

El sistema presenta una complejidad funcional estándar, con múltiples formularios, reglas de validación, y lógica de presentación. No obstante, no incluye cálculos matemáticos avanzados ni procesamiento intensivo que lo clasifiquen como altamente complejo.

- RUSE (Reusabilidad del software)

Nivel: Nominal

Aunque se diseñaron componentes reutilizables para la interfaz de usuario (como campos, secciones o tablas), el sistema no fue construido con la reusabilidad como un objetivo central, por lo que se asignó un nivel nominal en esta dimensión.

- DOCU (Necesidad de documentación)

Nivel: Nominal

Se prevé una documentación técnica adecuada para instalación, mantenimiento y uso del sistema. Esta documentación sigue un enfoque estándar, suficiente para futuras mejoras o soporte, pero sin requerimientos especiales ni regulaciones estrictas.

Tabla 48

Indicadores del producto calculados

Indicadores del producto		
Indicador	Nivel	Valor
RELY	Nominal	1
DATA	Alto	1.14
CPLX	Nominal	1
RUSE	Nominal	1
DOCU	Bajo	0.91
Σ		5.05

Anexo G. Estimación del ahorro y gasto de software de escala

Como parte del modelo COCOMO II, se estimó el exponente de escala que influye directamente en la magnitud del esfuerzo requerido. Este valor se obtuvo con base en la suma de los factores de escala previamente justificados (PREC, FLEX, RESL, TEAM, PMAT).

Este parámetro permite reflejar el comportamiento no lineal del desarrollo de software, considerando tanto economías como diseconomías de escala, y adaptándose al contexto particular del equipo y del proyecto.

Tabla 49

Estimación del ahorro y gasto de software de escala

Estimación del ahorro y gasto de software de escala	
Fórmula	$B = 0.91 + (0.01 * \sum SF_i)$
Valor	1.0833

Anexo H. Cálculo de πEM_i

El valor de πEM_i corresponde al producto de los multiplicadores de esfuerzo definidos por COCOMO II. En este caso, se enfatizó la evaluación de los indicadores del producto: fiabilidad (RELY), volumen de datos (DATA), complejidad lógica (CPLX), reusabilidad (RUSE) y documentación (DOCU), los cuales fueron valorados de acuerdo con las características específicas del sistema.

Cada uno de estos indicadores fue asociado a un nivel cualitativo, el cual se tradujo a su valor correspondiente según las tablas oficiales del modelo. El producto de estos valores conformó el resultado de πEM_i , que fue utilizado en el cálculo final del esfuerzo estimado.

Tabla 50

Cálculo de πEM_i

Cálculo de πEM_i	
Fórmula	$\pi EM_i = \sum \text{Indicadores producto} / 17$
Valor	0.297058824

Anexo I. Costos de consumo de energía

Para estimar el costo de energía asociado a cada etapa del proyecto, se utilizó como base el consumo promedio de una computadora durante la jornada laboral, el costo promedio del kilovatio-hora (kW/h) en córdobas nicaragüenses y el número de horas promedio de trabajo al mes. Este cálculo permite identificar el gasto operativo relacionado con el uso de equipos durante el desarrollo del sistema.

Tabla 51

Consumo de energía por hora de una PC promedio (Ce)

Consumo de energía por hora de una PC promedio (kW/h) (Ce)	
0.07312 kWh	73.12 Wh

Tabla 52

Costo de energía promedio (CKH)

Costo de energía promedio C\$ kw/h (CKH)
11.1827

Tabla 53

Tiempo de trabajo al mes (NoH)

Tiempo de trabajo al mes (Horas/Mes) (NoH)	
Horas	8
Días	24
Total	192

Luego, se determinó el Costo de Consumo de Energía (Cce), el cual representa el gasto mensual por equipo en función de su consumo promedio, la tarifa eléctrica y las horas de uso mensual. Para ello, se empleó:

Tabla 54

Costo de consumo de energía (Cce)

Costo de consumo de energía (Cce) (Total de consumo en C\$ (kW/h))	
Formula	$Cce = Ce * CKH * NoH$
Valor	C\$ 156.99 / computadora - mes

Se obtiene multiplicando el valor unitario (Cce) por la duración en meses y la cantidad de equipos de cada fase. Según la tabla 3.4.3.2, este modelo identifica a la etapa de Diseño y Desarrollo como la de mayor consumo, debido a que es la fase más extensa del proyecto.

Anexo J. Plantillas de Coleman

Tabla 55

Gestionar incidencias (fuente: elaboración propia)

Caso de uso (1):	Gestionar incidencias		
Definición:	Ingresar y manipular todos los datos de las incidencias		
Prioridad:	☐ (1) Vital	☒ (2) Importante	☐ (3) Conveniente
Urgencia:	☐ (1) Inmediata	☒ (2) Necesario	☐ (3) Puede esperar
Actores			
Nombre:	Administrador		
Escenario 1			
Nombre:	Crear incidencia		
Pre-Condiciones:			
Iniciado por:	Administrador		
Finalizado por:	Usuario		
Post-Condiciones:			
Operaciones:	1. Clic en botón "Crear incidencia" 2. Se despliega modal con solicitud de datos iniciales 3. Se crea la incidencia		
Excepciones:	Si los datos están mal ingresados, estos no se guardarán en la base de datos		
Escenario 2			
Nombre:	Editar incidencia		
Pre-Condiciones:			
Iniciado por:	Usuario		
Finalizado por:	Usuario		
Post-Condiciones:			
Operaciones:	1. Se activa la interfaz para gestionar datos de la incidencia 2. Se realiza una consulta para cargar datos ya existentes de la incidencia. 3. Se cargan los datos de la incidencia a la cual se desea modificar los datos. 4. Se verifica que todos los campos obligatorios del formulario estén llenos correctamente. 5. Se guardan los datos pertinentes en la base de datos.		
Excepciones:	Si los datos están mal ingresados, estos no se guardarán en la base de datos.		
Escenario 3			
Nombre:	Eliminar incidencia		
Pre-Condiciones:			
Iniciado por:	Administrador		
Finalizado por:	Administrador		
Post-Condiciones:			
Operaciones:	1. Se activa la interfaz para gestionar incidencias existentes. 2. Se realiza una consulta para listar las incidencias. 3. Se activa el proceso para eliminar una incidencia. 4. Muestra una ventana de confirmación. 5. Se elimina la incidencia.		
Excepciones:			

Tabla 56

Gestionar catálogos (fuente: elaboración propia)

Caso de uso (2):	Agregar universidades		
Definición:	Ingresar y manipular todos los datos con respecto a las universidades registradas.		
Prioridad:	☐ (1) Vital	☒ (2) Importante	☐ (3) Conveniente
Urgencia:	☐ (1) Inmediata	☐ (2) Necesario	☒ (3) Puede esperar
Actores			
Nombre:	Administrador		
Escenario 1			
Nombre:	Buscar universidad		
Pre-Condiciones:	Determinar si la universidad ya se encuentra registrada en el sistema.		
Iniciado por:	Administrador		
Finalizado por:	Administrador		
Post-Condiciones:	Determinar si la universidad ya se encuentra registrada en el sistema.		
Operaciones:	1. Se activa la interfaz para la gestión de universidades. 2. Se ingresa el nombre o parte del nombre de la universidad a consultar. 3. Se realiza una búsqueda en la base de datos. 4. Se muestra una lista con las coincidencias encontradas. 5. Se verifica si la universidad ya se encuentra registrada.		
Excepciones:	1. Si no se ingresan datos para la búsqueda, el sistema solicitará completar el campo. 2. Si no se encuentran coincidencias, se muestra un mensaje informativo.		
Escenario 2			
Nombre:	Registrar nueva universidad		
Pre-Condiciones:	El usuario desea registrar una nueva universidad en el sistema.		
Iniciado por:	Administrador		
Finalizado por:	Administrador		
Post-Condiciones:	La universidad queda registrada correctamente en el sistema.		
Operaciones:	2. Se activa la interfaz para capturar datos de la universidad. 3. Se verifica que todos los campos obligatorios estén llenos correctamente. 6. Se guardan los datos en la base de datos.		
Excepciones:	Si los datos están mal ingresados, estos no se guardarán en la base de datos.		
Escenario 3			
Nombre:	Modificar datos de la universidad		
Pre-Condiciones:	Debe existir una universidad registrada con datos modificables.		
Iniciado por:	Administrador		
Finalizado por:	Administrador		
Post-Condiciones:	La información de la universidad queda actualizada.		
Operaciones:	1. Se activa la interfaz para búsqueda de universidades. 2. Se consulta la base de datos para listar universidades. 3. Se selecciona la universidad a modificar. 4. Se editan los datos y se verifica que estén completos. 5. Se guardan los datos actualizados.		
Excepciones:	Si los datos están mal ingresados, estos no se guardarán en la base de datos.		
Escenario 4			
Nombre:	Cambiar estado de la universidad		

Pre-Condiciones:	1. La universidad haya cerrado. 2. Cambio administrativo.
Iniciado por:	Administrador
Finalizado por:	Administrador
Post-Condiciones:	La universidad cambia de estado correctamente
Operaciones:	1. Se accede a la interfaz para gestionar estados. 2. Se listan las universidades registradas. 3. Se selecciona la universidad y el nuevo estado. 4. Se muestra ventana de confirmación. 5. Se aplica el cambio de estado.
Excepciones:	Si ocurre un error en el proceso, no se cambia el estado.

Tabla 57

Gestionar usuarios (fuente: elaboración propia)

Caso de uso (2):	Gestionar incidencias		
Definición:	Modulo encargado de realizar operaciones CRUD con los datos de usuarios permitidos para manipular el sistema.		
Prioridad:	☐ (1) Vital	☒ (2) Importante	☐ (3) Conveniente
Urgencia:	☐ (1) Inmediata	☒ (2) Necesario	☐ (3) Puede esperar
Actores			
Nombre:	Administrador		
Escenario 1			
Nombre:	Visualizar usuarios		
Pre-Condiciones:	1. Usuario autenticado.		
Iniciado por:	Administrador		
Finalizado por:	Administrador		
Post-Condiciones:			
Operaciones:	4. Al ingresar al módulo. 5. Se visualiza la lista de usuarios.		
Excepciones:			
Escenario 2			
Nombre:	Crear registro de usuario		
Pre-Condiciones:	1. Usuario autenticado.		
Iniciado por:	Administrador		
Finalizado por:	Administrador		
Post-Condiciones:	1. Debe rellenar todos los datos necesarios para crear el usuario		
Operaciones:	6. Se muestra un modal con formulario de información propia del usuario. 7. Se validan todos los campos requeridos. 8. Se presiona el botón "Crear". 9. Se realizan las operaciones necesarias para crear el registro de usuario.		
Excepciones:	Si los datos están mal ingresados, no debe permitir crear usuario y debe indicar al usuario exactamente donde ocurre la verificación fallida.		
Escenario 3			
Nombre:	Modificar información de usuario		
Pre-Condiciones:	1. Usuario autenticado. 2. Seleccionar usuario a modificar.		
Iniciado por:	Administrador		
Finalizado por:	Administrador		
Post-Condiciones:			
Operaciones:	1. Se selecciona el usuario a modificar. 2. Se presiona botón "Actualizar". 3. Se modifican los campos del formulario. 4. Se verifica el llenado de información requerida. 5. Se realizan las operaciones necesarias para actualizar la información del registro.		
Excepciones:	Si los datos están mal ingresados, no debe permitir crear usuario y debe indicar al usuario exactamente donde ocurre la verificación fallida.		
Escenario 4			
Nombre:	Dar baja a usuario		

Pre-Condiciones:	1. Usuario autenticado 2. Seleccionar usuario a modificar
Iniciado por:	Administrador
Finalizado por:	Administrador
Post-Condiciones:	
Operaciones:	1. Se selecciona el usuario a dar baja. 2. Se presiona botón "Dar baja". 3. Se realizan las operaciones necesarias para cambiar el estado del usuario a desactivado.
Excepciones:	No se puede dar baja a usuario super-administrador

Tabla 58

Gestionar roles (fuente: elaboración propia)

Caso de uso (2):	Gestionar roles de usuario		
Definición:	Modulo encargado de realizar operaciones CRUD con los datos de roles para control de accesos al sistema.		
Prioridad:	☐ (1) Vital	☒ (2) Importante	☐ (3) Conveniente
Urgencia:	☐ (1) Inmediata	☒ (2) Necesario	☐ (3) Puede esperar
Actores			
Nombre:	Administrador		
Escenario 1			
Nombre:	Visualizar rol		
Pre-Condiciones:	2. Usuario autenticado.		
Iniciado por:	Administrador		
Finalizado por:	Administrador		
Post-Condiciones:			
Operaciones:	6. Al ingresar al módulo. 7. Se visualiza la lista de roles.		
Excepciones:			
Escenario 2			
Nombre:	Crear registro de rol		
Pre-Condiciones:	2. Usuario autenticado.		
Iniciado por:	Administrador		
Finalizado por:	Administrador		
Post-Condiciones:	2. Debe rellenar todos los datos necesarios para crear el rol		
Operaciones:	10. Se muestra un modal con formulario de información propia del rol. 11. Se validan todos los campos requeridos. 12. Se presiona el botón "Crear". 13. Se realizan las operaciones necesarias para crear el registro de rol.		
Excepciones:	Si los datos están mal ingresados, no debe permitir crear rol y debe indicar al usuario exactamente donde ocurre la verificación fallida.		
Escenario 3			
Nombre:	Modificar información de rol		
Pre-Condiciones:	3. Usuario autenticado. 4. Seleccionar usuario a modificar.		
Iniciado por:	Administrador		
Finalizado por:	Administrador		
Post-Condiciones:			
Operaciones:	6. Se selecciona el rol a modificar. 7. Se presiona botón "Actualizar". 8. Se modifican los campos del formulario. 9. Se verifica el llenado de información requerida. 10. Se realizan las operaciones necesarias para actualizar la información del registro.		
Excepciones:	Si los datos están mal ingresados, no debe permitir modificar los datos y debe indicar al usuario exactamente donde ocurre la verificación fallida.		
Escenario 4			
Nombre:	Dar baja a rol		

