

Facultad de Ciencias y Sistemas

“Sistema de información para la automatización en la generación de nómina en la empresa Distribuidora General, S.A. (DIGESA)”

Trabajo Monográfico para optar al título de
Ingeniero de Sistemas

Elaborado por

Br. Eddy Vladimir
Sarantez Jarquín
Carnet: 2005-21260

Br. Jeferson Francisco
Cano Calero
Carnet: 2006-24064

Br. Harold Alejandro
Solís Contreras
Carnet: 97-12911-3

Tutor:

M.Sc. Danilo Humberto
Noguera Rivera

14 de octubre de 2023
Managua, Nicaragua

RESUMEN

En este documento se pueden observar las etapas de desarrollo del sistema de información para la generación de nómina de la empresa “Distribuidora General S.A.” (DIGESA) utilizando la metodología UWE.

Para el desarrollo de esta propuesta fue necesario iniciar con el modelado del negocio para identificar la organización de la institución, los recursos con los que cuenta y los actores de los procesos que fue fundamental para elaboración del proyecto. También se determinó los requerimientos funcionales y no funcionales necesarios, en base a estos se estableció un diseño conceptual a partir de un análisis detallado en los elementos solicitados permitiendo lograr los objetivos propuestos.

Para la construcción del sistema de entorno web se utilizaron las bases del diseño planteado con las tecnologías de desarrollo Asp.Net MVC, lenguaje C# y como gestor de base de datos Microsoft SQL Server.

La implementación del sistema de información para realización de la nómina quincenal, selección, contratación y gestión del personal como también la generación de reportes administrativos permite a la empresa reducir los tiempos de ejecución de cada proceso involucrado, incrementando la transparencia activa y disponibilidad de información para la toma de decisiones; así mismo contribuye en una mejor atención a los usuarios internos y un debido control del departamento de gestión humana.

ÍNDICE

DEDICATORIA	v
AGRADECIMIENTOS.....	vi
I INTRODUCCIÓN	1
II PLANTEAMIENTO DE LA SITUACIÓN PROBLÉMICA	2
III ANTECEDENTES.....	4
IV OBJETIVOS	6
a. OBJETIVO GENERAL.....	6
b. OBJETIVOS ESPECÍFICOS.....	6
V JUSTIFICACIÓN	7
VI MARCO TEÓRICO	8
CAPÍTULO I. ANÁLISIS DE REQUERIMIENTOS.....	38
1.1. GIRO DEL NEGOCIO.....	38
1.1.1. MISIÓN.....	38
1.1.2. VISIÓN	38
1.1.3. VALORES	39
1.1.4. ORGANIGRAMA DE LA EMPRESA	39
1.1.5. NORMATIVAS.....	40
1.2. DESCRIPCIÓN DE LOS PROCESOS DE NEGOCIO.....	42
1.2.1. PROCESO DE SELECCIÓN Y CONTRATACIÓN.....	42
1.2.2. PROCESO DE GENERACIÓN DE NÓMINA	45
1.3. DIAGRAMAS BPMN	48
1.4. OBJETIVOS DEL SISTEMA	50
1.5. ALCANCE DEL SISTEMA	51
1.6. BENEFICIOS DEL SISTEMA	51
1.7. REQUERIMIENTOS FUNCIONALES.....	56
1.8. REQUERIMIENTOS NO FUNCIONALES	59
1.9. TABULACIÓN DE LOS REQUERIMIENTOS	60
1.10. MATRIZ DE TRAZABILIDAD.....	63
CAPÍTULO II DISEÑO DEL SISTEMA	64
2.1. DISEÑO DE ARQUITECTURA DEL SISTEMA.....	64

2.2.	PATRÓN DE DISEÑO DE LA ARQUITECTURA	65
2.3.	MODELOS UWE.....	66
2.4.	DISEÑO FÍSICO DE LA BASE DE DATOS	76
CAPÍTULO III – CONSTRUCCIÓN DEL SISTEMA.....		77
3.1.	SELECCIÓN DEL HARDWARE PARA LA CONSTRUCCIÓN Y ALOJAMIENTO DEL SISTEMA	77
3.2.	SELECCIÓN DEL SOFTWARE PARA LA CONSTRUCCIÓN	79
3.3.	ESPECIFICACIÓN DE LAS CLASES DEL SISTEMA	80
3.4.	IMPLEMENTACIÓN DE PATRONES DE DISEÑO	93
3.5.	ORGANIZACIÓN DE LOS ARCHIVOS DEL PROYECTO.....	96
3.6.	DISEÑO DE LA INTERFAZ DEL SISTEMA.....	97
VII CONCLUSIONES		105
VIII RECOMENDACIONES.....		106
IX BIBLIOGRAFÍA		107
X ANEXOS.....		110

ÍNDICE DE FIGURAS

FIGURA 1: EL PROCESO DE ADQUISICIÓN Y ANÁLISIS DE REQUERIMIENTOS	
.....	23
FIGURA 2: UWE MODELO DE REQUERIMIENTOS:	30
FIGURA 3: MODELO DE CONTENIDO	30
FIGURA 4: MODELO DE NAVEGACIÓN	31
FIGURA 5: MODELO DE PRESENTACIÓN	31

ÍNDICE DE ILUSTRACIONES

ILUSTRACIÓN 1 : PROCESO DE SELECCIÓN Y CONTRATACIÓN DEL PERSONAL.	48
ILUSTRACIÓN 2 : PROCESO DE GENERACIÓN DE NÓMINA.	49
<i>ILUSTRACIÓN 3 : MATRIZ DE TRAZABILIDAD.</i>	63
ILUSTRACIÓN 4 : DIAGRAMA DE CASOS DE USO - REGISTRAR EMPLEADO.	
.....	67
ILUSTRACIÓN 5 : DIAGRAMA DE CASOS DE USO - REGISTRAR COMPENSACIONES Y DEDUCCIONES DEL EMPLEADO.	68
ILUSTRACIÓN 6 : DIAGRAMA DE CONTENIDO.	69
ILUSTRACIÓN 7 : DIAGRAMA DE NAVEGACIÓN - RESPONSABLE IT.	70
ILUSTRACIÓN 8 : DIAGRAMA DE NAVEGACIÓN - ANALISTA DE NÓMINA.	71
ILUSTRACIÓN 9 : DIAGRAMA DE NAVEGACIÓN - ANALISTA DE RECLUTAMIENTO Y SELECCIÓN.	72
ILUSTRACIÓN 10 : DIAGRAMA DE PROCESO - GENERACIÓN DE NÓMINA.	73
ILUSTRACIÓN 11 : DIAGRAMA DE ACTIVIDAD - PROCESAR NÓMINA.	74
ILUSTRACIÓN 12 : DIAGRAMA DE PRESENTACIÓN - PERÍODO NÓMINA.	75
ILUSTRACIÓN 13 : MODELO RELACIONAL.	76

ÍNDICE DE TABLAS

TABLA 1 : MATRIZ DE TRAZABILIDAD	28
TABLA 2 : IMPUESTO SOBRE LA RENTA (IR).....	47
TABLA 3 : ESPECIFICACIONES DE LAPTOP UTILIZADAS.	77
TABLA 4 : ESTRUCTURA DEL MODELO.....	93
TABLA 5 : ESTRUCTURA DE LAS VISTAS.	94
TABLA 6 : ESTRUCTURA DEL CONTROLADOR.	95
TABLA 7 : ORGANIZACIÓN DE ARCHIVOS DEL PROYECTO.	96

ÍNDICE DE ANEXOS

ANEXO 1 : ESTRUCTURA ORGANIZACIONAL DE LA EMPRESA.....	110
ANEXO 2 : TABULACIÓN DE REQUERIMIENTOS.	111
ANEXO 3 : DIAGRAMAS DE CASOS DE USO.	124
ANEXO 4 : DIAGRAMA DE CONTENIDO.	134
ANEXO 5 : MODELO DE PROCESO.....	136
ANEXO 6 : DIAGRAMAS DE ACTIVIDAD.....	147
ANEXO 7 : DIAGRAMA DE PRESENTACIÓN.	149
ANEXO 8 : MODELO RELACIONAL.	150

DEDICATORIA

Dedicamos este logro primeramente a Dios, que nos ha dado el conocimiento y privilegio de ser Ingenieros de Sistemas, por las fuerzas y recursos que nos distes para sobre pasar cada obstáculo del camino.

A nuestras familias, hijos, esposas y padres quienes nos apoyaron incondicionalmente con su paciencia mientras se completaba este proceso de titulación.

AGRADECIMIENTOS

Al cuerpo directivo de nuestra facultad que nos apoyaron para lograr culminar esta meta en tan prestigiosa universidad y en especial al MSc. Danilo Humberto Noguera Rivera quien nos guio en el proceso, aportando de sus conocimientos y consejos.

Finalmente, queremos agradecer a la empresa DIGESA por la atención e información brindada a lo largo de este proceso.

I INTRODUCCIÓN

Una de las complicaciones que surgen en empresas relativamente nuevas con un crecimiento acelerado es la elaboración de su nómina, la cual a medida que pasa el tiempo y no teniendo un sistema de información se hace una tarea tediosa dado que se maneja una gran cantidad de registros de los empleados.

El proceso de elaboración y control de la nómina de una empresa, es un proceso sensible que requiere de una gran precisión en sus cálculos y de un seguimiento minucioso, debido a esto es necesario implementar un sistema de información que automatice el proceso y así se evita pérdidas monetarias, desgaste de parte de los encargados de la actividad, errores constantes e inconformidad con el activo más importante que tiene las empresas como son sus colaboradores.

Este trabajo monográfico se enfocó en la generación quincenal de la nómina para el departamento de Gestión Humana de la empresa “Distribuidora General S.A.” (DIGESA), ubicada en Managua, en la cuarta entrada a las colinas, 300 metros al oeste, quien es distribuidor autorizado de la empresa de telecomunicaciones TIGO; proceso que actualmente se generaba de manera manual con archivos en Excel.

Para la creación del sistema de información que genera la nómina de la empresa DIGESA, se determinó los requerimientos funcionales y no funcionales necesarios mediante un estudio de proceso del negocio. También se realizó el diseño del sistema de información según los requerimientos obtenidos, utilizando la metodología UWE. Para la construcción del sistema de entorno web se utilizaron las bases del diseño planteado con las tecnologías de desarrollo Asp.Net MVC, lenguaje C# y como gestor de base de datos Microsoft SQL Server.

II PLANTEAMIENTO DE LA SITUACIÓN PROBLÉMICA

La generación de la nómina en tiempo y forma es de vital importancia, ya que se lleva el control de las prestaciones laborales de cada colaborador, uno o varios retrasos en dicho proceso conllevan a descontento o inconformidad en los colaboradores.

Actualmente la empresa se apoya de una plantilla en un archivo de Excel, el área de ventas entrega el reporte de comisiones, viáticos de alimentación y transporte, vacaciones y reporte de movimiento de personal.

Desde el área de gestión humana ingresan cada uno de los registros en la plantilla antes mencionada, así como otras compensaciones y deducciones de las otras áreas. Generar la nómina puede tomar aproximadamente de 16 a 20 horas laborales quincenalmente.

La gerencia ha expresado preocupación por el tiempo considerable que le toma a dicha área en elaborar y generar la nómina, así como también los incidentes presentados por varias causas, tales como:

Tiempo excesivo para la modificación del archivo Excel en cada periodo quincenal, ya que el analista de reclutamiento y selección, así como los analistas de nómina deben de reordenar la información en la hoja de Excel.

Al no dar de baja a un colaborador en tiempo y forma, esto ha ocasionado pérdidas monetarias ya que se han presentado eventualidades en determinados casos, en los cuales se deposita el pago a un colaborador que ya no labora para la empresa.

La entrega tardía del reporte de las comisiones de ventas y los errores en las referencias de las fórmulas, así como la duplicidad de información, conllevan a un mayor esfuerzo físico y mental por parte de los analistas de nómina. también registrar las compensaciones y deducciones en la hoja de Excel y demasiados pasos para

realizar determinados cálculos, requieren de más tiempo para el registro de los datos de los colaboradores de la empresa.

Las causas antes mencionadas conllevan a elaborar una nómina con considerables errores en los cálculos y en requerir más tiempo de lo previsto para la revisión de la misma.

III ANTECEDENTES

La empresa Distribuidora General S.A. surgió como un emprendimiento fundado en el año 2000. En el año 2005 incursiona en el mercado nicaragüense como distribuidor autorizado de la marca Movistar de la empresa internacional de telecomunicaciones Telefónica S.A. Con ello se expandió en diferentes zonas con su principal rubro de recarga electrónicas en la zona de sur oriente, centro norte y el caribe.

En el año 2015 hasta la actualidad comenzó a incursionar el rubro de planes pospago, televisión por cable e internet con lo cual ha crecido su personal teniendo presencia en los departamentos de Masaya, Granada, Carazo, Rivas, Boaco, Río Blanco y Siuna.

Debido al crecimiento del personal, el departamento de gestión humana ha pasado por un proceso de estructuración, dejando tres cargos los cuales son: Gerencia de Gestión Humana, Analista de Reclutamiento y Selección, Analista de Nómina, en este último se adicionó una persona más para apoyar al procesamiento de la nómina, ya que anteriormente solamente había una persona, la nómina siempre se ha trabajado en el producto de ofimática Microsoft Excel.

Este archivo cuenta con una plantilla predefinida, la cual se actualiza en cada periodo quincenal, donde se agregan pestañas con nombres clave para identificar los datos necesarios para el cálculo y la generación de la nómina, como los siguientes:

- Información de los colaboradores.
- Una hoja para el cálculo de las compensaciones y deducciones.
- Una hoja para los datos de las cuentas bancarias de los colaboradores.

Dicho archivo se guarda en carpetas que llevan la estructura de año y mes, en la cual se almacena el archivo de Excel que contiene la nómina generada quincenalmente.

En el año 2010 la empresa adquirió un sistema ERP, el cual tenía los siguientes módulos: contabilidad, finanzas y ventas, pero el módulo de nómina no fue implementado, ya que no cumplía con las necesidades en su momento y su funcionalidad incrementaba el costo de implementación.

En la parte de revisión documental no se encontraron temas monográficos con relación a esta misma empresa con su problemática.

IV OBJETIVOS

a. OBJETIVO GENERAL

Desarrollar un sistema de información que automatice la generación de nómina de la empresa “Distribuidora General S.A.” (DIGESA).

b. OBJETIVOS ESPECÍFICOS

- Determinar los requerimientos necesarios para el desarrollo del sistema de información mediante un estudio del proceso del negocio.
- Realizar el diseño del sistema de generación de nómina, según los requerimientos y utilizando la metodología UWE a fin de documentar los procesos que reduzcan la generación de nómina.
- Construir el sistema de información web utilizando las bases del diseño planteado, con las tecnologías de desarrollo Asp.Net MVC, el lenguaje de programación C# y con el gestor de base de datos Microsoft SQL Server.

V JUSTIFICACIÓN

El crecimiento de la empresa DIGESA, ha conllevado a un incremento de la fuerza laboral por lo que es necesario que la nómina sea procesada eficientemente, manteniendo la integridad, confidencialidad y disponibilidad de la información en todo momento.

El presente proyecto tiene como propósito el desarrollo de un sistema de información que permita automatizar el proceso de generación de nómina para la empresa DIGESA.

Con dicho desarrollo se desea reducir los tiempos en el procesamiento de la nómina realizando menos pasos en el registro de compensaciones y deducciones de los colaboradores de la empresa, reducir el margen de error al automatizar los cálculos de la nómina, llevar un mejor control de las compensaciones y deducciones, reportes de vacaciones, reporte de altas y bajas de los colaboradores de la empresa.

También reducir el esfuerzo físico y mental de los analistas de nómina en la elaboración de la misma, y generar los informes que sean de apoyo en tiempo y forma para el análisis y toma de decisiones de la Gerencia General, Gerencia Financiera y Gerencia de Recursos Humanos.

El desarrollo del software beneficiará a la empresa al disminuir los costos relacionados con el pago de horas extras, al reducir el tiempo y esfuerzo en la elaboración de la nómina, mantener la integridad, disponibilidad y confidencialidad de la información al disponer un sistema automatizado.

VI MARCO TEÓRICO

El presente documento tiene como finalidad dar a conocer el trabajo de estudio a realizar, para determinar los requerimientos, funcionales como no funcionales del desarrollo del sistema de información para la automatización de la generación de la nómina en la empresa DIGESA.

SISTEMA.

Según (Souza & Oz, 2016, pág. 10), “Sistemas es una matriz de componentes que colaboran para alcanzar una meta en común, o varias, al aceptar entradas, procesarlas y producir salidas de una manera organizada”.

Subsistema: Según (Souza & Oz, 2016, pág. 10), “a menudo un sistema está formado por varios subsistemas -componentes de uno mayor- con metas secundarias las cuales contribuyen a alcanzar la meta principal. Los subsistemas pueden recibir entradas a otros sistemas o subsistemas y transferir salidas de otros sistemas o subsistemas”.

El departamento de Gestión Humana debe garantizar la contratación, capacitación, normas y políticas que requiere cada colaborador, así como también el pago de prestaciones con base en la ley de constitución política de Nicaragua, se puede decir que dicho departamento es un subsistema del sistema principal, que debe de cumplir con una o varias metas secundarias de un sistema mayor.

SISTEMA DE INFORMACIÓN (SI)

Según (Stair & Reynolds, 2015, pág. 10), “partiendo de la definición del sistema, un sistema de información es un conjunto de elementos interrelacionados que recaban, procesan, almacenan y distribuyen datos e información, y, además, proporcionan mecanismos de retroalimentación para alcanzar un objetivo”.

Dado al importante papel que los sistemas de información tienen asignado, un sistema de información debe de complementarse de sistemas informáticos, para la empresa DIGESA, es fundamental contar un sistema de información que automatice la generación de la nómina.

6.1. CATEGORÍAS DE SISTEMAS DE INFORMACIÓN

Según (Souza & Oz, 2016, pág. 55), “los sistemas de información tienen por objetivo principal suministrar a los distintos niveles de la empresa la información necesaria para la toma de decisiones”. Dada la complejidad de los procesos se hace necesario distintas categorías que se mencionan a continuación:

Nivel Estratégico: Los sistemas a este nivel tienen como propósito fundamental ser de apoyo para establecer estrategias a largo plazo, obteniendo información de varias fuentes internas y externas a la empresa.

Nivel Táctico: Los sistemas de información a este nivel deben permitir a los directivos establecer objetivos y tomar decisiones, planificar la asignación de recursos, así como medir y controlar el resultado de las acciones emprendidas.

Nivel Operativo: En este nivel se ubican los sistemas de información que apoyan decisiones estructuradas y programables.

El software a desarrollar por ser un sistema orientado al procesamiento de la nómina se considera como un sistema de información de procesamiento de datos (EDP), el cual se ubica en el nivel operativo, y tratan de mejorar la eficiencia operativa automatizando los flujos de información y procesos.

6.2. SISTEMA DE INFORMACIÓN WEB

Los sistemas de información web forman parte de la informática. Además, es una herramienta útil que permite mejorar la experiencia del usuario. Según (Senso., 2016, pág. 1).

“Los sistemas de información han ido evolucionando a lo largo de los años empleando diferentes programas y entornos hasta encontrar, en la Web, el hábitat ideal. La World Wide Web emplea un conjunto de recomendaciones, técnicas y tecnologías muy robustas que ofrecen un gran rendimiento y fiabilidad, así como un mecanismo de navegación muy familiar para la mayoría de los usuarios”.

Las aplicaciones web suelen estar desarrolladas con HTML, JavaScript, CSS, entre otras tecnologías. Es recomendable tener una web optimizada y responsive. De ese modo todos los usuarios podrán navegar, con independencia del sistema operativo que utilicen.

6.3. MODELADO DE NEGOCIO

El modelado del negocio forma parte del análisis del sistema, como paso previo al diseño y construcción del sistema de información propuesto, (Sommerville, 2011, pág. 39). La mayoría del software tiene interfaz junto con otros sistemas de software.

En ellos se incluyen sistema operativo, base de datos, middleware y otros sistemas de aplicación. Éstos constituyen la “plataforma de software”, es decir, el entorno donde se ejecutará el software.

Para ello se hace uso de un modelo lógico el cual es una representación gráfica del proceso del negocio.

Los modelos lógicos muestran lo que un sistema es o lo que hace. Son independientes de su implementación; es decir, bosquejan el sistema siendo independiente de cualquier implementación técnica.

Como tales, los modelos lógicos ilustran la esencia del sistema. Según (KENDALL & KENDALL, 2011, pág. 258), “los sinónimos populares incluyen modelo elemental, modelo conceptual y modelo de negocios”.

Los analistas de sistemas hace mucho tiempo han reconocido el valor de separar los aspectos de negocios y los técnicos.

(Whitten & Bentley, 2008, pág. 258) “por esto es que usan modelos de sistemas lógicos para definir los requisitos de negocios y modelos de sistemas físicos para esquematizar diseños técnicos, por consiguiente, dichos requisitos del negocio serán definidos por medio de la técnica de modelado de procesos”.

6.4. MODELADO DE PROCESO

El modelado de proceso es la captura de una secuencia de actividades de negocio y de la información de soporte. Dichos procesos de negocios describen como la empresa logra sus objetivos.