Pre-Condiciones:	3. Usuario autenticado 4. Seleccionar rol a modificar
Iniciado por:	Administrador
Finalizado por:	Administrador
Post-Condiciones:	
Operaciones:	4. Se selecciona el usuario a dar baja. 5. Se presiona botón "Dar baja". 6. Se realizan las operaciones necesarias para cambiar el estado del rol a desactivado.
Excepciones:	

Tabla 59

Gestionar permisos (fuente: elaboración propia)

Caso de uso (2):	Gestionar permisos de usuario		
Definición:	Modulo encargado de realizar operaciones CRUD con los datos de permisos para control de accesos al sistema.		
Prioridad:	☐ (1) Vital	☒ (2) Importante	☐ (3) Conveniente
Urgencia:	☐ (1) Inmediata	☒ (2) Necesario	☐ (3) Puede esperar
Actores			
Nombre:	Administrador		
Escenario 1			
Nombre:	Visualizar permiso		
Pre-Condiciones:	3. Usuario autenticado.		
Iniciado por:	Administrador		
Finalizado por:	Administrador		
Post-Condiciones:			
Operaciones:	8. Al ingresar al módulo. 9. Se visualiza la lista de permisos.		
Excepciones:			
Escenario 2			
Nombre:	Crear registro de permiso		
Pre-Condiciones:	3. Usuario autenticado.		
Iniciado por:	Administrador		
Finalizado por:	Administrador		
Post-Condiciones:	3. Debe rellenar todos los datos necesarios para crear el permiso		
Operaciones:	14. Se muestra un modal con formulario de información propia del permiso. 15. Se validan todos los campos requeridos. 16. Se presiona el botón "Crear". 17. Se realizan las operaciones necesarias para crear el registro de permiso.		
Excepciones:	Si los datos están mal ingresados, no debe permitir crear permisos y debe indicar al usuario exactamente donde ocurre la verificación fallida.		
Escenario 3			
Nombre:	Modificar información de permiso		
Pre-Condiciones:	5. Usuario autenticado. 6. Seleccionar usuario a modificar.		
Iniciado por:	Administrador		
Finalizado por:	Administrador		
Post-Condiciones:			
Operaciones:	11. Se selecciona el permiso a modificar. 12. Se presiona botón "Actualizar". 13. Se modifican los campos del formulario. 14. Se verifica el llenado de información requerida. 15. Se realizan las operaciones necesarias para actualizar la información del registro.		
Excepciones:	Si los datos están mal ingresados, no debe permitir modificar los datos y debe indicar al usuario exactamente donde ocurre la verificación fallida.		
Escenario 4			
Nombre:	Dar baja a permiso		

Pre-Condiciones:	5. Usuario autenticado 6. Seleccionar permiso a modificar
Iniciado por:	Administrador
Finalizado por:	Administrador
Post-Condiciones:	
Operaciones:	7. Se selecciona el usuario a dar baja. 8. Se presiona botón "Dar baja". 9. Se realizan las operaciones necesarias para cambiar el estado del permiso a desactivado.
Excepciones:	

1. [Download the software](#)