Un proceso es una concatenación lógica de actividades, a través del tiempo y lugar, impulsado por eventos y que, a través de su proceso de transformación, cumplen un determinado fin. (Freund & Rucker, 2014, pág. 12).

6.5. CLASIFICACIÓN DE MODELO DE PROCESO

Las metodologías basadas en técnicas de diagramación, las podemos clasificar en técnicas orientadas al flujo de datos, al flujo de control y orientadas al objeto. (Freund & Rucker, 2014, pág. 123).

Cada grupo está compuesto por diferentes técnicas, la técnica a utilizar para el modelado del negocio en el presente proyecto monográfico será la técnica orientada al flujo de control y más específicamente trabajar con la metodología BPMN que se describe a continuación.

MODELO Y NOTACIÓN DE PROCESOS DE NEGOCIOS (BPMN)

Business Process Management and Notation (BPMN), que en español significa Modelo y Notación de procesos de Negocio, es un enfoque sistemático para identificar, levantar, documentar, diseñar, ejecutar, medir y controlar tanto los procesos manuales como automatizados, con la finalidad de lograr a través de sus resultados en forma consistente los objetivos de negocio que se encuentran alineados con la estrategia de la organización. (Freund & Rucker, 2014, pág. 12).

BPM abarca el apoyo creciente de TI con el objetivo de mejorar, innovar y gestionar los procesos de principio a fin, que determinan los resultados de negocio, crean valor para el cliente y posibilitan el logro de los objetivos de negocio con mayor agilidad.

De forma resumida, es un estándar para el modelado de procesos de negocios y servicios web, es una notación a través de la cual se expresan los procesos de negocios en un diagrama de procesos de negocio (BPD).

ELEMENTOS DE LOS DIAGRAMAS

Actividades: Una actividad representa una acción de un proceso que sirve para denominar el trabajo que realiza un participante. (Standards Development organization (OMG), 2010).

También puede ser un conjunto de tareas, pero si son varias tareas entonces se denomina subproceso. El gráfico con el cual se representa una actividad es con un rectángulo, simple o compuesto.

Eventos: Un evento es algo que “sucede” durante el curso de un proceso. Estos eventos afectan el flujo del proceso y generalmente tienen una causa o un impacto. (Standards Development organization (OMG), 2010).

El término "evento" es lo suficientemente general como para cubrir muchas cosas en un proceso.

Sin embargo, BPMN ha restringido el uso de eventos para incluir solo aquellos tipos de eventos que afectarán la secuencia o el tiempo de las actividades de un proceso.

Estos se representan con un círculo y se dividen en tres tipos de eventos los cuales son:

- **Inicio:** marca el inicio del evento y se representa con un círculo con una línea de traza.
- **Intermedio:** muestran un estado que el proceso ha alcanzado y que en el modelo por alguna razón lo queremos retener, su uso no es muy frecuente, pero es útil en ocasiones, por ejemplo, si el estado representa un hito y se quiere medir el transcurrido hasta alcanzar el hito.
- **Fin:** son eventos finales los cuales indican que se logró finalizar una trayectoria del proceso.

Puerta de Enlace: Las puertas de enlace se utilizan para controlar cómo interactúan los flujos de secuencia cuando convergen y divergen dentro de un proceso. Si no es necesario controlar el flujo, no se necesita una puerta de enlace. (Standarts Development organization (OMG), 2010).

Una puerta de enlace es un rombo, que se ha utilizado en muchas notaciones de diagramas de flujo para bifurcaciones exclusivas y es familiar para la mayoría de los modeladores.

Flujos de Secuencia: El flujo de secuencia describe la secuencia temporal y lógica en el cual se combinan los elementos de flujo, es decir las actividades, eventos y Gateway.

El flujo de secuencia es también la trayectoria del proceso por el cual marcha el token. El token «nace» junto a una instancia con el evento de inicio.

A través del flujo de secuencia se llega a las actividades, a los estados intermedios y al evento final, donde es «consumido» y desaparece, al mismo tiempo «muere» nuestra instancia. (Standarts Development organization (OMG), 2010).

Se representa con una línea continua con una punta de flecha sólida que refleja el orden en que las actividades se llevan a cabo.

6.8.1. NOTACIONES

Como notación, BPMN especifica la representación de cada uno de los elementos BPMN.

6.8.2. OBJETOS DE FLUJO

Son elementos que están conectados y forman un flujo de proceso.

6.8.3. OBJETOS DE CONEXIÓN

Los objetos de conexión simbolizan como se conectan los objetos entre si y como representar cosas que fluyen mediante proceso.

Existen los siguientes objetos de conexión:

- Flujo de secuencia
- Flujo de mensaje
- Asociación

6.8.4. CARRILES

Los carriles o lanes son utilizados para la asignación de responsables de ejecutar determinada tarea o actividad. El pool representa una instancia invisible que dirige y controla el flujo de actividades en el proceso.

La especificación de BPMN no hace ningún tipo de clasificación para asignar personas a los carriles, por lo general en la práctica se utiliza para:

- Cargos de algún área (como gerente, supervisor, etc.).
- Roles (como auditor, usuario de negocio, ejecutivo, etc.).
- Roles generalizados (como cliente, proveedor, fiscalizador, etc.).
- Departamentos (como contabilidad, ventas, etc.).
- Aplicaciones o sistemas (como CRM, SAP, SOFTLAND, etc.).

6.8.5. CONTENEDOR

BPMN denomina contenedor o pools a los participantes del proceso descrito. Representa los límites de un proceso y siempre contiene dentro los carriles o lanes.

6.8.6. ARTEFACTOS

La notación de BPMN también permite introducir simbología propia como artefactos. Con esta posibilidad se pueden enriquecer los diagramas ilustrativos típicos de un negocio. (Standarts Development organization (OMG), 2010). Los artefactos clasifican las actividades y pueden emplearse para incluir más información sobre un proceso en el BPMN. Los más relevantes son los siguientes:

Objeto de datos: Los datos requeridos para un proceso. Se parece a una hoja de papel con la esquina superior derecha doblada.

Grupo: Consiste en un rectángulo con bordes redondeados con líneas discontinuas y agrupa diferentes actividades.

Anotación de texto: Es un texto que proporciona más información.

6.9. HERRAMIENTAS DE MODELADO

Hay variedad de herramientas diferentes para el modelado de negocios que se encuentran disponibles tanto como programas, así como en línea, los más relevantes son los siguientes:

- LucidChart url: <https://www.lucidchart.com/pages/>
- Bpmn.io url: <https://bpmn.io/>
- Flokzu url: <https://flokzu.com/es/>
- Bizagi url: <https://www.bizagi.com/es>
- BonitaSoft url: <https://es.bonitasoft.com/>
- Draw.io url: <https://app.diagrams.net/>
- Softwareideas.net url: <https://www.softwareideas.net/en/features>

La herramienta de modelado a utilizar para el modelado de proceso será draw.io, ya que esta herramienta es open source, online, cuenta con soporte de software para escritorio, y sobre todo es intuitiva para utilizar.

INGENIERÍA DE REQUERIMIENTOS

La especificación del software o la ingeniería de requerimientos consisten en el proceso de comprender y definir qué servicios se requieren del sistema, así como la identificación de las restricciones sobre la operación y el desarrollo del sistema. (Sommerville, 2011).

Un requerimiento es una característica que debe incluirse en un nuevo sistema y puede consistir en una forma de captar o procesar datos, producir información, controlar una actividad o dar apoyo a una tarea.

Los requerimientos del usuario y los requerimientos del sistema se definen del siguiente modo:

1. Los requerimientos del usuario son enunciados, en un lenguaje natural junto con diagramas, acerca de qué servicios esperan los usuarios del sistema, y de las restricciones con las cuales éste debe operar.
2. Los requerimientos del sistema son descripciones más detalladas de las funciones, los servicios y las restricciones operacionales del sistema de software. El documento de requerimientos del sistema (llamado en ocasiones especificación funcional) tiene que definir con exactitud lo que se implementará. Puede formar parte del contrato entre el comprador del sistema y los desarrolladores del software. (Sommerville, 2011, pág. 83).

DEFINIR EL ALCANCE DEL PROYECTO

Los requerimientos son esenciales para el alcance de un proyecto. Los requerimientos de software ayudan a determinar qué características estarán en el producto final, cuánto tiempo llevará desarrollar esas características y cuánto costará. Si el alcance de un proyecto no está bien definido, puede conducir a un deslizamiento del alcance.

El deslizamiento del alcance ocurre cuando el alcance de un proyecto comienza a expandirse más allá del acuerdo inicial. Puede suceder porque las partes interesadas comienzan a incluir una nueva característica o porque el equipo del proyecto no hizo un excelente trabajo al definir el alcance en primer lugar.

TIPOS DE REQUERIMIENTOS

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

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.

En principio, la especificación de los requerimientos funcionales de un sistema debe ser completa y consistente. Totalidad significa que deben definirse todos los servicios requeridos por el usuario.

Consistencia quiere decir que los requerimientos tienen que evitar definiciones contradictorias. En la práctica, para sistemas complejos grandes, es casi imposible lograr la consistencia y la totalidad de los requerimientos.

Una causa para ello es la facilidad con que se cometen errores y omisiones al escribir especificaciones para sistemas complejos. Otra es que hay muchos participantes en un sistema grande.

Un participante es un individuo o una función que se ve afectado de alguna forma por el sistema. Los participantes tienen diferentes necesidades, pero con frecuencia son inconsistentes.

Tales inconsistencias tal vez no sean evidentes cuando se especifican por primera vez los requerimientos, de modo que en la especificación se incluyen requerimientos inconsistentes. Los problemas suelen surgir sólo después de un análisis en profundidad o después de que se entregó el sistema al cliente. (Sommerville, 2011, pág. 86).

REQUERIMIENTOS NO FUNCIONALES

Son limitaciones sobre servicios o funciones que ofrece el sistema. Incluyen restricciones tanto de temporización y del proceso de desarrollo, como impuestas por los estándares. Los requerimientos no funcionales se suelen aplicar al sistema como un todo, más que a características o a servicios individuales del sistema. (Sommerville, 2011, pág. 84).

Los requerimientos son importantes en el desarrollo de software y aplicaciones porque proporcionan una hoja de ruta para el proyecto. Sin requerimientos, sería difícil saber qué se necesita construir, cómo debe funcionar y cómo probarlo. Los requerimientos también ayudan a garantizar que todas las partes interesadas estén en la misma página y tengan una comprensión clara de los objetivos del proyecto.

Los requerimientos no funcionales surgen a través de necesidades del usuario, debido a restricciones presupuestales, políticas de la organización, necesidad de interoperabilidad con otro software o sistemas de hardware, o factores externos como regulaciones de seguridad o legislación sobre privacidad. (Sommerville, 2011, pág. 87).

Un problema común con requerimientos no funcionales es que los usuarios o clientes con frecuencia proponen estos requerimientos como metas generales, como facilidad de uso, capacidad de que el sistema se recupere de fallas, o rapidez de respuesta al usuario.

Las metas establecen buenas intenciones; no obstante, ocasionan problemas a los desarrolladores del sistema, pues dejan espacio para la interpretación y la disputa posterior una vez que se entregue el sistema. (Sommerville, 2011, pág. 89).

Los requerimientos no funcionales por lo general se dividen en tres categorías: rendimiento, seguridad y calidad.

- **REQUERIMIENTOS DE RENDIMIENTO**

Los requerimientos de rendimiento suelen dividirse en dos categorías: tiempo de respuesta y rendimiento.

El tiempo de respuesta es el tiempo que tarda un sistema en responder a la solicitud de un usuario, mientras que el rendimiento es el número de solicitudes que un sistema puede manejar.

Son más críticos para los sistemas interactivos, como las aplicaciones de escritorio y los sitios web, donde los usuarios esperan respuestas inmediatas a sus acciones.

- **REQUERIMIENTOS DE SEGURIDAD**

Los requerimientos de seguridad especifican las medidas que un sistema debe tomar para proteger los datos del acceso no autorizado.

En algunos casos, los requerimientos de seguridad también pueden especificar el nivel de protección requerido, como confidencial o de alto secreto. Implica autenticación, autorización y cifrado.

- **REQUERIMIENTOS DE CALIDAD**

Especifica el nivel de calidad que debe cumplir un sistema. En algunos casos, los requerimientos de calidad también pueden especificar los métodos utilizados para medir la calidad, como la densidad de defectos o la satisfacción del cliente.

Los requerimientos de calidad son generalmente cuatro medidas de calidad: conformidad, usabilidad, confiabilidad y mantenibilidad.

Los requerimientos no funcionales conforman restricciones de los servicios o funciones ofrecidos por el sistema. Incluyen restricciones de tiempo, sobre el proceso de desarrollo, estándares, etc.

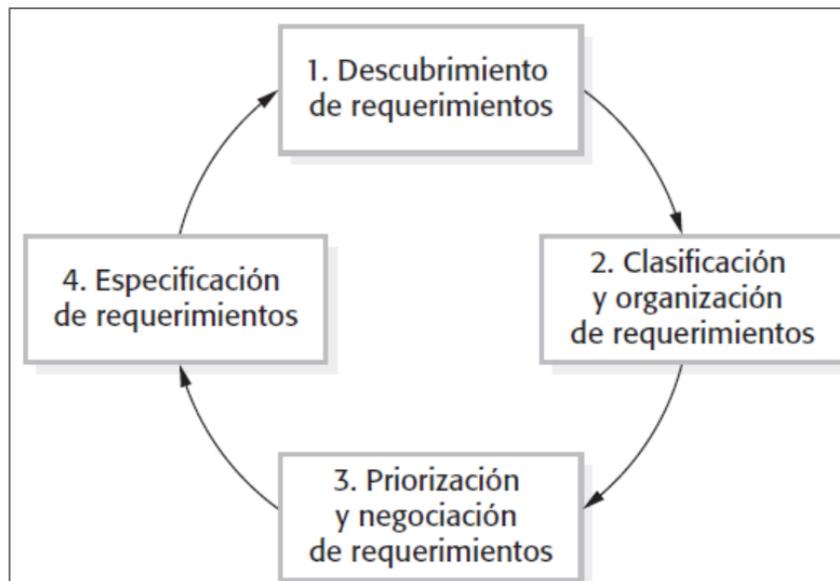
Los requerimientos son importantes en el desarrollo de software y aplicaciones porque proporcionan una hoja de ruta para el proyecto. Sin requerimientos, sería difícil saber qué se necesita construir, cómo debe funcionar y cómo probarlo.

Los requerimientos también ayudan a garantizar que todas las partes interesadas estén en la misma página y tengan una comprensión clara de los objetivos del proyecto.

ADQUISICIÓN Y ANÁLISIS DE LOS REQUERIMIENTOS

En esta actividad, los ingenieros de software trabajan con clientes y usuarios finales del sistema para descubrir el dominio de aplicación, qué servicios debe proporcionar el sistema, el desempeño requerido de éste, las restricciones de hardware, etcétera. (Sommerville, 2011, pág. 100). Ver Figura 1.

Figura 1: El proceso de adquisición y análisis de requerimientos



Fuente: Tomado de (Sommerville, 2011, pág. 101).

En una organización, la adquisición y el análisis de requerimientos pueden involucrar diversas clases de personas. Un participante en el sistema es quien debe tener alguna influencia directa o indirecta sobre los requerimientos de este. Los participantes incluyen a usuarios finales que interactuarán con el sistema, y a cualquiera en una organización que resultará afectada por el.

Otros participantes del sistema pueden ser los ingenieros que desarrollan o mantienen otros sistemas relacionados, administradores de negocios, expertos de dominio y representantes de asociaciones sindicales.

En la figura 1 se muestra un modelo del proceso de adquisición y análisis. Cada organización tendrá su versión o ejemplificación de este modelo general, dependiendo de factores locales, tales como experiencia del personal, tipo de sistema a desarrollar, estándares usados, etcétera.

Los participantes varían desde administradores y usuarios finales de un sistema hasta participantes externos como los reguladores, quienes certifican la aceptabilidad del sistema. (Sommerville, 2011, p. 103).

ENTREVISTAS

Las entrevistas formales o informales con participantes del sistema son una parte de la mayoría de los procesos de ingeniería de requerimientos.

En estas entrevistas, el equipo de ingeniería de requerimientos formula preguntas a los participantes sobre el sistema que actualmente usan y el sistema que se va a desarrollar. Los requerimientos se derivan de las respuestas a dichas preguntas.

Contiene preguntas de tipo:

- 1. Cerradas:** Donde los participantes responden a un conjunto de preguntas preestablecidas.
- 2. Abiertas:** En las cuales no hay agenda predefinida. El equipo de ingeniería de requerimientos explora un rango de conflictos con los participantes del sistema y, como resultado, desarrolla una mejor comprensión de sus necesidades.

En la práctica, las entrevistas con los participantes son por lo general una combinación de ambas. Quizá se deba obtener la respuesta a ciertas preguntas, pero eso a menudo conduce a otros temas que se discuten en una forma menos estructurada.

Rara vez funcionan bien las discusiones completamente abiertas. Con frecuencia debe plantear algunas preguntas para comenzar y mantener la entrevista enfocada en el sistema que se va a desarrollar.

Las entrevistas son valiosas para lograr una comprensión global sobre qué hacen los participantes, cómo pueden interactuar con el nuevo sistema y las dificultades que enfrentan con los sistemas actuales.

A las personas les gusta hablar acerca de sus trabajos, así que por lo general están muy dispuestas a participar en entrevistas. Sin embargo, las entrevistas no son tan útiles para comprender los requerimientos desde el dominio de la aplicación. (Sommerville, 2011, pág. 104).

CASOS DE USO

Los casos de uso son una técnica de descubrimiento de requerimientos que se introdujo por primera vez en el método Objectory (Jacobson et al., 1993). Ahora se ha convertido en una característica fundamental del modelado de lenguaje unificado.

En su forma más sencilla, un caso de uso identifica a los actores implicados en una interacción y nombra el tipo de interacción. Entonces, esto se complementa con información adicional que describe la interacción con el sistema. La información adicional puede ser una descripción textual, o bien, uno o más modelos gráficos como una secuencia UML o un gráfico de estado.

Los casos de uso se documentan con el empleo de un diagrama de caso de uso de alto nivel. El conjunto de casos de uso representa todas las interacciones posibles que se describirán en los requerimientos del sistema.

Los actores en el proceso, que pueden ser individuos u otros sistemas, se representan como figuras sencillas. Cada clase de interacción se constituye como una elipse con etiqueta que vincula a los actores con la interacción.

De manera opcional, se agregan puntas de flecha a las líneas para mostrar cómo se inicia la interacción. (Sommerville, 2011, pág. 107).

En síntesis, un caso de uso es una descripción de las acciones de un sistema desde el punto de vista del usuario. Es una herramienta valiosa dado que es una técnica de aciertos y errores para obtener los requerimientos del sistema, justamente desde el punto de vista del usuario. Los diagramas de caso de uso modelan la funcionalidad del sistema usando actores y casos de uso. Los casos de uso son servicios o funciones provistas por el sistema para sus usuarios.

MODELADO DEL SISTEMA

El modelado de sistemas es el proceso para desarrollar modelos abstractos de un sistema, donde cada modelo presenta una visión o perspectiva diferente de dicho sistema. En general, el modelado de sistemas se ha convertido en un medio para representar el sistema usando algún tipo de notación gráfica, que ahora casi siempre se basa en notaciones en el Lenguaje de Modelado Unificado (UML). Sin embargo, También es posible desarrollar modelos formales (matemáticos) de un sistema, generalmente como una especificación detallada del sistema.

Los modelos se usan durante el proceso de ingeniería de requerimientos para ayudar a derivar los requerimientos de un sistema, durante el proceso de diseño para describir el sistema a los ingenieros que implementan el sistema, y después de la Implementación para documentar la estructura y la operación del sistema. Es posible desarrollar modelos tanto del sistema existente como del sistema a diseñar:

1. Los modelos del sistema existente se usan durante la Ingeniería de Requerimientos. Ayudan a aclarar lo que hace el sistema existente y pueden utilizarse como base para discutir sus fortalezas y debilidades. Posteriormente, conducen a los requerimientos para el nuevo sistema.
2. Los modelos del sistema nuevo se emplean durante la Ingeniería de Requerimientos para ayudar a explicar los requerimientos propuestos a otros participantes del sistema.

Los ingenieros usan tales modelos para discutir las propuestas de diseño y documentar el sistema para la implementación. En un proceso de ingeniería dirigido por modelo, es posible generar una implementación de sistema completa o parcial a partir del modelo del sistema.

MATRIZ DE TRAZABILIDAD

La trazabilidad de requisitos designa la capacidad de rastrear la estructura de relaciones entre los requisitos, a modo de poder comprender los impactos que el cambio en un requisito específico puede provocar en el desarrollo de un sistema (Arias, 2015, pág. 108). Para gestionar la trazabilidad, en general se utiliza la matriz de trazabilidad, en este tipo de matriz, se cruzan los requisitos entre sí y se verifica aquellos que interfieren entre sí.

También es importante hacer mención que es posible cruzar los objetivos del sistema contra los requisitos funcionales, siendo estos últimos aplicados en el presente proyecto. Con el fin de valorar que requerimiento funcional afecta o aporta al desarrollo de los objetivos del sistema propuesto.