Anexo L. Otros diagramas

Figura 32

Caso de uso: Gestión de incidencias

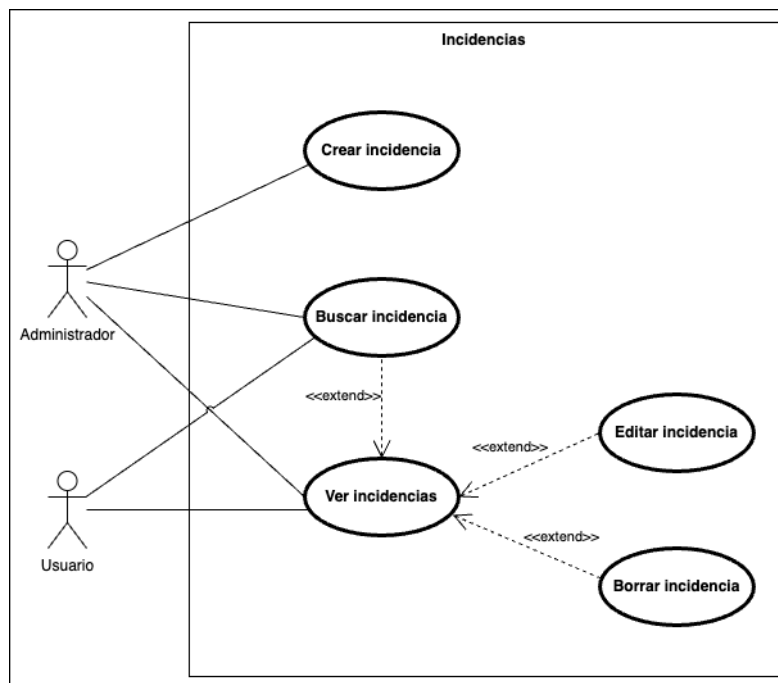


Figura 33

Caso de uso: Gestión de catálogo

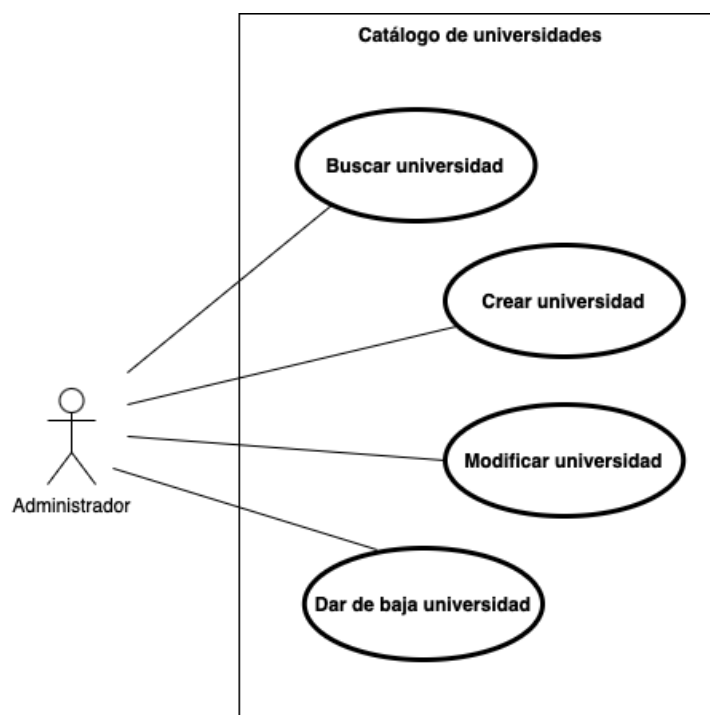


Figura 34

Diagramas de actividades de incidencias

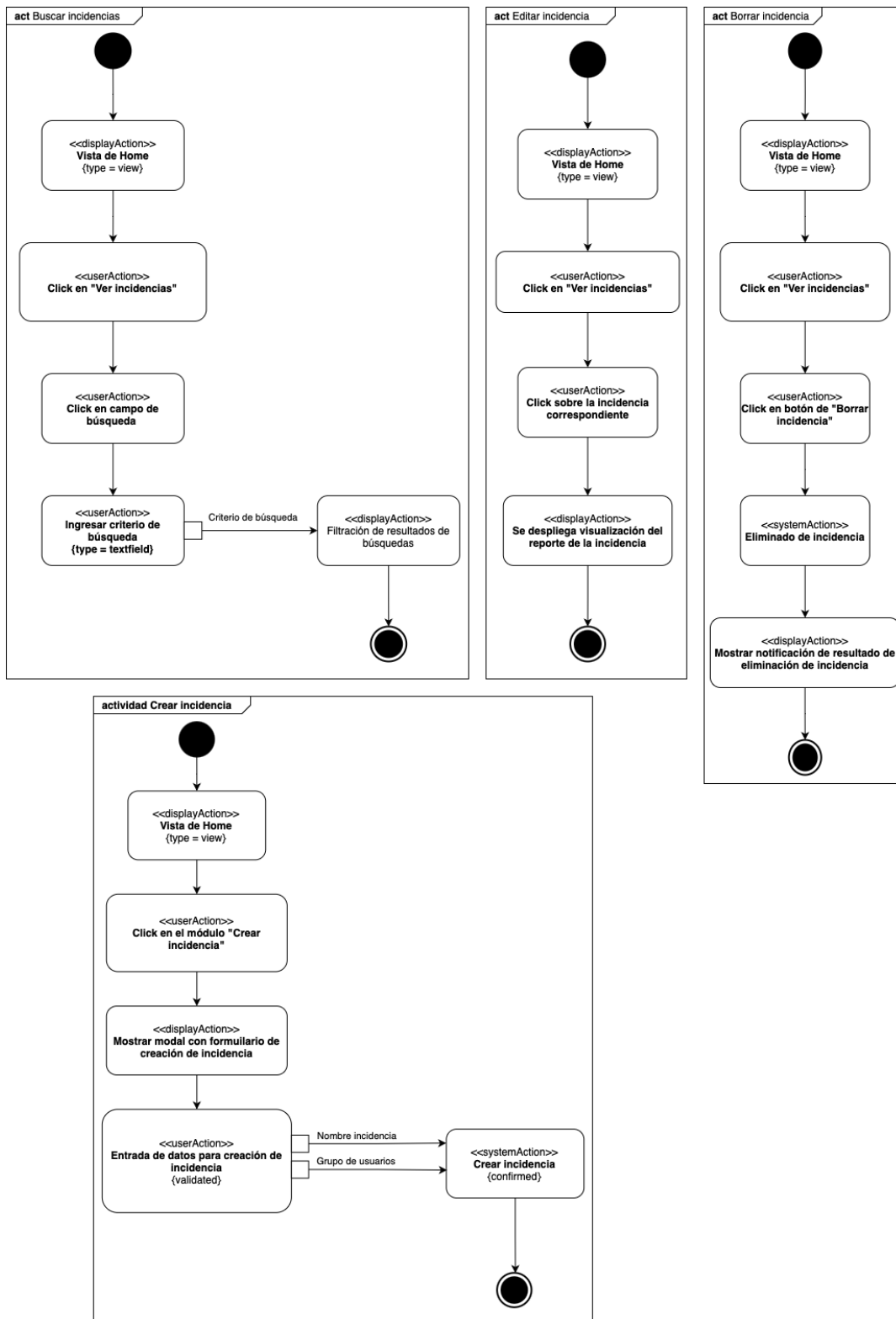


Figura 35

Diagrama actividad de buscar universidades

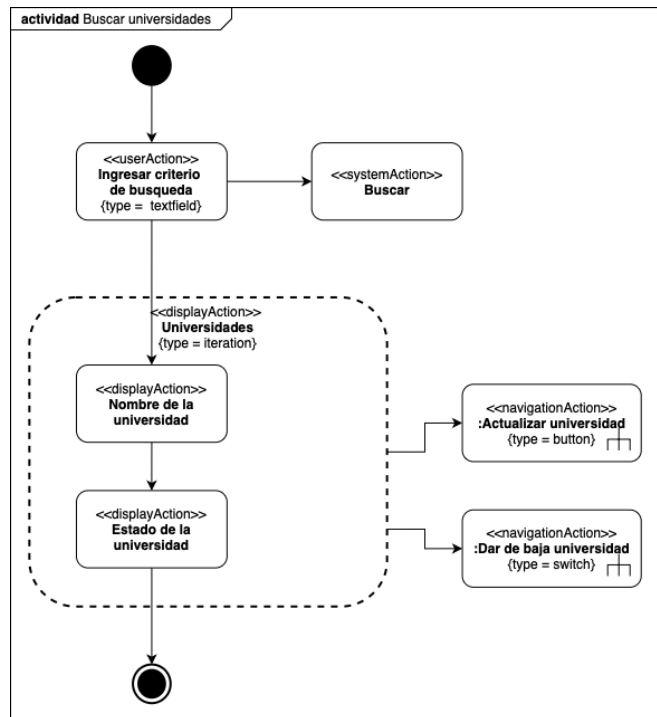


Figura 36

Diagrama de actividad de actualizar universidad

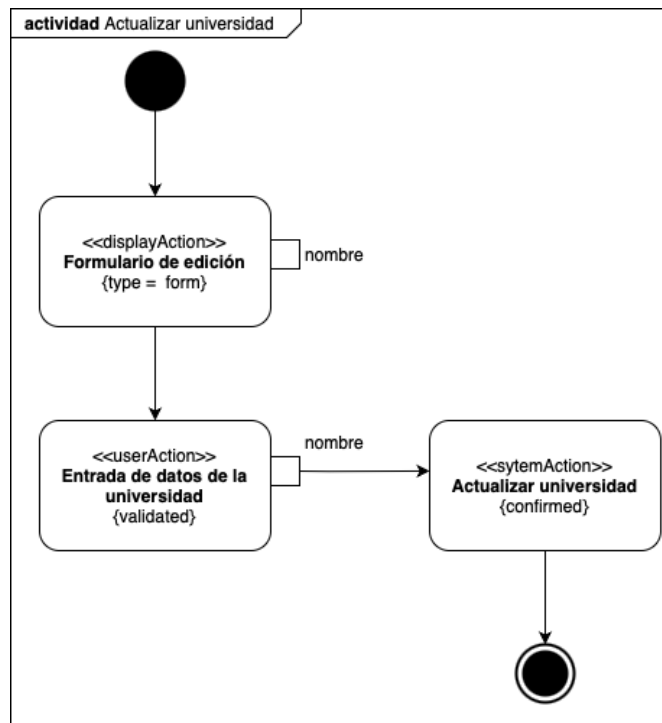


Figura 37

Diagrama de actividad de dar de baja universidad

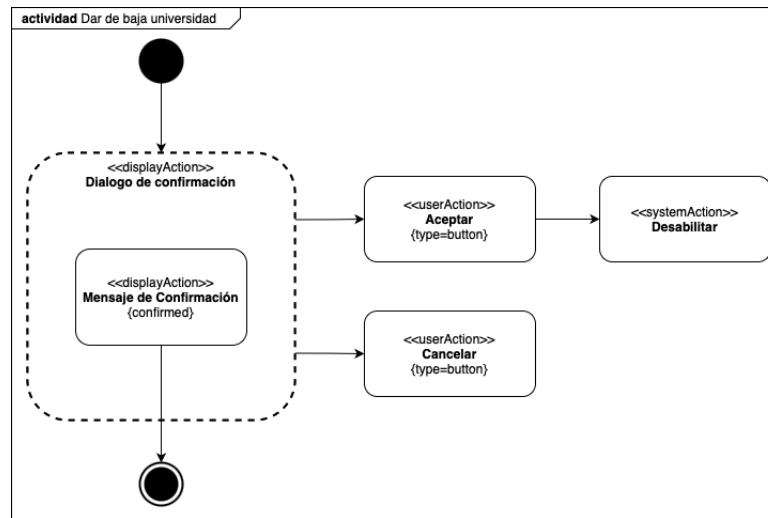


Figura 38

Diagrama de actividad de crear usuario

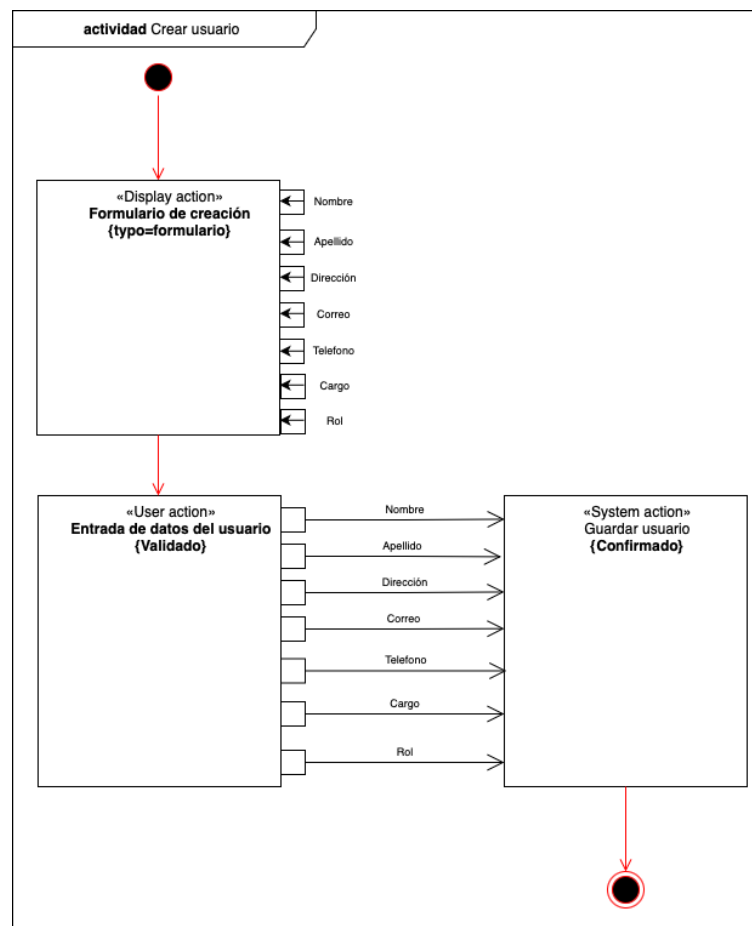


Diagrama de actividad de editar usuario



Diagrama de actividad de dar de baja un usuario

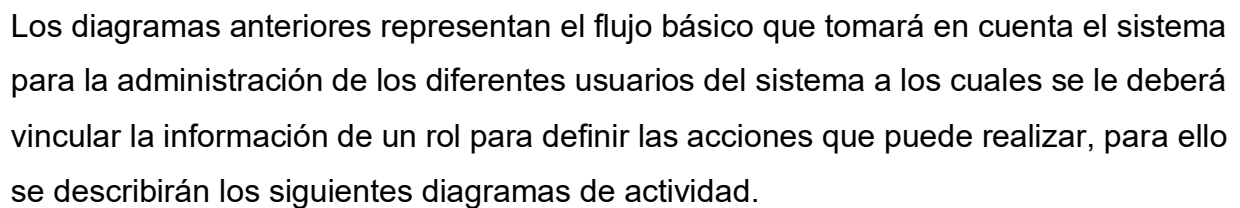


Figura 41

Diagrama actividad de creación de rol

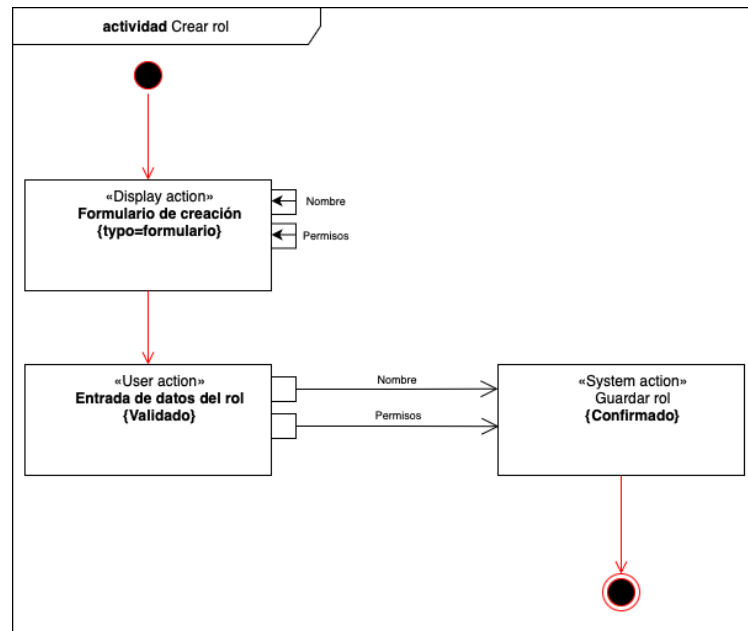


Figura 42

Diagrama de actividad de dar de baja un rol

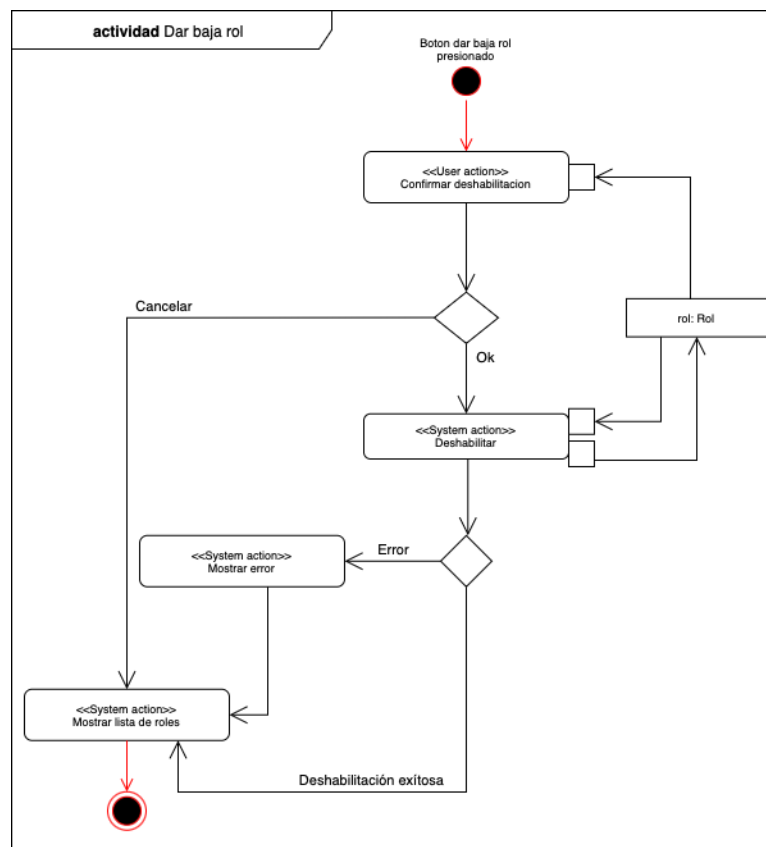
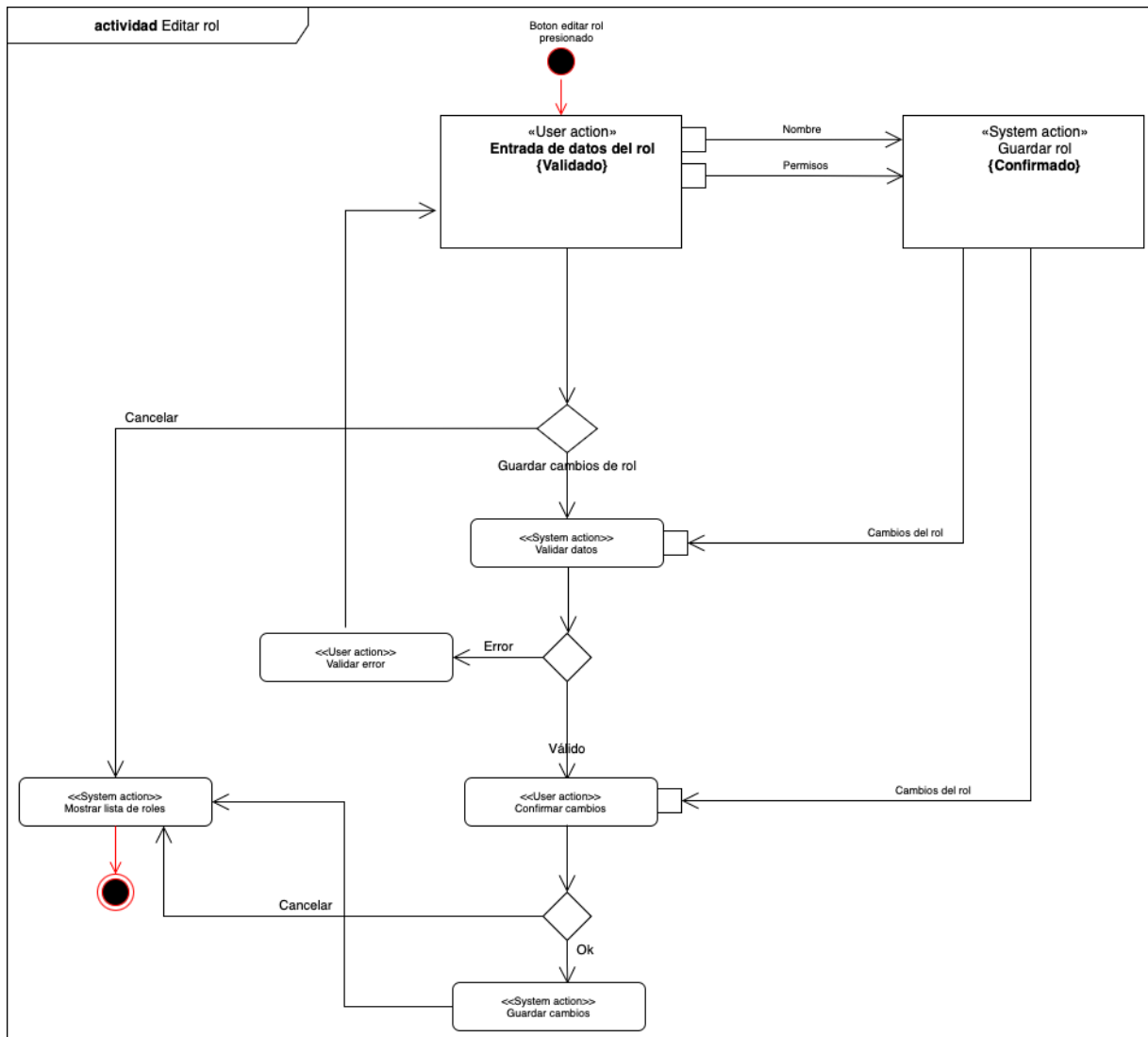


Figura 43

Diagrama de actividad de editar un rol



Como se puede apreciar en los diagramas anteriores, un rol está conformado por un conjunto de permisos los cuales representa los niveles de accesos de los usuarios vinculados a determinado rol, por ello también presentamos las actividades básicas relacionadas al proceso de la administración de permisos.

Figura 44

Diagrama de actividad de creación de permiso

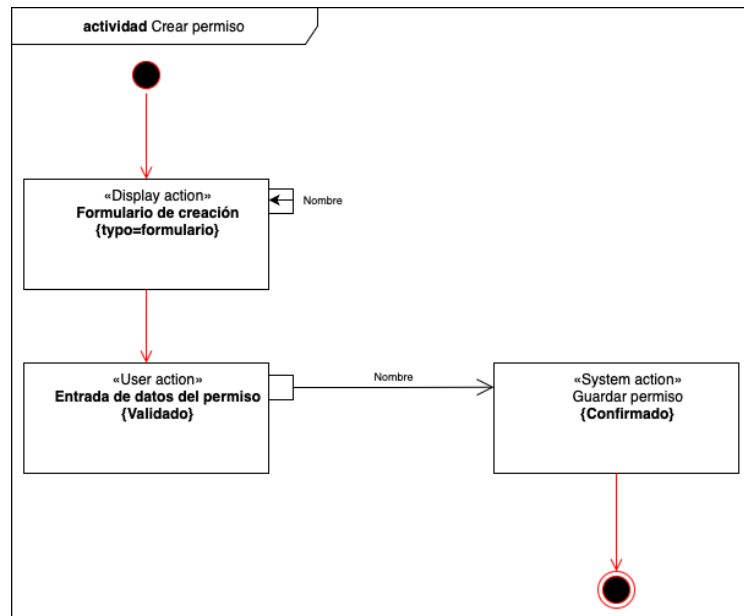


Figura 45

Diagrama de actividad de dar de baja un permiso

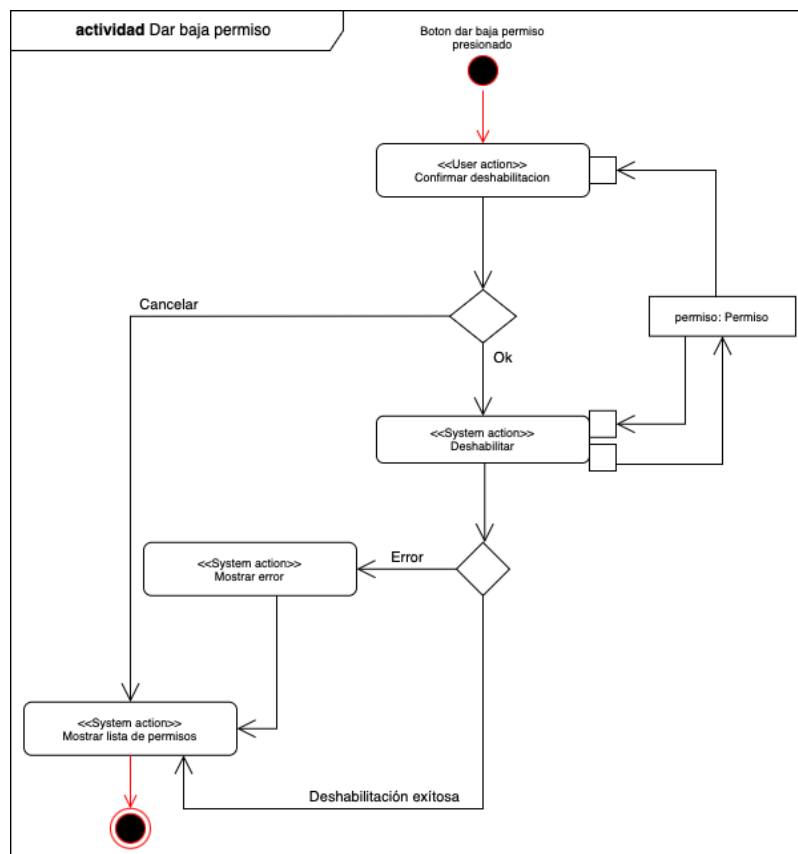


Figura 46

Diagrama de actividad edición de permiso

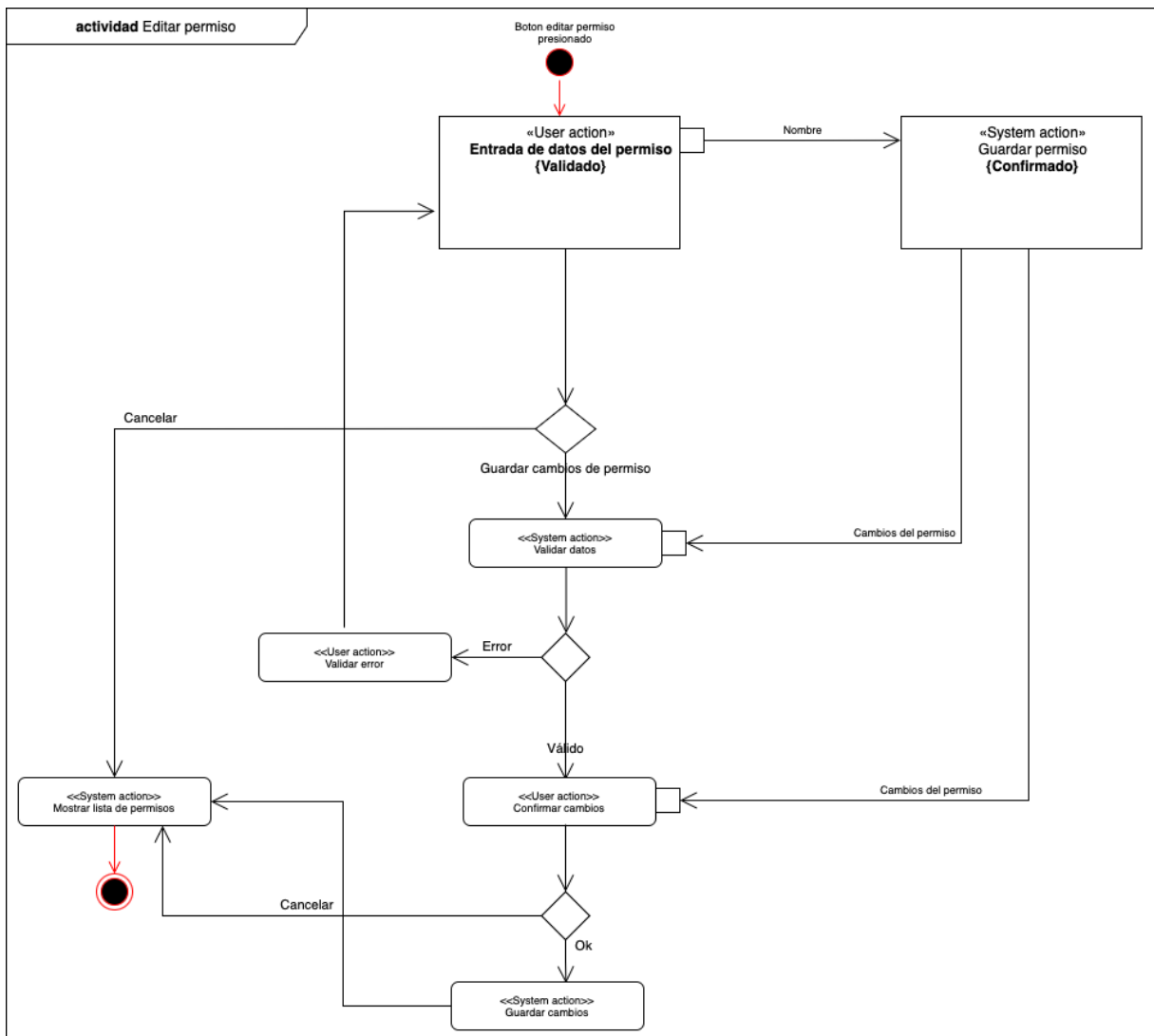


Figura 47

Diagrama de proceso - Actualizar universidad

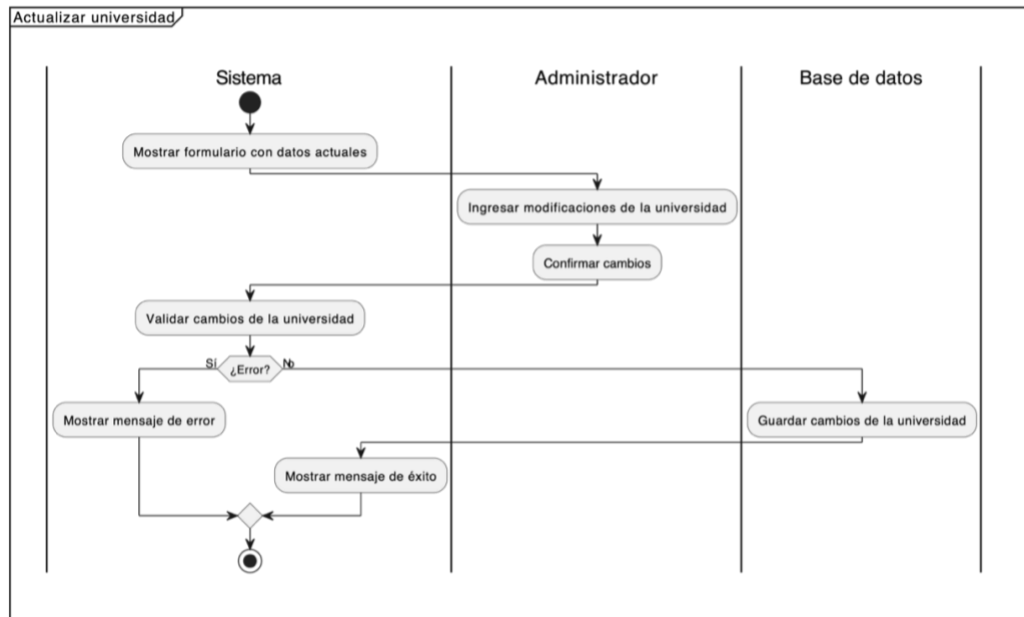


Figura 48

Diagrama de proceso - Dar de baja universidad

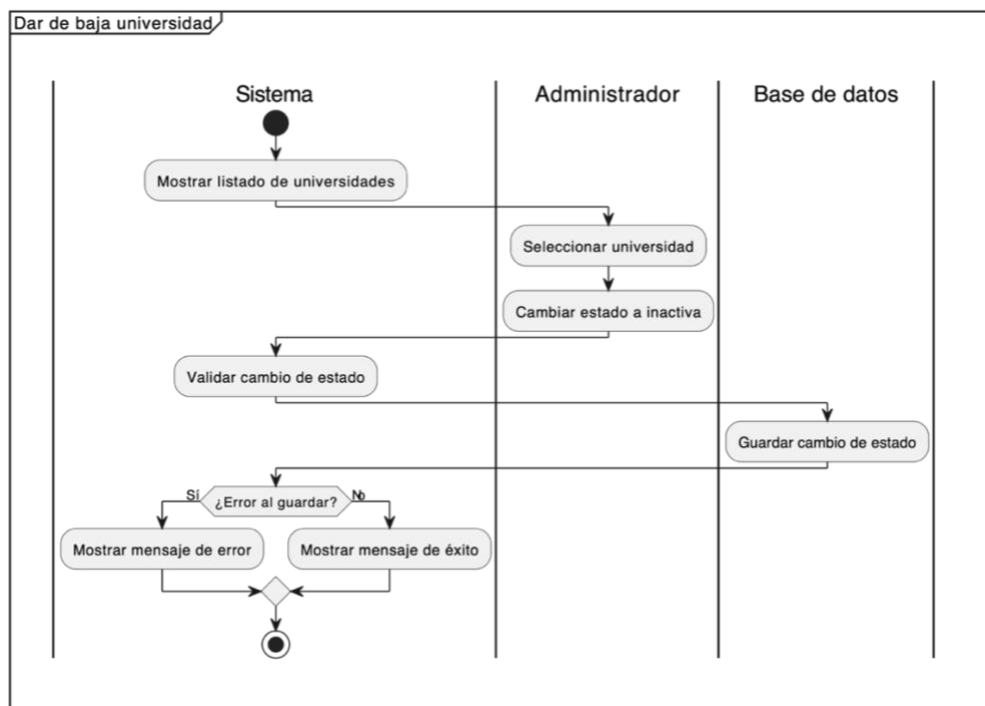


Figura 49

Diagrama de proceso - Descargar informe

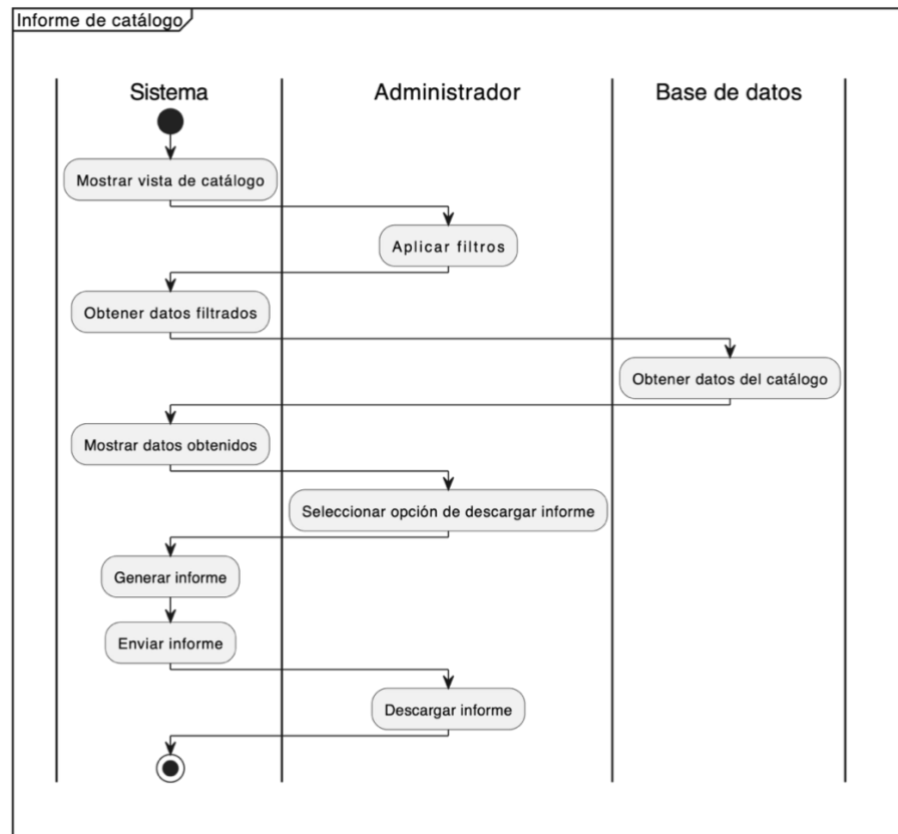


Figura 50

Diagrama de proceso - Crear permiso

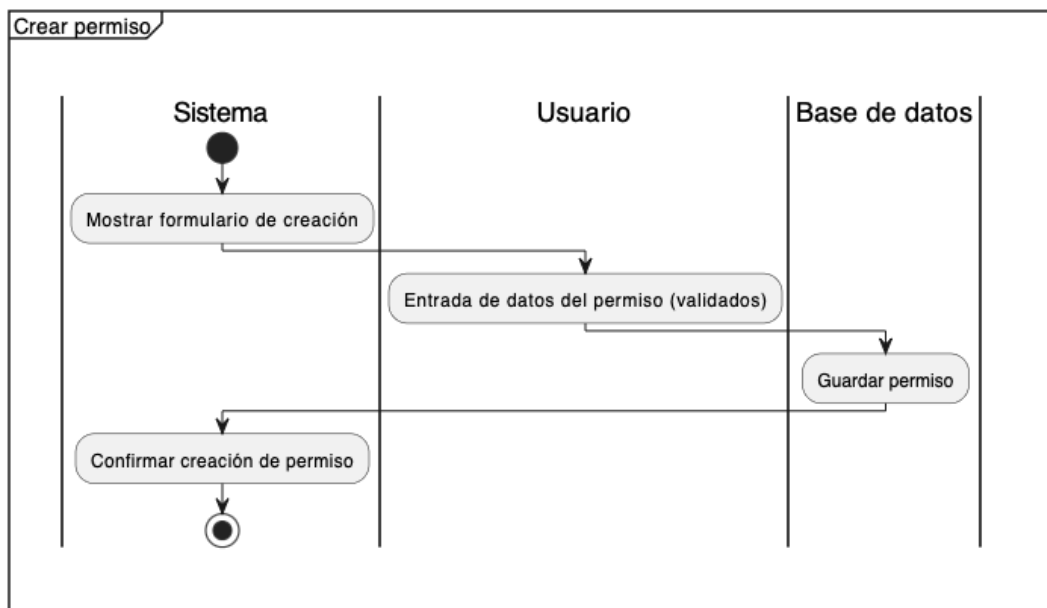


Figura 51

Diagrama de proceso - Dar de baja a permiso

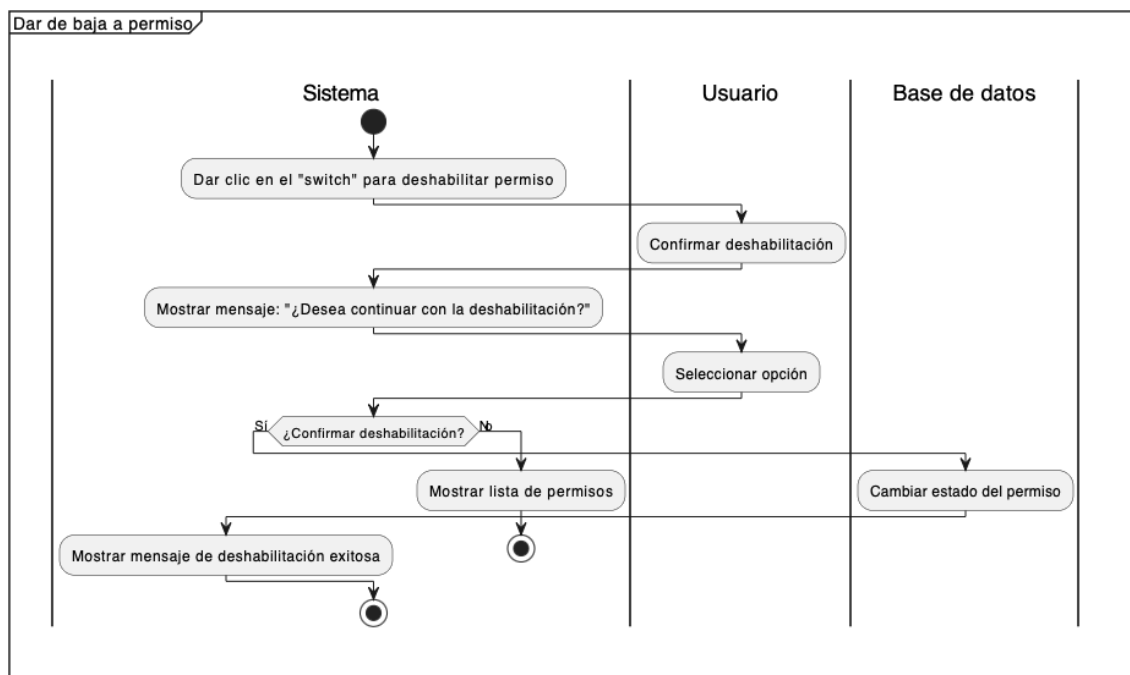


Figura 52

Diagrama de proceso - Editar permiso

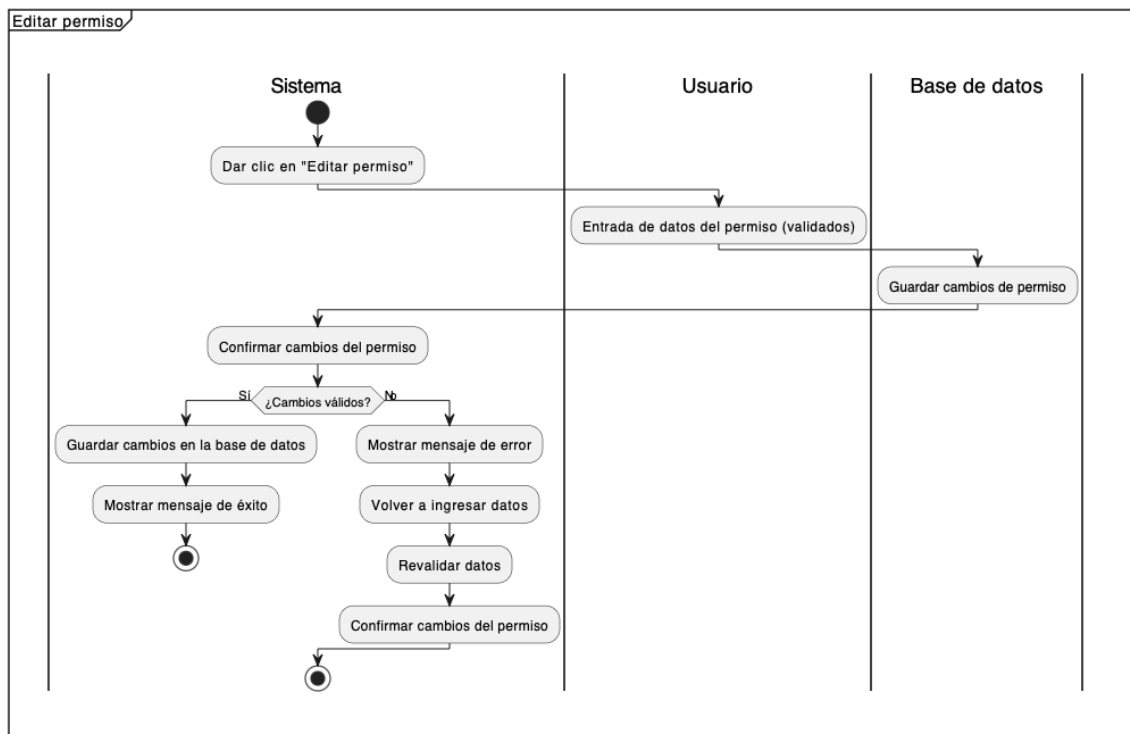


Figura 53

Diagrama de proceso - Crear rol

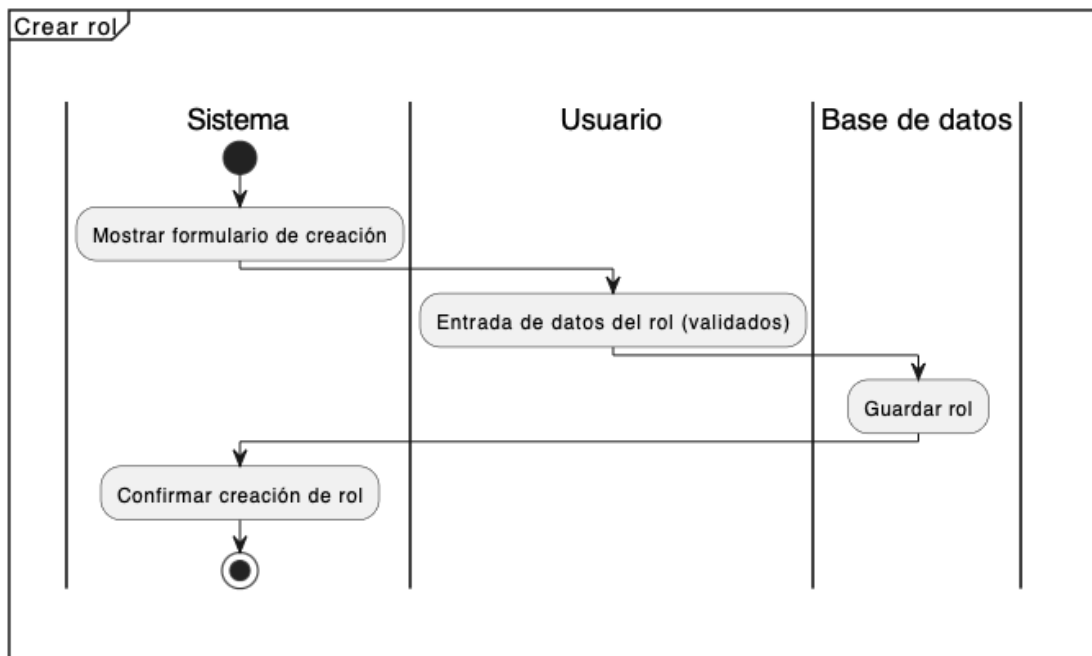


Figura 54

Diagrama de proceso - Dar de baja a rol