El modelo de matriz de trazabilidad posee la siguiente forma:

Tabla 1: Matriz de trazabilidad

TRM-0001	OBJ-0001	OBJ-0002	OBJ-0003	OBJ-0004	OBJ-0005	OBJ-0006	OBJ-0007	OBJ-0008
FRQ-0001	-	-	-	↑	-	-	-	-
FRQ-0002	↑	-	-	↑	-	-	-	↑
FRQ-0003	↑	↑	↑	-	↑	-	-	↑
FRQ-0004	↑	-	↑	-	-	-	-	-
FRQ-0005	-	-	-	-	↑	-	-	-
FRQ-0006	-	-	↑	-	↑	-	-	-
FRQ-0007	-	-	↑	-	-	-	-	↑
FRQ-0008	-	-	↑	-	-	-	-	-
FRQ-0009	-	-	↑	-	-	-	-	-
FRQ-0010	-	-	↑	-	-	-	-	-
FRQ-0011	-	-	↑	-	↑	-	-	-
FRQ-0012	-	-	↑	-	-	-	-	-
FRQ-0013	-	-	↑	-	-	-	-	-
FRQ-0014	↑	↑	↑	-	↑	↑	↑	-
FRQ-0015	-	↑	-	-	-	-	-	-
FRQ-0016	-	↑	-	-	-	-	-	-
FRQ-0017	-	-	↑	-	-	-	-	-
FRQ-0018	-	↑	-	-	-	-	-	-
FRQ-0019	↑	↑	↑	-	↑	↑	↑	-
FRQ-0020	↑	-	-	-	-	-	-	-
FRQ-0021	↑	-	-	-	-	-	-	↑
FRQ-0022	-	-	↑	-	-	-	-	-
FRQ-0023	-	-	-	-	-	-	-	↑

Fuente: Elaboración propia.

La columna vertical de requisitos representa los elementos activos de las relaciones y la columna horizontal representa los elementos pasivos de la relación. Estos datos utilizados en la matriz de trazabilidad fueron obtenidos de los requerimientos funcionales de la empresa DIGESA y de los objetivos planteados del sistema en este trabajo monográfico.

METODOLOGÍA DE DESARROLLO

La metodología para el desarrollo de un servicio web es un conjunto de actividades llevadas a cabo para hacer más eficiente el desarrollo y aplicación de un servicio web ya que indica cómo hay que obtener los distintos productos parciales y finales.

METODOLOGÍA UWE

Según (Nieves-Guerrero y otros, 2014, pág. 137):

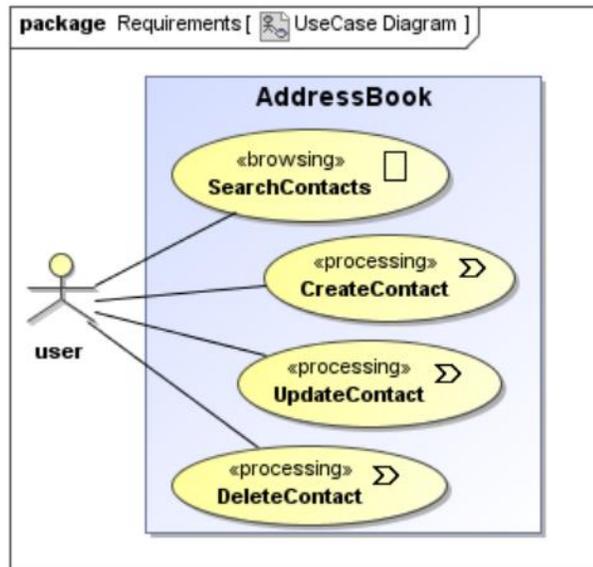
La Metodología UWE (Ingeniería Web basada en UML) permite especificar de mejor manera una aplicación web en su proceso de creación, mantiene una notación estándar basada en el uso de UML para sus modelos y sus métodos. La metodología define claramente la construcción de cada uno de los elementos del modelo.

La combinación de la ingeniería web y el lenguaje unificado de modelado (UML) da como resultado la UWE basada en UML. (Gutiérrez González & López Goytia, 2017, pág. 99).

El modelo que propone UWE está compuesto por etapas, las cuales son:

- **Modelo de casos de uso:** Modelo para capturar los requisitos del sistema. especifica los requisitos funcionales para definir los casos de uso (Gutiérrez González & López Goytia, 2017, pág. 99). Ver Figura 2.

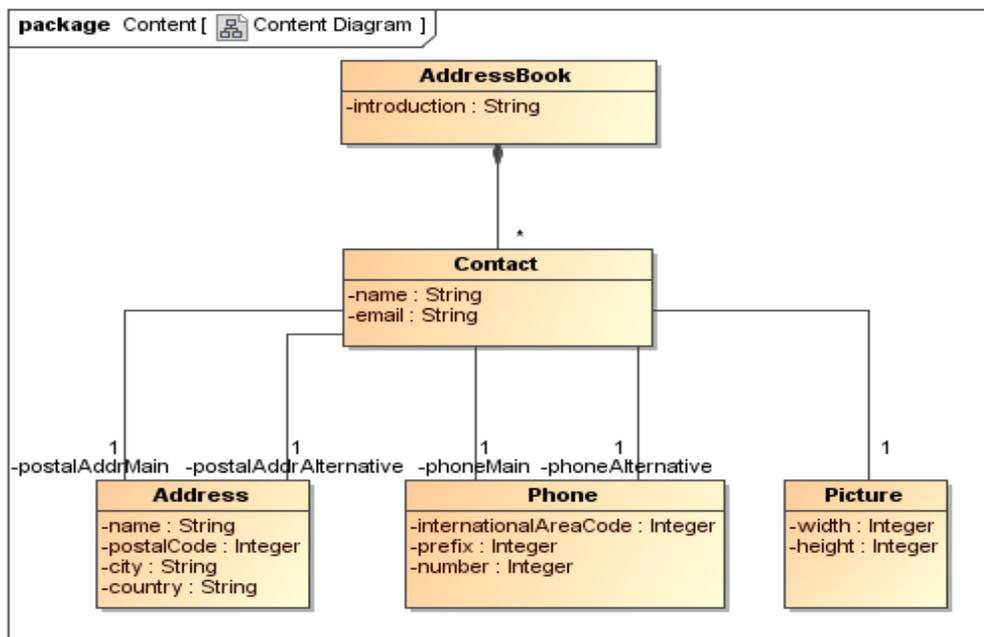
Figura 2: UWE modelo de requerimientos:



Fuente: (UWE – UML-based Web Engineering, 2016)

- 1. Modelo de contenido:** Es un modelo conceptual para el desarrollo del contenido. Ver Figura 3.

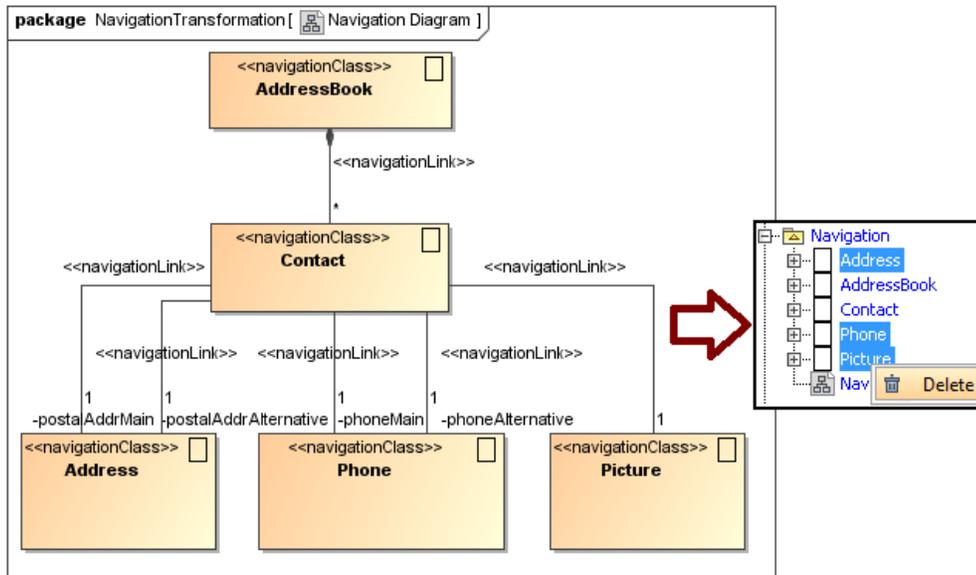
Figura 3: Modelo de contenido



Fuente: (Engineering, UWE – UML-based Web Engineering, 2016)

2. **Modelo de usuario o modelo de navegación:** Se incluyen modelos estáticos y modelos dinámico. Ver Figura 4.

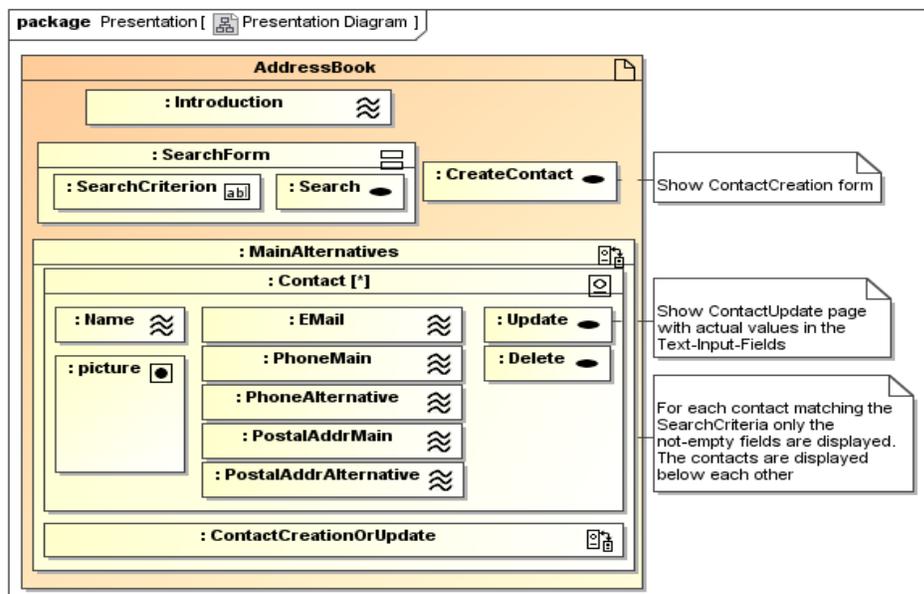
Figura 4: Modelo de navegación



Fuente: (Engineering, UWE – UML-based Web Engineering, 2016)

3. **Modelo de estructura o presentación:** En el cual se encuentra la presentación del sistema y el modelo de flujo. Ver Figura 5.

Figura 5: Modelo de presentación



Fuente: (Engineering, UWE – UML-based Web Engineering, 2016)

Modelo abstracto: Incluye el modelo de interfaz de usuario y el modelo de ciclo de vida del objeto.