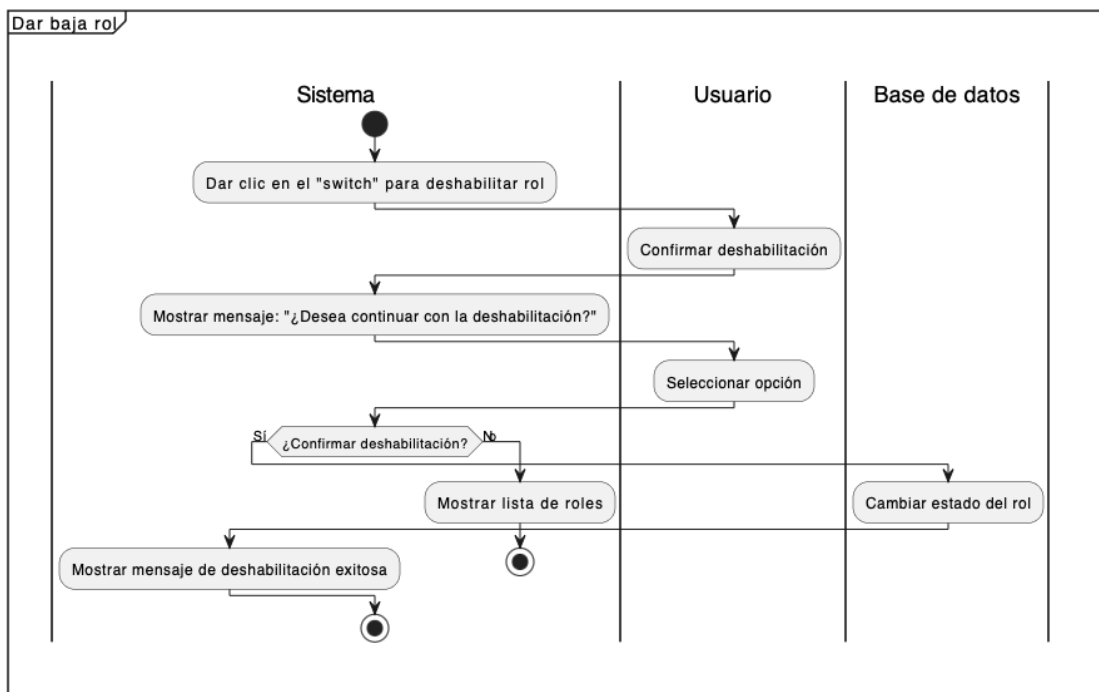


Figura 55

Diagrama de proceso - Editar rol

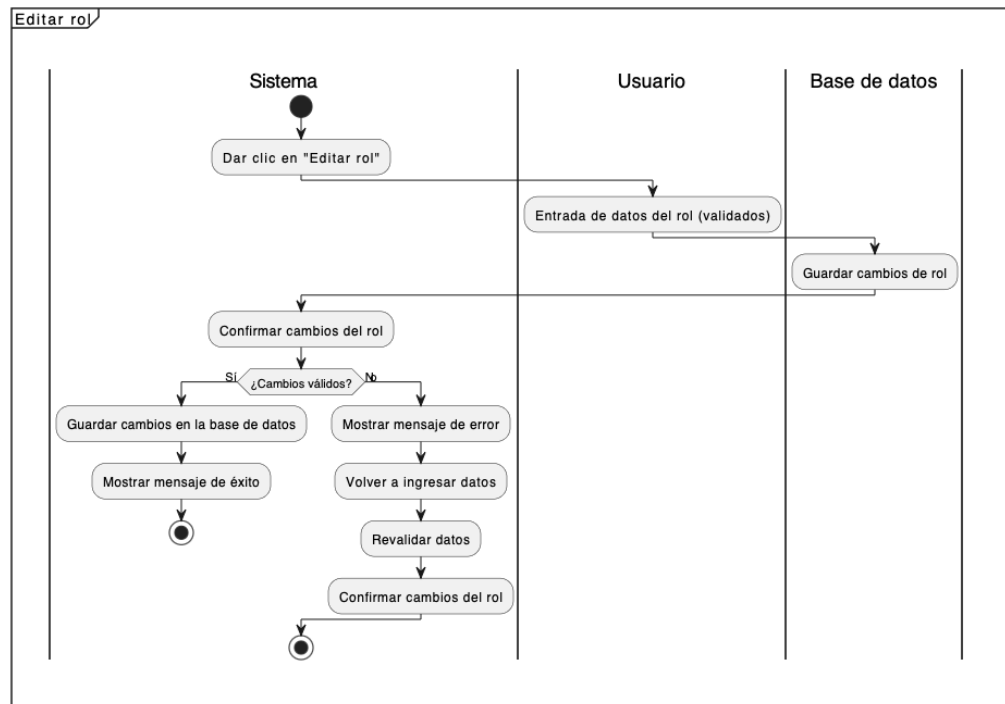


Figura 56

Diagrama de proceso - Crear usuario

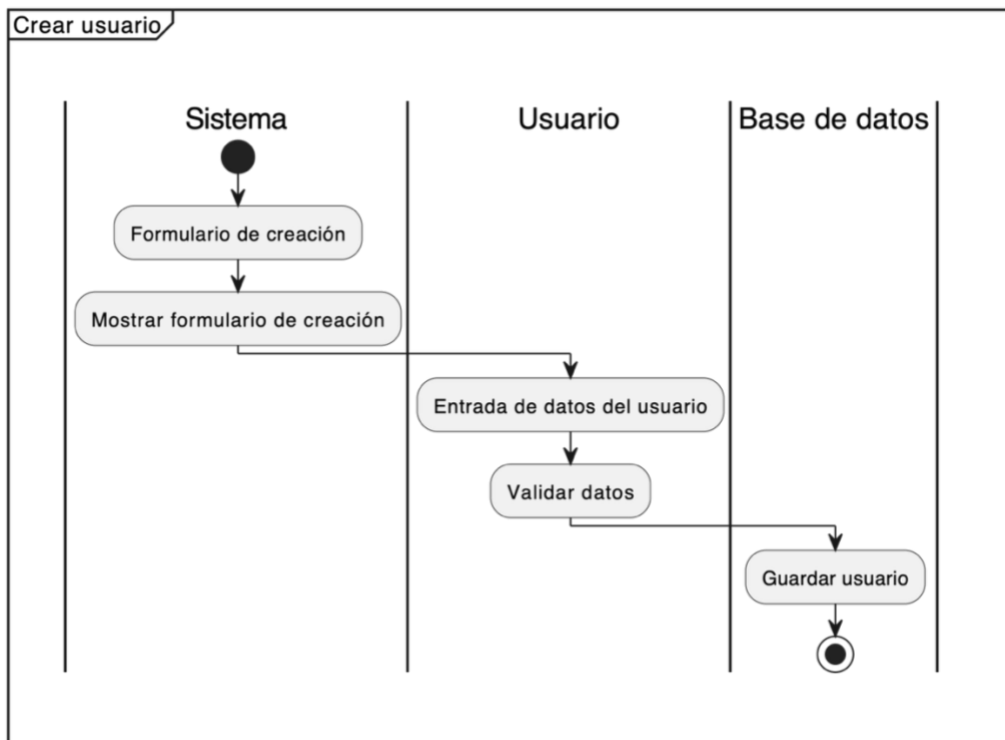


Figura 57

Diagrama de proceso - Dar de baja a usuario

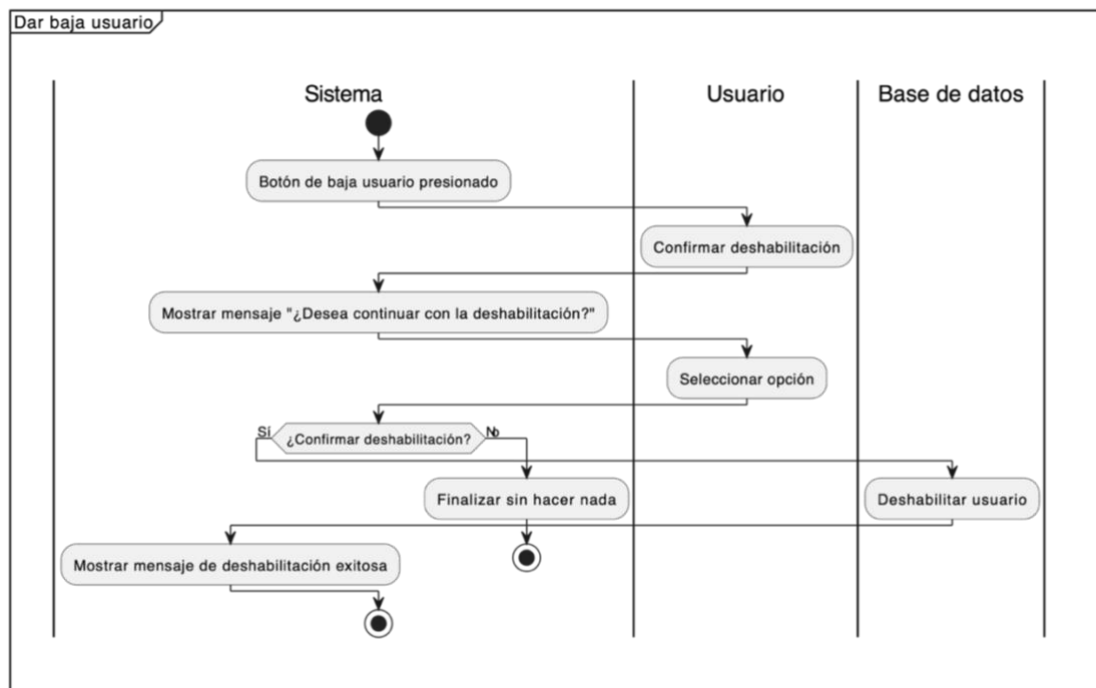
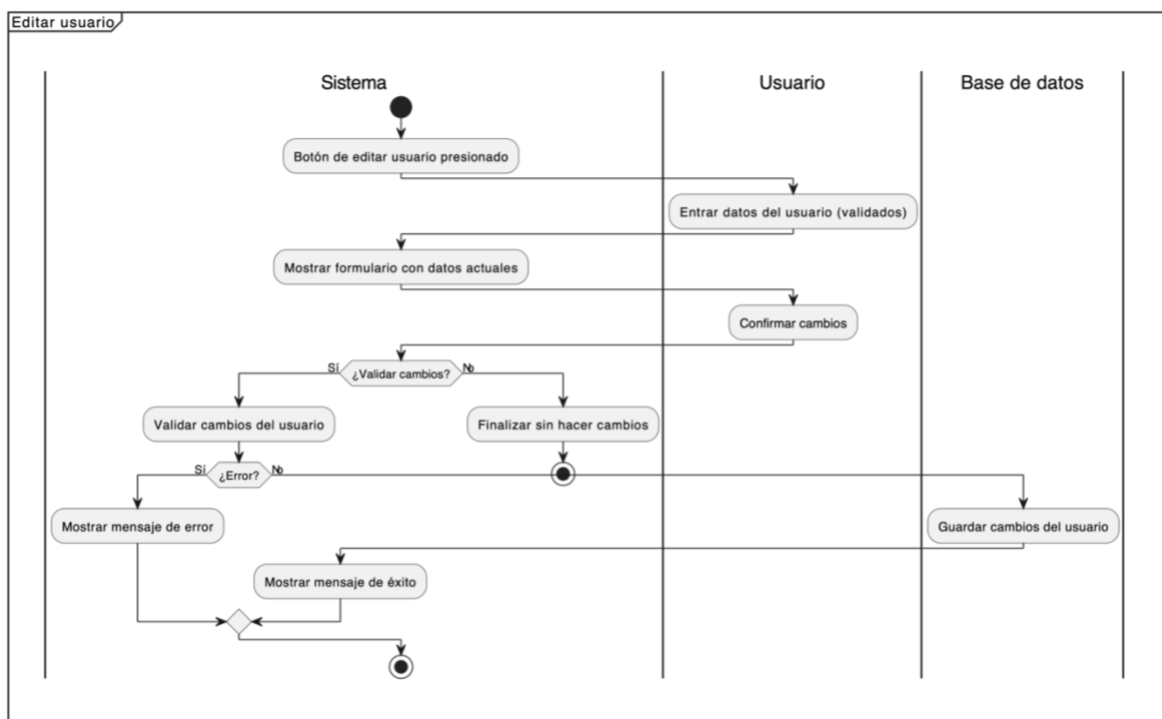


Figura 58

Diagrama de proceso - Editar usuario



Anexo M. Imágenes del sistema

Figura 59

Inicio de sesión del sistema

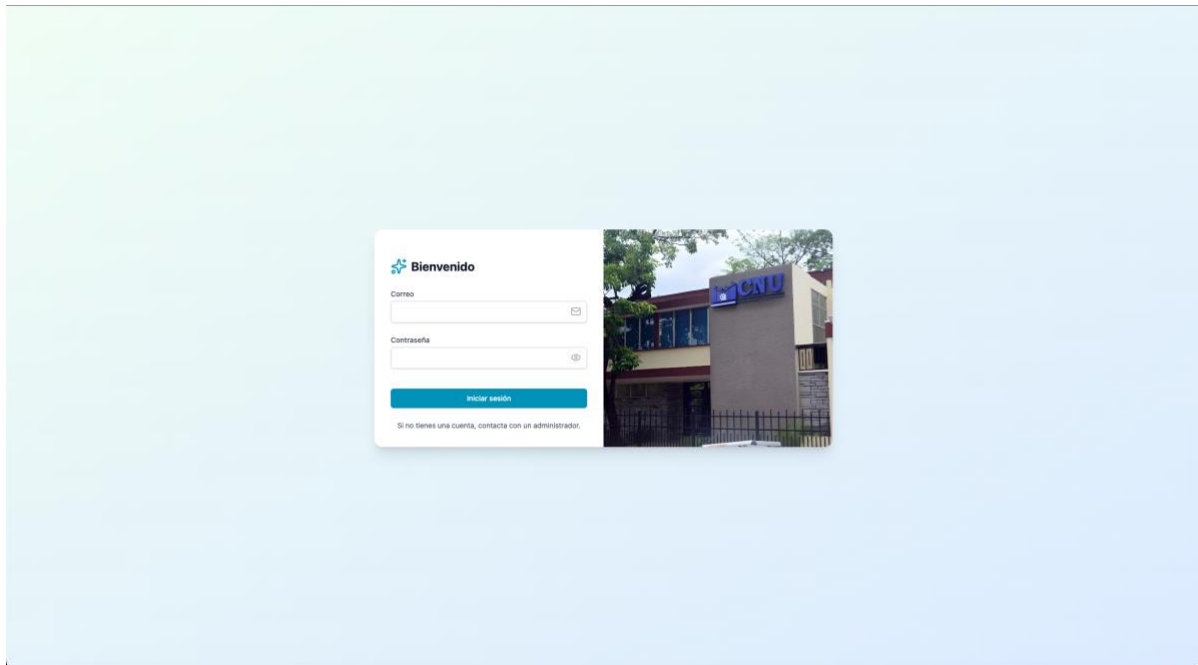


Figura 60

Vista de inicio

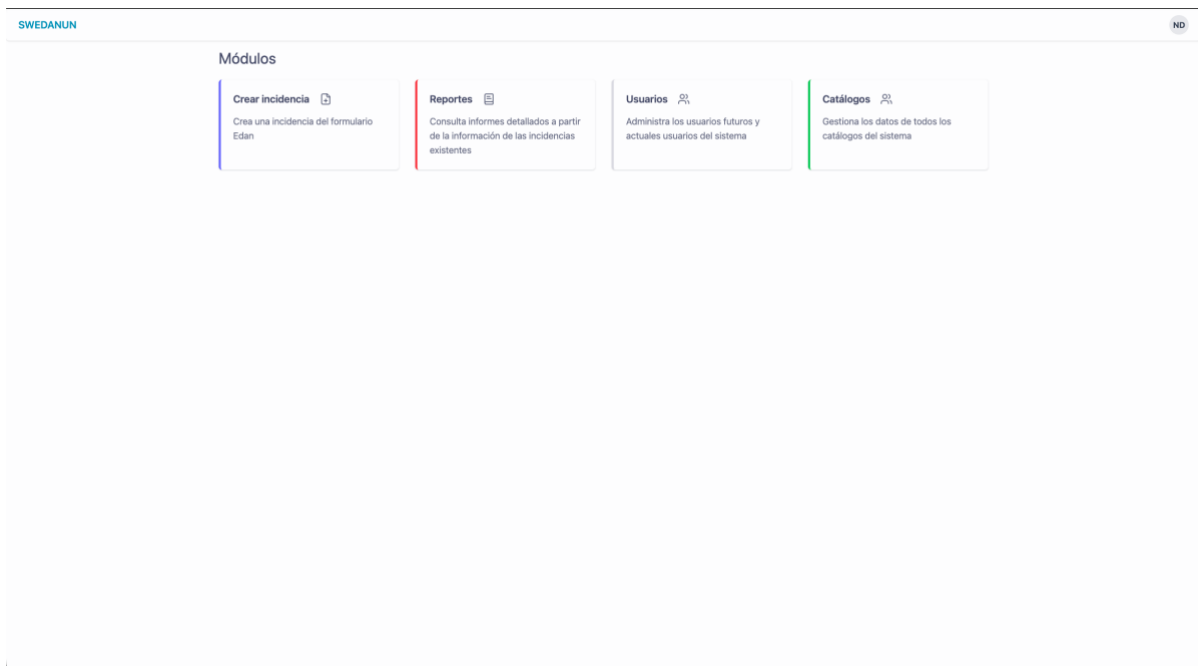


Figura 61

Vista principal de gestión de incidencias

SWEDANUN

Crear Incidencia Reportes Usuarios Catálogos Configuración EN

Incidencias

Gestiona las incidencias de formularios EDAN

+ Nueva Incidencia

Mostrar 10 registros por página

Buscar elementos

#	NOMBRE DE LA INCIDENCIA	EDAN	OPERADORES	ESTADO	
1	Evaluación Infraestructura Campus Principal	EDAN 72	2		
2	Evaluación Sistema de Agua FAREM	EDAN 120	2		
3	Inspección Edificios Campus Médico	EDAN 72	4		
4	Evaluación Servicios Básicos UCA	EDAN 120	2		
5	Diagnóstico Instalaciones UPOLI	EDAN 72	5		
6	Revisión Infraestructura Estelí	EDAN 72	3		
7	Evaluación Seguridad Edificios UPOLI Estelí	EDAN 120	3		
8	Revisión Sistema Eléctrico Chinandega	EDAN 72	4		
9	Análisis Daños Estructurales UCA	EDAN 120	2		
10	Evaluación Recursos Hídricos Matagalpa	EDAN 72	3		

Viendo 1 al 10 de 14 registros

<< 1 2 >>

Figura 62

Creación de incidencia

SWEDANUN

Crear Incidencia Reportes Usuarios Catálogos Configuración EN

Incidencias

Gestiona las incidencias de formularios EDAN

+ Nueva Incidencia

Mostrar 10 registros por página

Buscar elementos

Crear Nueva Incidencia

Nombre de la incidencia

Ingrese el nombre de la incidencia

Tipo de EDAN Campus

EDAN 72 Seleccione un campus

Operadores

Jose Guerrero ✓

Jose Dubon ✓

Marcos Zuniga

Dennis Hernandez

2 operadores seleccionados

Detalles: Al crear la incidencia del tipo seleccionado no se podrá editar más adelante.