- 4. Modelo de adaptación.** Refiere al término de personalización y conciencia del contexto.

LENGUAJE UNIFICADO DE MODELADO (UML)

Según (Edelmiro González Monsiváis, 2010, p. 1):

UML es una herramienta que ayuda a capturar mediante un conjunto de símbolos y diagramas a comunicar la idea de un sistema (software orientado a objetos), a quien esté involucrado en su proceso de desarrollo sirviendo de apoyo en los procesos de análisis y diseño de un problema.

Un diagrama ofrece una vista del sistema a modelar. UML ofrece varios diagramas para visualizar desde varias perspectivas el sistema (los diagramas más interesantes y más usados aparecen subrayados).

PATRONES DE DISEÑO

Describen soluciones adecuadas a problemas que ocurren repetidamente, haciendo al diseño más flexible, elegante y principalmente reusable.

Esta solución puede ser aplicada repetidamente para producir estructuras de diseño que lucen de forma similar al desarrollar aplicaciones diferentes.

Cada patrón se enfoca sobre un problema de diseño orientado a un objeto particular.

MODELO VISTA CONTROLADOR MVC

El patrón MVC es una arquitectura para crear aplicaciones que separan los diferentes aspectos de la aplicación (lógica de entrada, lógica comercial, lógica de versión y lógica de interfaz de usuario), al tiempo que proporciona un acoplamiento débil entre estos elementos.

El patrón especifica dónde debe ubicarse cada tipo de lógica en la aplicación.

La lógica de la interfaz de usuario pertenece a la vista. La lógica de entrada pertenece al controlador. La lógica empresarial pertenece al modelo. Esta separación lo ayuda a administrar la complejidad cuando crea una aplicación, ya que le permite trabajar en un aspecto de la implementación a la vez sin afectar el código de otro (Anderson, 2016, pág. 13). La versión de MVC utilizada para el desarrollo del sistema será la cinco.

Las tres secciones anteriormente mencionadas del patrón de diseño MVC, se definen de la siguiente manera.

- **Modelo:** Son clases que representan los datos de la aplicación y que usan la lógica de validación para hacer cumplir las reglas comerciales para esos datos. (Anderson, 2016, pág. 12), y que en termino de programación se refiere al manejo de datos y lógica de negocios.
- **Vista:** Las vistas son componentes que muestran una interface gráfica de usuario o GUI (Anderson, 2016, pág. 12). Es la encargada del diseño y presentación de la aplicación web.

- o **Controlador:** Son clases que manejan las solicitudes del navegador, recuperan datos del modelo y luego especifican plantillas de vista que devuelven una respuesta al navegador.

En una aplicación MVC, la vista solo muestra información; el controlador maneja y responde a la entrada e interacción del usuario (Anderson, 2016). Por ejemplo, el controlador maneja los datos de ruta y los valores de cadena de consulta y pasa estos valores al modelo. El modelo podría usar estos valores para consultar la base de datos.

Se utilizará en el desarrollo del proyecto el patrón antes mencionado, con el fin de separar la presentación del usuario y la lógica del negocio.

TECNOLOGÍAS DE DESARROLLO WEB

Los modelos y tecnologías de desarrollo web han evolucionado mucho en la última década, existen multitud de aplicaciones, frameworks, librerías, arquitecturas y sistemas de publicación en diferentes versiones que a su vez reciben mejoras con el tiempo.

El progreso también ha tenido lugar en lo relacionado con la administración de sistemas, servicios de alojamiento, técnicas de escalabilidad, monitorización y gestión de centros de procesos de datos.

TECNOLOGÍAS FRONT-END

Son aquellas que permiten crear interfaces de usuario y establecer comunicación con el servidor basadas en HTML, CSS y JavaScript, en este caso, el navegador actúa como intérprete.

HTML: (Lenguaje de Marcas de Hipertexto, del inglés HyperText Markup Language) es el componente más básico de la web. Define el significado y la estructura del contenido web.

Además de HTML, generalmente se utilizan otras tecnologías para describir la apariencia/presentación de una página web (CSS) o la funcionalidad/comportamiento (JavaScript) (html mozilla org, 2022). La versión actual es HTML5.

CSS: (Cascading Style Sheets o Hojas de estilo en cascada) son un mecanismo simple para agregar estilos como (colores, tipos de letra, fondos, efectos, etc.) en documentos HTML, XML, SVG o incluso interfaces de usuario de otras tecnologías. (CSS mozilla, 2023).

JavaScript: Es un lenguaje de programación ligero, interpretado o compilado con funciones de primera clase, basado en el lenguaje de scripting ECMAScript. (javascript mozilla, 2023) Se define como un lenguaje basado en prototipos, con soporte para programación orientada a objetos, imperativo, débilmente tipado y dinámico.

JQUERY: Es una biblioteca de JavaScript rápida, pequeña y rica en funciones. Hace que cosas como el recorrido y la manipulación de documentos HTML, el manejo de eventos, la animación y Ajax sean mucho más simples con una API fácil de usar que funciona en una multitud de navegadores. Con una combinación de versatilidad y extensibilidad (Jquery.com, 2023).

AJAX: (Asynchronous JavaScript +XML) no es una tecnología por sí misma, es un término que describe un nuevo modo de utilizar conjuntamente varias tecnologías existentes.

Esto incluye: HTML o XHTML, CSS, JavaScript, DOM, XML, XSLT, y lo más importante, el objeto XMLHttpRequest.

Cuando estas tecnologías se combinan en un modelo AJAX, es posible lograr aplicaciones web capaces de actualizarse continuamente sin tener que volver a cargar la página completa.

Esto crea aplicaciones más rápidas y con mejor respuesta a las acciones del usuario. (javascript Mozilla MDN, 2022) Estas aplicaciones se ejecutan en el cliente, es decir, en el navegador de los usuarios mientras se mantiene la comunicación asíncrona con el servidor en segundo plano.

DOM: (Document Object Model, en español Modelo de Objetos del Documento) es una API definida para representar e interactuar con cualquier documento HTML o XML.

El DOM es un modelo de documento que se carga en el navegador web y que representa el documento como un árbol de nodos, en donde cada nodo representa una parte del documento (puede tratarse de un elemento, una cadena de texto o un comentario) (DOM Mozilla, 2022).

En resumen, consta de una librería (API) para manipular el documento HTML cargado en el navegador, permitiendo la gestión de eventos, o la inserción y eliminación de elementos.

TECNOLOGÍAS BACK-END

ASP.NET MVC: Es un frameworks o marco de trabajo para compilar aplicaciones web y API mediante el patrón de diseño Modelo-Vista-Controlador (Anderson, 2016).

Basado en ASP.NET, permite a los desarrolladores de software construir una aplicación web como una composición de tres funciones: modelo, vista y controlador.

C#: Es un lenguaje de programación moderno orientado a objetos que permite desarrollar una amplia gama de aplicaciones para la nueva plataforma Microsoft .NET (Ceballos Sierra, 2014).

SQL: El lenguaje de consulta estructurada (SQL) es un lenguaje gestor para el manejo de la información en las bases de datos relacionales.

Este tipo de lenguaje de programación permite comunicarse con la base de datos y realizar operaciones de acceso y manipulación de la información almacenada (sql Universidad Europea, 2022).

Este lenguaje se considera el lenguaje estándar para las bases de datos según ANSI (Instituto Nacional Estadounidense de Estándares).

CAPÍTULO I. ANÁLISIS DE REQUERIMIENTOS

La especificación del software o la ingeniería de requerimientos consisten en el proceso de comprender y definir qué servicios se requieren del sistema, así como la identificación de las restricciones sobre la operación y el desarrollo del sistema (Sommerville, 2011).

1.1. GIRO DEL NEGOCIO

La empresa Distribuidora Generales S, A. es un distribuidor de la marca TIGO y se dedica a la distribución de equipos celulares, planes y ventas de recarga a nivel nacional en las zonas de Managua, Tipitapa, Masaya, Granada, Carazo, Rivas, Boaco, Rio Blanco, Siuna y puerto Cabezas.

1.1.1. MISIÓN

Satisfacer las necesidades de nuestros clientes a través de la distribución de productos y servicios de telecomunicaciones, fortaleciendo relaciones de largo plazo con los operadores generando valor a las personas de interés conectándolas alrededor del mundo. (DIGESA, 2023).

1.1.2. VISIÓN

Ser líder en distribución de telecomunicaciones en Nicaragua. (DIGESA, 2023).

1.1.3. VALORES

A continuación, los valores corporativos brindados por la empresa.

- **Transparencia:** Ser transparentes con los clientes en los servicios y productos brindados.
- **Proactividad:** Tomar la iniciativa al promover nuevas formas de hacer las cosas ofreciendo una variedad de productos y servicios de última tecnología.
- **Empatía:** La empatía se ve reflejada en el reconocimiento del esfuerzo de cada uno de sus colaboradores; también en el servicio que brinda a los clientes ajustando los productos y servicios a todo sector social.
- **Tolerancia:** Tratar a los clientes, empleados y socios con respeto, sin discriminación social alguna y tratando de llegar a acuerdos objetivamente.

1.1.4. ORGANIGRAMA DE LA EMPRESA

A continuación, se muestra cómo está estructurado el organigrama de la empresa actualmente, mostrando cada una de las unidades organizativas con sus respectivas jefaturas. Ver [Anexo No. 1](#): Estructura organizacional de la empresa.

1.1.5. NORMATIVAS

Cada empresa actúa de una manera determinada en función de una serie de pautas, valores o normas que rigen el comportamiento de los grupos que participan en la misma, y que la llevan a relacionarse con las demás empresas de una forma concreta y diferente. a continuación, se muestra las siguientes definiciones que forman parte de las normativas de ley correspondientes a la nómina, ya que por política de sigilo de la empresa no puede revelar sus políticas y procedimientos internos en referencia a la nómina.

El código del trabajo: es el documento jurídico laboral que regula los derechos y obligaciones de empleadores y colaboradores, así como las sanciones a imponerse ante el incumplimiento de los mismo por cualquiera de estas partes.

El código del trabajo de la república de Nicaragua establece las siguientes normativas en base a ley No. 185, publicada en la 18^{va} edición del 2016, editorial Jurídica.

Salario: Título IV. Cap. I. Pág. 32 art. 81.- Se considera salario la retribución que paga el empleador al trabajador en virtud del contrato de trabajo o relación laboral.

Vacaciones: Título III. Cap. III. Pág. 30 Art 76.- Todo trabajador tiene derecho a disfrutar de quince días de descanso continuo y remunerado en concepto de vacaciones, por cada seis meses de trabajo ininterrumpido al servicio de un mismo empleador.