Crear

#	NOMBRE DE LA INCIDENCIA	EDAN	OPERADORES	ESTADO	
1	Evaluación Infraestructura Campus Principal	EDAN 72	2		
2	Evaluación Sistema de Agua FAREM	EDAN 120	2		
3	Inspección Edificios Campus Médico	EDAN 72	4		
4	Evaluación Servicios Básicos UCA	EDAN 120	2		
5	Diagnóstico Instalaciones UPOLI	EDAN 72	5		
6	Revisión Infraestructura Estelí	EDAN 72	3		
7	Evaluación Seguridad Edificios UPOLI Estelí	EDAN 120	3		
8	Revisión Sistema Eléctrico Chinandega	EDAN 72	4		
9	Análisis Daños Estructurales UCA	EDAN 120	2		
10	Evaluación Recursos Hídricos Matagalpa	EDAN 72	3		

Viendo 1 al 10 de 14 registros

<< 1 2 >>

Figura 63

Formulario Edan 120

SWEDANUN Crear incidencia Reportes Usuarios Configuración EN

SECCIONES

- 1. EVALUACIÓN DE DAÑOS EN EL SUMINISTRO DE AGUA POTABLE
- 2. PARÁMETROS DE CALIDAD DEL AGUA
- 3. RESUMEN DE NECESIDADES EN LA RED DE AGUA POTABLE
- 4. EVALUACIÓN DE DAÑOS EN SANEAMIENTO
- 5. RESUMEN DE NECESIDADES EN LA RED DE ALCANTARILLADO SANITARIO
- 6. SISTEMAS DE TRATAMIENTO EN LA UNIVERSIDAD
- 7. EVALUACIÓN DEL SISTEMA DE TRATAMIENTO DE AGUAS RESIDUALES EN LA UNIVERSIDAD
- 8. ACTIVIDAD AGRÍCOLA

FORMATO No.2 DE EVALUACIÓN DE DAÑOS Y ANÁLISIS DE NECESIDADES A LAS 120 HRS (ESPECIALIZADO)

1. EVALUACIÓN DE DAÑOS EN EL SUMINISTRO DE AGUA POTABLE

Tipo de evento: * Fecha: *
 DD / MM / YYYY - HH:MM

Nombre del evaluador: *

Institución: * Cargo: *

Fecha evaluación: * Region: *
 DD / MM / YYYY

Departamento: * Municipio: *

Universidad: * Población servida: *

Clima: * Energía Eléctrica: *
 SI

Figura 64

Formulario Edan 72

SWEDANUN Crear incidencia Reportes Usuarios Catálogos Configuración EN

SECCIONES

- 1. DATOS GENERALES
- 2. VÍAS DE ACCESO
- 3. AFECTACIÓN EN LA INFRAESTRUCTURA DE AGUA Y SANEAMIENTO
- 4. AFECTACIÓN EN LA ACTIVIDAD AGRÍCOLA
- 5. AFECTACIÓN PECUARIA
- 6. DAÑOS EN LA COBERTURA FORESTAL

FORMATO No. 1 DE EVALUACIÓN DE DAÑOS Y ANÁLISIS DE NECESIDADES (EVALUACIÓN PRELIMINAR A LAS 72 HORAS POST EVENTO) POR UNIVERSIDAD

1. DATOS GENERALES

TIPO DE EVENTO: * FECHA: *
 DD / MM / YYYY

HORA: *
 HH:MM

DATOS DEL ENTREVISTADO (NOMBRE/CARGO): *

REGIÓN: * DEPARTAMENTO: * MUNICIPIO: *

UNIVERSIDAD: *

DIRECCIÓN DE LA UNIVERSIDAD O RECINTO: * TELÉFONO: *

POBLACIÓN UNIVERSITARIA: * VARONES: * MUJERES: *

PERSONAL QUE NO PUEDEN INCORPORARSE A LABORES DEBIDO A LA EMERGENCIA:

ESTUDIANTES QUE NO PUEDEN INCORPORARSE A CLASES DEBIDO A LA EMERGENCIA:

Figura 65
Modulo de gestión de usuarios

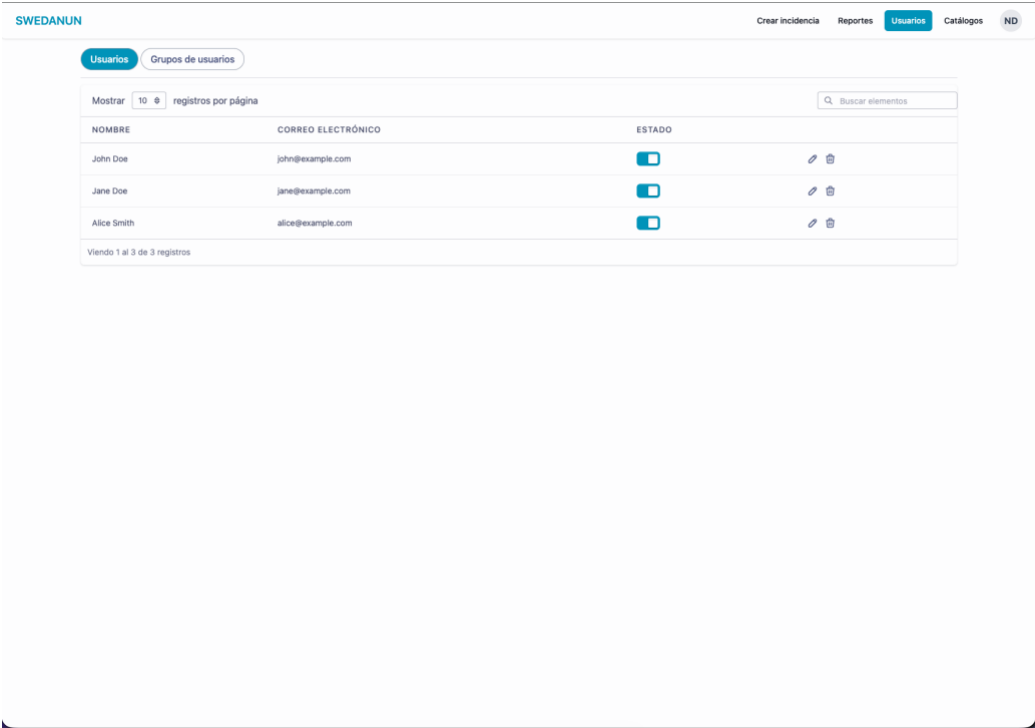


Figura 66
Módulo de gestión de catálogos

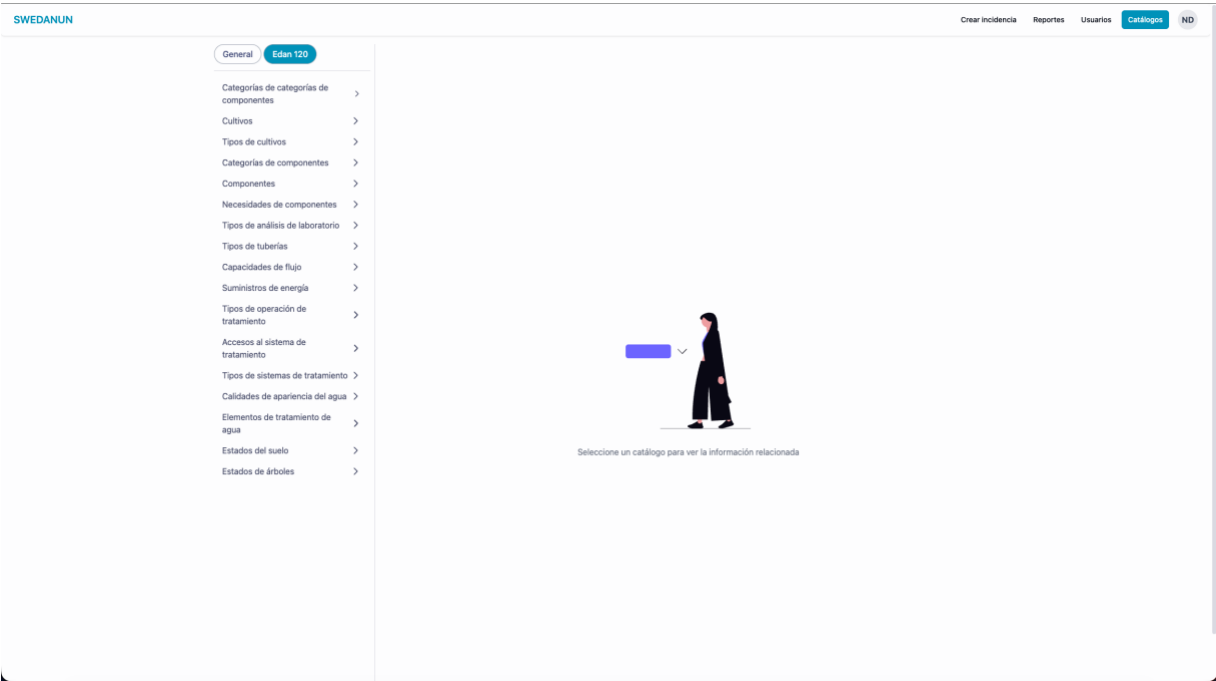


Figura 67

Detalle de un catalogo

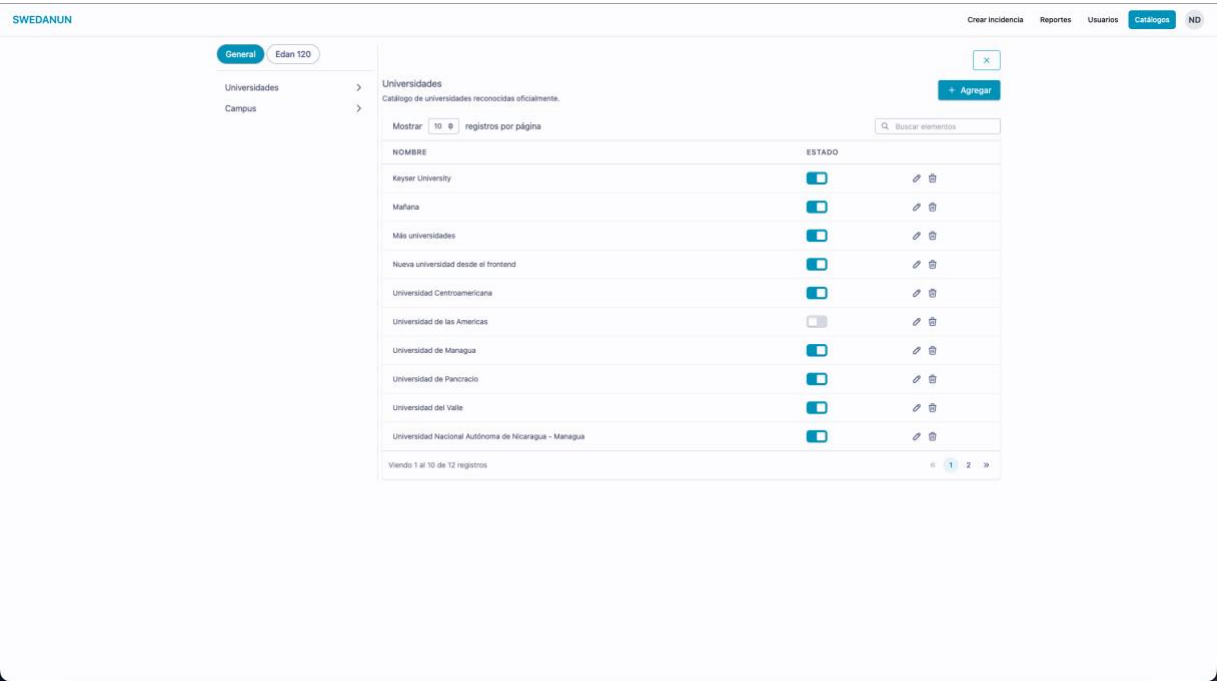


Figura 68

Reporte departamental

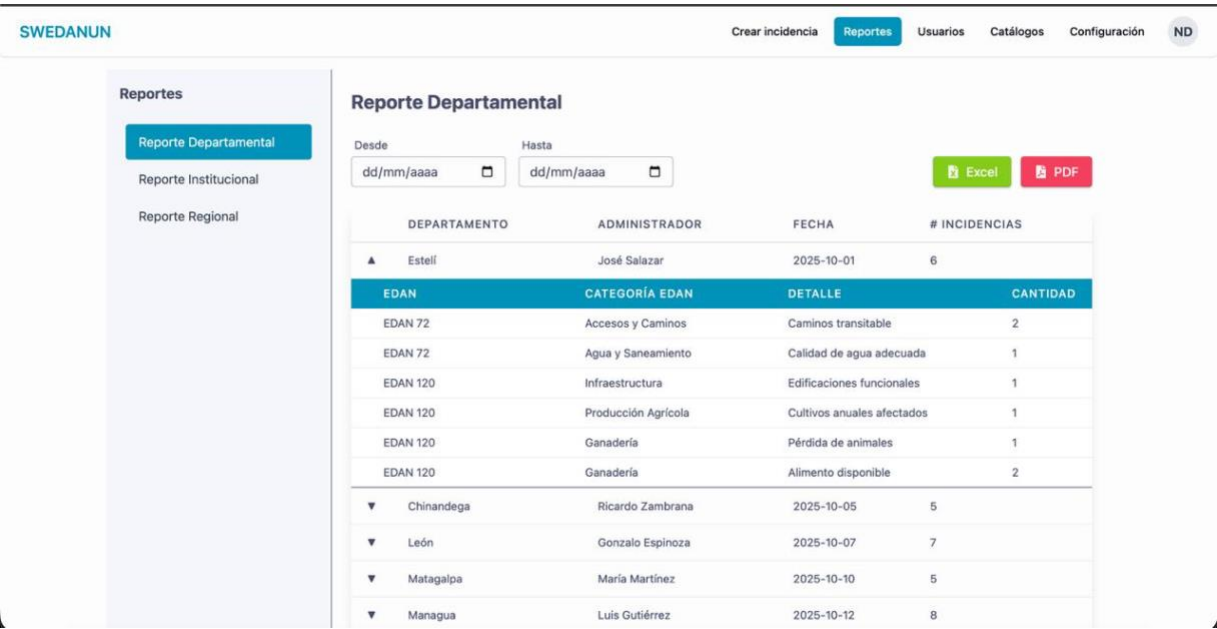


Figura 69

Reporte institucional

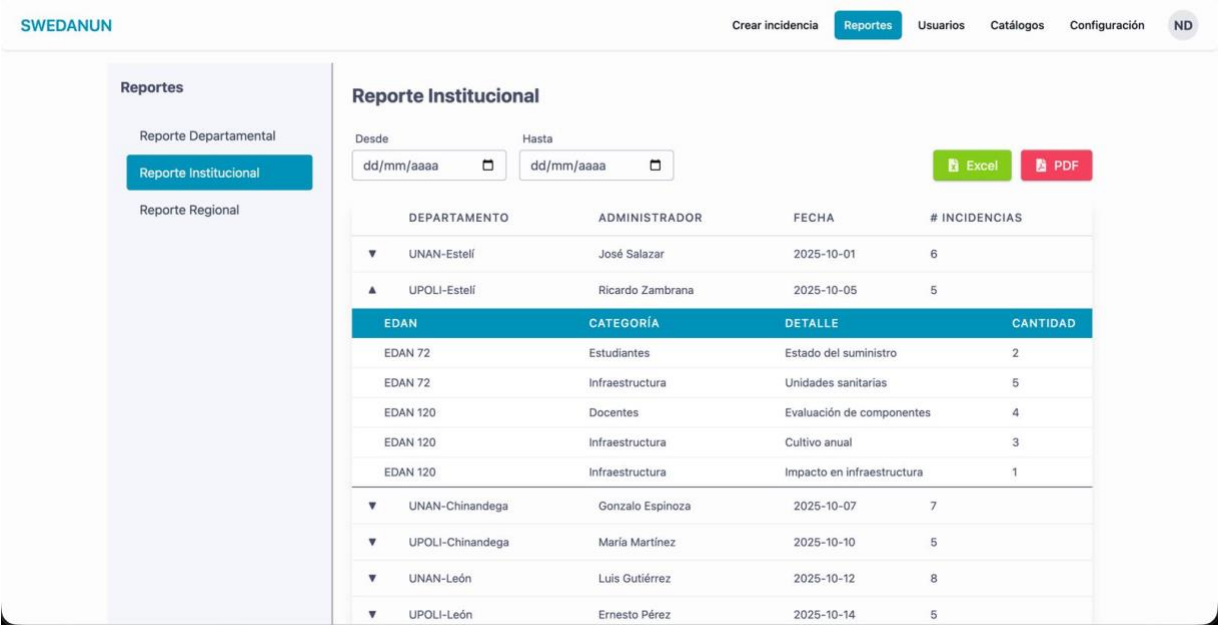
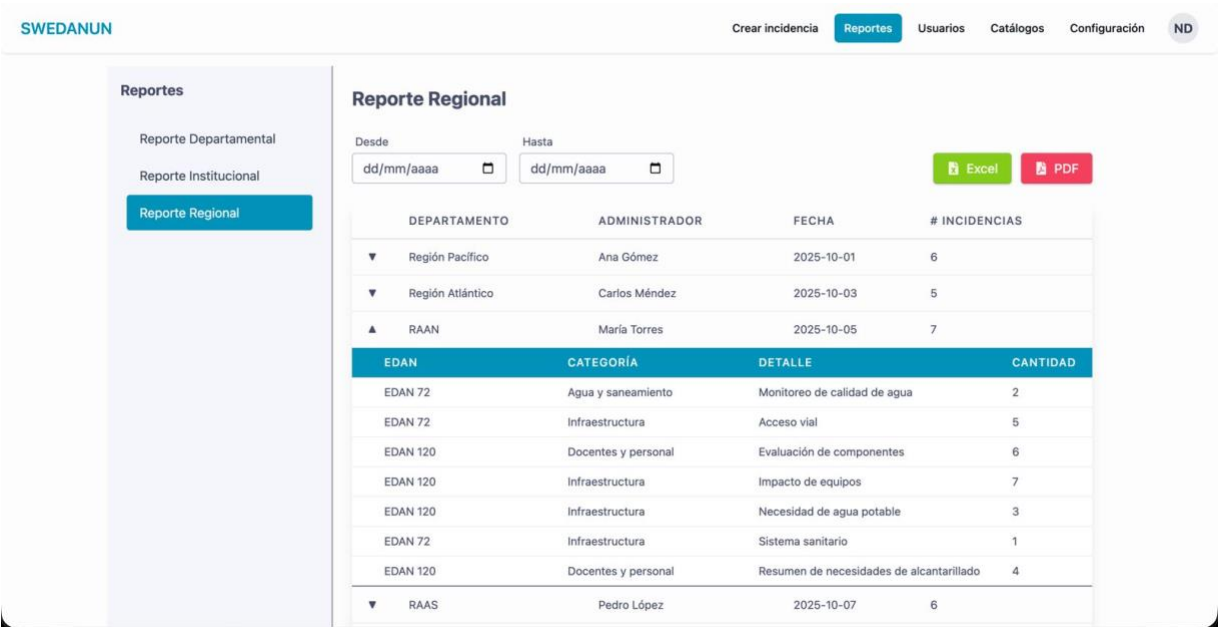


Figura 70

Reporte regional



Anexo N. Estructura de carpetas backend y frontend

Figura 71

Estructura de carpetas del backend

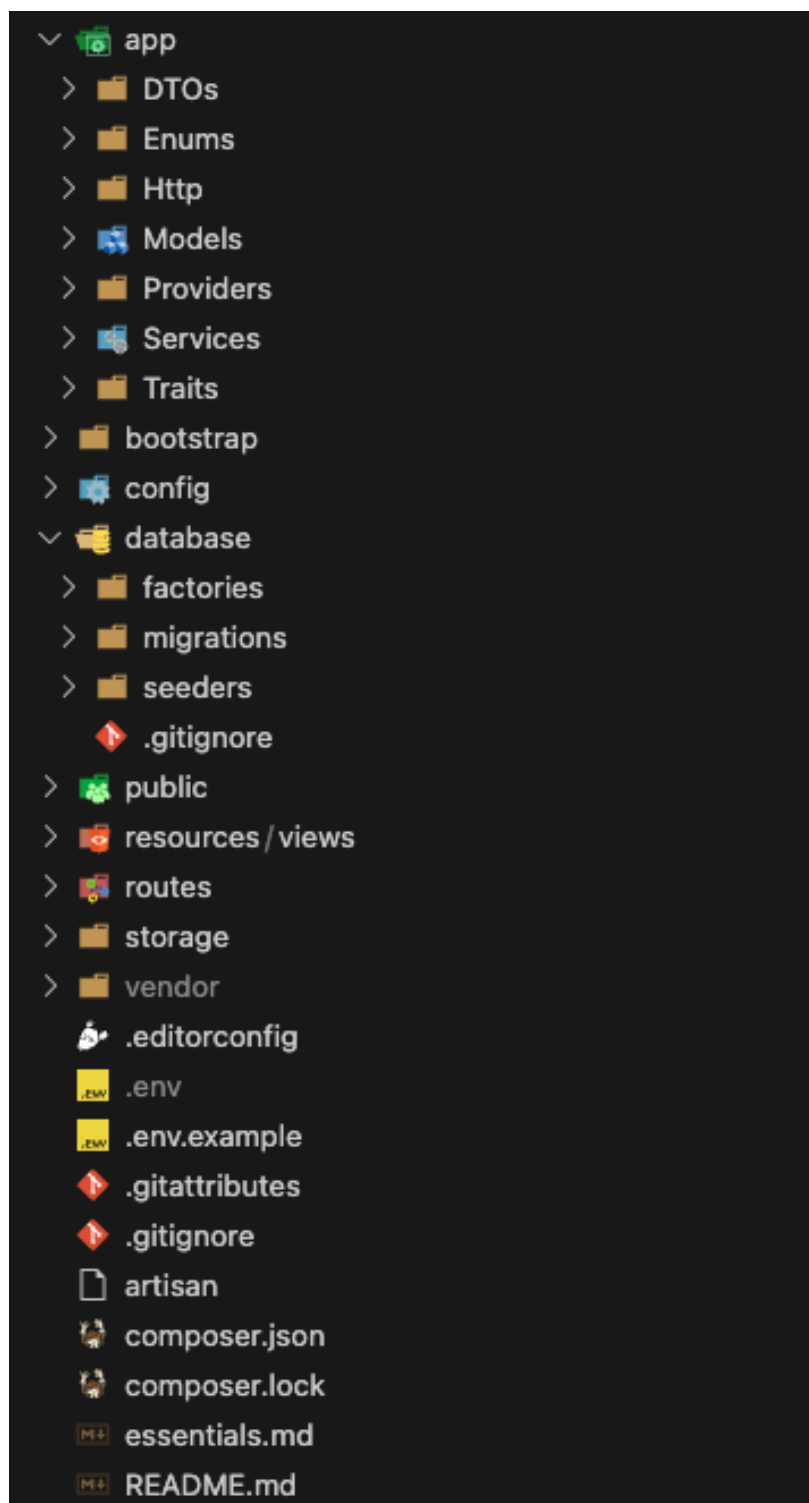
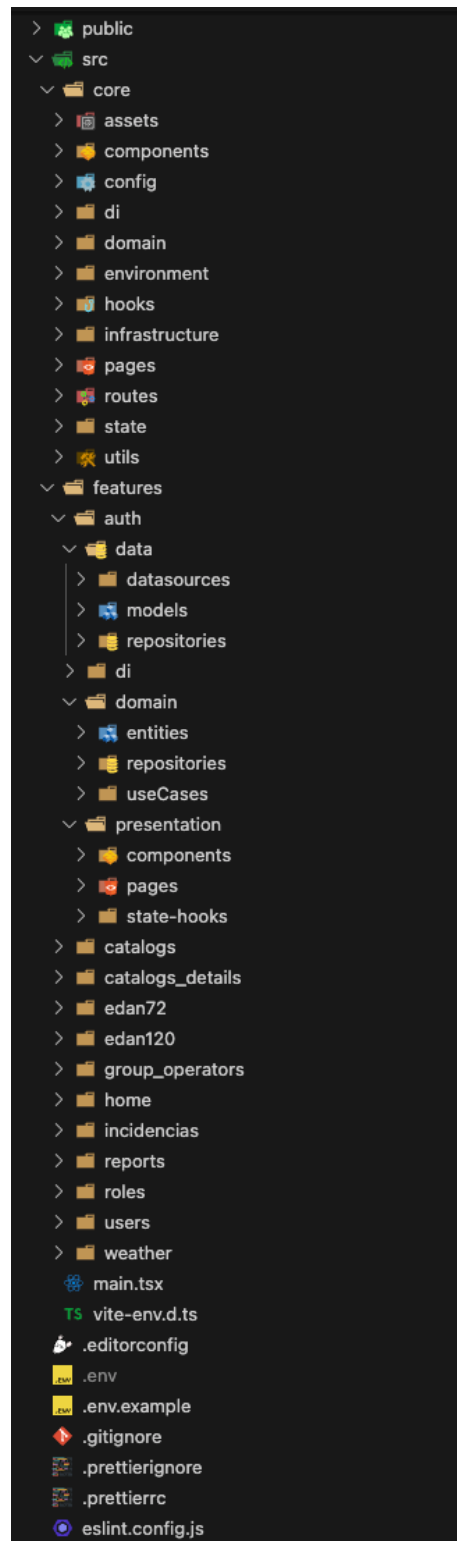


Figura 72

Estructura de carpetas del frontend



[illegible]

[illegible][illegible][illegible][illegible][illegible][illegible][illegible][illegible][illegible][illegible]

Afectaciones	Cantidad	Costo por unidad en cóndulas	Total de la inversión en cóndulas
Unidad de pasto (cuya)			
Unidad de pasto de corta (cuya)			
Volumen de pasto fertilizado (cuya)			
Volumen de pasto sembrado (cuya)			
Volumen de pasto feno (kg) a comprar (cuya)			

[illegible][illegible]