Horas Extraordinarias: Título III. Cap. III. Pág. 27 Art 62.- Las horas extraordinarias y las que labore el trabajador en su día de descanso o compensatorio por cualquier causa, se pagarán un cien por ciento más de lo estipulado para la jornada normal respectiva.

Salario mínimo: Título IV. Cap. I. Pág. 32 Art 85.- Todo trabajador tiene derecho a un salario mínimo. Salario mínimo es la menor retribución que debe percibir el trabajador por los servicios prestados en una jornada ordinaria de trabajo, de modo que le asegure la satisfacción de las necesidades básicas y vitales de un jefe de familia.

Empleo: Título I. Cap. III. Pág. 8 Art. 13.- se encuentra establecido la definición de Empleo - El empleo o cargo es la ocupación o profesión ejercida por un trabajador con subordinación a otra persona denominada empleador, para prestar sus servicios de acuerdo a las responsabilidades que deben ser cumplidas.

Relación laboral y contrato de trabajo: Cap1. Artículo 19.- Relación laboral o de trabajo, cualquiera sea la causa que le de origen, es la prestación de trabajo de una persona natural subordinada a un empleador mediante el pago de una remuneración.

Contrato individual de trabajo es el convenio verbal o escrito entre un empleador y un trabajador, por el cual se establece entre ellos una relación laboral para ejecutar una obra o prestar personalmente un servicio. (Rigoberto Navas S., 2016, pág. 6).

Seguro Social: Ley de seguridad social decreto No 974, Título I, del seguro social y su campo de aplicación, Capítulo 1 de la institución y sus objetivos, Artículo 1. Se establece como parte del sistema de la Seguridad Social de Nicaragua, el Seguro Social Obligatorio, como un servicio público de carácter nacional, cuyo objetivo es la protección de los trabajadores y sus familias, de acuerdo a las actividades señaladas en esta Ley y su Reglamento.

El sistema de información a desarrollar realizará la generación de la nómina incorporando las normativas de leyes vigentes a nivel nacional que rigen el ejercicio del trabajo. Debido a lo antes expuesto se citan los artículos de la “Ley de seguridad social” y “Ley 185 del código del trabajo”; las mismas que fueron establecidas por el

Gobierno de la República de Nicaragua y que también se encuentran en la Constitución Política del país.

Estas leyes son de estricto cumplimiento para todas las entidades públicas como privadas; teniendo una implicación directa en los derechos y deberes tanto del empleador como del colaborador por ende repercuten en la generación de la nómina.

1.2. DESCRIPCIÓN DE LOS PROCESOS DE NEGOCIO

Un proceso es una concatenación lógica de actividades, a través del tiempo y lugar, impulsado por eventos y que, a través de su proceso de transformación, cumplen un determinado fin (Freund & Rucker, 2014).

El modelado de proceso es la captura de una secuencia de actividades de negocio y de la información de soporte. Dichos procesos de negocios describen como la empresa logra sus objetivos.

Para la generación de la nómina se identificó el proceso de selección y contratación del personal, así como proceso de generación de la nómina, los cuales se describen a continuación.

1.2.1. PROCESO DE SELECCIÓN Y CONTRATACIÓN

La selección de personal es el proceso que permite elegir al empleado ideal de entre los postulantes. Ese momento empieza con el contacto del candidato y que se produce justo después del reclutamiento.

El jefe de cada área que requiere contratar personal realiza la solicitud vía correo al departamento de gestión humana para una nueva contratación, en la solicitud se debe de indicar el nombre y las habilidades requeridas para desempeñar el puesto.

El área de gestión humana recibe la solicitud, y se procede en publicar la vacante por medio de los canales autorizados (redes sociales y agencias de empleos autorizadas), el tiempo estimado que espera Recursos humanos es de una semana máx. para que los posibles candidatos envíen su aplicación a la vacante.

En el proceso se seleccionan a varios candidatos que cumplan con los requisitos solicitados, se contacta a cada candidato para confirmar si está dispuesto a continuar con el proceso de selección, de ser así se establece una fecha para realizar varias pruebas técnicas, la evaluación de las pruebas va en dependencia del cargo a desempeñar.

Se remiten las hojas de vida de los candidatos seleccionados al jefe de área, quien realiza programación de citas para entrevistar a cada candidato. Posteriormente el jefe del área notifica al departamento de gestión humana el candidato seleccionado.

Para la contratación se solicita previamente que el candidato entregue copias de diplomas, certificados, constancias laborales, récord de policía, constancia judicial, copia de cédula de identidad, completar la solicitud de trabajo donde debe de llenar los datos personales del mismo, así como también se programa una cita médica para realizar exámenes médicos.

Una vez completado todos los requisitos se procede con informar al candidato las condiciones del contrato individual del trabajo, como el cargo a desempeñar, salario básico, horario laboral, funciones específicas del cargo, tipo de contrato (determinado, indeterminado o por servicio profesionales), periodos de pagos, tipo de moneda a pagar, condiciones por incumplimiento del contrato, prestaciones en base a la ley del código del trabajo de la república de Nicaragua y fecha en que inicia a laborar.

Los documentos físicos del nuevo colaborador, así como también el contrato son resguardados en el archivo interno.

El nuevo colaborador debe de cumplir con un periodo de prueba de tres meses, posterior el jefe inmediato debe de realizar una evaluación del desempeño laboral para la confirmación del cargo.

Concluido el periodo de prueba, el nuevo colaborador comienza a acumular prestaciones desde el primer día en que inicia la relación laboral. El colaborador aplica a los beneficios otorgados por la empresa, y es obligación de la empresa cumplir con los artículos estipulado en el Código del Trabajo de la República de Nicaragua, cabe mencionar que se aplican las deducciones de ley correspondientes.

La contratación por servicios profesional no aplica a beneficios otorgados por la empresa, así como también no aplica a los beneficios estipulados por la ley del código del trabajo de la república de Nicaragua, y se debe de aplicar la deducción del impuesto sobre la renta (IR). Se rige bajo el código civil de la república de Nicaragua.

1.2.2. PROCESO DE GENERACIÓN DE NÓMINA

La nómina es un documento soporte que permite llevar el control de los pagos y deducciones que se realizan a cada uno de los colaboradores, como retribución por los servicios que ellos prestan dentro de la empresa. Dicho documento se debe generar de forma quincenal. La remuneración que se detalla en la nómina se realiza de acuerdo con el tipo de contrato que ha firmado cada colaborador.

Para el proceso de la generación de la nómina, la analista de nómina solicita vía correo al analista de reclutamiento y selección las incidencias del personal (altas, bajas, cambios y llegadas tardes), también solicita al área de ventas el reporte de comisiones, viáticos de alimentación y transporte, depreciación.

De igual manera cada área entrega el reporte de horas extras y vacaciones del personal. La entrega de estos reportes se debe realizar con tres días de anticipación a la generación nómina.

La analista de nómina debe realizar el registro de cada una de las incidencias, compensaciones y deducciones de cada colaborador en un archivo de Excel.

Posteriormente la analista de nómina valida los datos ingresados en el archivo de Excel para proceder con la impresión de este y remitirlo a las autoridades correspondientes para su revisión.

Si el documento presenta inconsistencia esta se remite nuevamente al área de gestión humana para su corrección. Si el documento esta correcto se procede con la entrega de dicho documento al área de contabilidad para su contabilización, cabe mencionar que el área de gestión humana archiva una copia. Todo el proceso en promedio toma aproximadamente 20 horas laborales.

LAS POLÍTICAS DE COMPENSACIONES

El **salario básico** de cada colaborador está en dependencia del cargo que desempeña.

El pago de **horas extras** se establece el salario básico por hora multiplicado por dos.

El **viático** de transporte y alimentación se aplica para el analista de operaciones, jefe de operaciones, supervisores regionales y supervisores de zona, también para colaboradores que laboran en horarios extraordinarios a partir de las 5:30 pm.

La **depreciación** de vehículo y combustible se aplica a los vendedores y supervisores.

Las **comisiones** de ventas se aplican a los vendedores.

LAS POLÍTICAS DE DEDUCCIONES

La deducción por llegadas tarde el cual se aplican a partir de las 8:15 am, y se calcula: El total de horas de llegadas tardes en la quincena, multiplicado por salario básico por hora.

La deducción por seguro social (INSS) laboral el cual se aplica el 7% en base al ingreso.

La deducción de impuesto sobre la renta (IR), es aplicado después del INSS, y se aplica en base a la tabla establecida. Ver tabla-2: Impuesto sobre la renta.

Tabla 2 : Impuesto sobre la renta (IR)

Estratos de renta neta anual		Impuesto base	Porcentaje aplicable	Sobre exceso de
De C\$	Hasta C\$	C\$	%	C\$
-	1,000,000	-	0%	-
100,000	200,000	-	15%	100,000
200,000	350,000	15,000	20%	200,000
350,000	500,000	45,000	25%	350,000
500,000	A mas	82,500	30%	500,000

Fuente: (DGI, 2013).

El analista de nómina valida el salario mínimo según lo acordado con los sectores que integran la comisión nacional de salario mínimo, Acta 1, CNSM-23/02/2023.

El empleador debe cumplir las obligaciones de seguro social (INSS) patronal en el cual se aplica en 22.5% en base a ingreso por cada colaborador.

El empleador debe cumplir las obligaciones del pago de INATEC que es el 2% por cada colaborador.

Posteriormente la analista de nómina realiza la actualización de cada colaborador en el archivo de Excel, valida los registros de cada colaborador, así como también el resultado de los cálculos correspondientes en el archivo de Excel. Posteriormente procede a imprimir la hoja de cálculo de Excel que tiene la nómina, posteriormente debe ser autorizado por la Gerente de gestión humana y remitido al área de contabilidad para su revisión y aprobación.

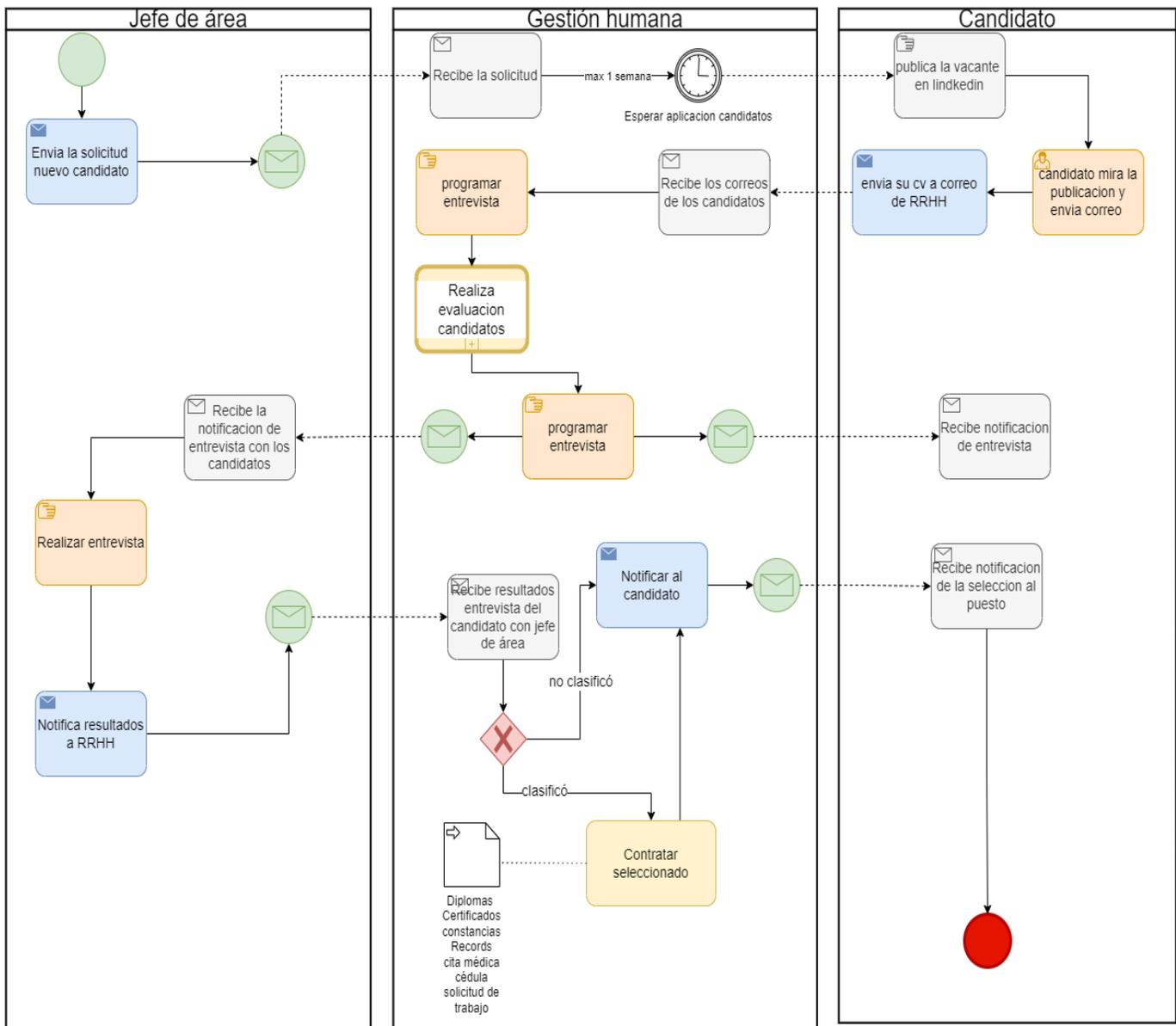
En caso de que el área de contabilidad identifique un error, el informe se regresa nuevamente al área de gestión humana para que rectifique el hallazgo encontrado.

1.3. DIAGRAMAS BPMN

A partir de la descripción de los procesos identificados, la siguiente ilustración muestra el diagrama de proceso de selección y contratación del personal. Ver ilustración 1: Proceso de selección y contratación del personal.

Ilustración 1 : Proceso de selección y contratación del personal.

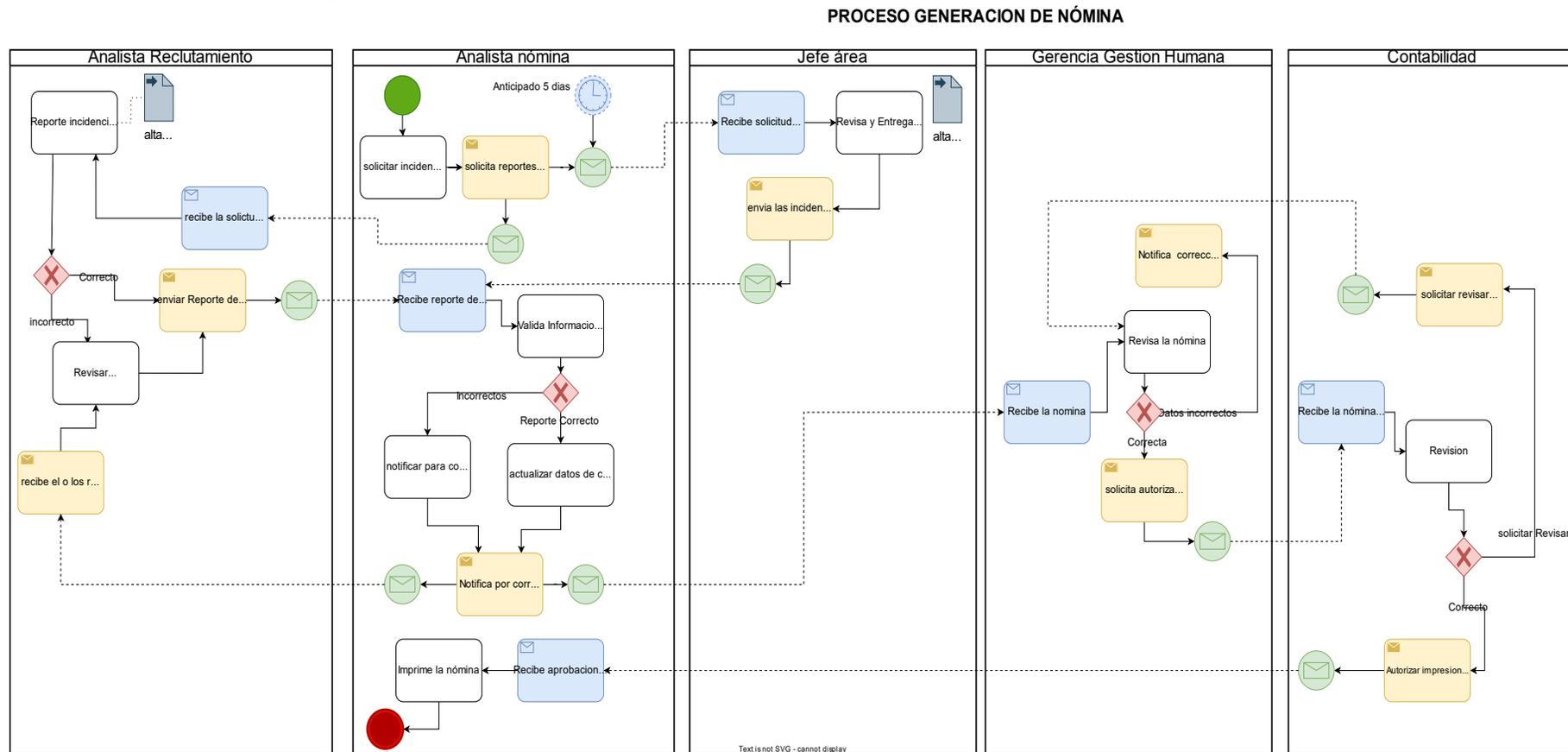
PROCESO DE SELECCIÓN Y CONTRATACIÓN DEL PERSONAL



Fuente: Elaboración propia.

Partiendo de la descripción anterior del proceso de generación de nómina, la siguiente ilustración muestra este proceso a como se encuentra estructurado actualmente. Ver ilustración 2: Proceso de generación de nómina.

Ilustración 2 : Proceso de generación de nómina.



Fuente: Elaboración propia.

1.4. OBJETIVOS DEL SISTEMA

Los principales objetivos de un sistema de información (basado en ordenadores) son proporcionar información que sirva: en el proceso de toma de decisiones y para el control de una organización.

Para el sistema información a desarrollar se establece los siguientes objetivos:

- Obj1.** Generar la nómina, llevando el control y generación del pago de salarios, una vez que todos los datos de los empleados y de la nómina estén correctos.
- Obj2.** Control de deducciones, cumpliendo con las obligaciones de pagos del personal con base en la ley y en las deducciones autorizadas.
- Obj3.** Generación de información administrativa y contable para la empresa y entes encargados de regular las relaciones laborales.
- Obj4.** Reducir los márgenes de errores en la generación de la nómina, teniendo la capacidad de reducir los márgenes de errores en la generación de la nómina, a través de los controles de la validación de los datos ingresados por el usuario.
- Obj5.** Generación de reportes gerenciales para análisis y toma de decisión, el sistema con la nueva herramienta generara reportes preestablecidos los cuales serán de mucha ayuda en la administración del negocio.
- Obj6.** Reducir el tiempo y el esfuerzo en la elaboración de los informes, con la nueva herramienta se agilizará esta actividad.
- Obj7.** Mantener la integridad, disponibilidad y confidencialidad de la información, al disponer de un sistema automatizado con la implementación de usuarios con sus respectivos roles.
- Obj8.** Generación del contrato laboral de forma automatizada para cada empleado contratado con la nueva herramienta.

1.5. ALCANCE DEL SISTEMA

El alcance de un sistema define sus límites, lo que está al alcance (dentro del sistema) y lo que está fuera de él. Por lo general el proyecto cuenta con un presupuesto que ayuda a definir el alcance, además del tiempo inicial y final. Los actores siempre están fuera del alcance del sistema.

El proceso de la generación de la nómina comienza cada vez que se inicia un período de pago, a partir del reporte del ingreso de novedades del personal y finaliza con la entrega del reporte de nómina para su posterior revisión y aprobación por parte del jefe de gestión humana.

1.6. BENEFICIOS DEL SISTEMA

Beneficios Tangibles: son las ventajas que se pueden medir en dinero y se acumulan en la organización a través del uso del sistema de información, estos permiten aumentar la eficiencia, calidad, rentabilidad y productividad de una empresa.

A continuación, se detallan los beneficios tangibles del sistema de información para la generación de la nómina:

- 1. Reducción del tiempo para la generación de la nómina:** Los cálculos y revisión de los datos conlleva a muchas horas cada mes, con un sistema para la generación de la nómina el personal encargado puede hacer un uso más eficiente de esas horas, así como reducir la fatiga y el estrés laboral del personal encargado.

2. No duplicidad de la información: Con la ayuda de un sistema de información para la generación de la nómina, los operadores pueden reducir los errores de esta índole, evitando así duplicidad en pagos y deducciones del personal.

3. Mejorar la respuesta, la atención y servicio al cliente: Con la ayuda del sistema se puede mantener actualizada la información de los colaboradores, de forma rápida y eficiente, lo cual mejora las respuestas a solicitudes, la atención y servicio al cliente interno.

4. Mejora en la gestión de recursos y crecimiento de la empresa: El sistema de información permitirá llevar un mejor control del recurso humano que tiene la empresa, tomando en cuenta los atributos que posee e incluso con la generación automatizada del contrato laboral se logra agilidad en esta gestión, permitiendo tomar decisiones en tiempo real en base a la administración del personal.

Beneficios Intangibles: son los menos cuantificables pero que económicamente tienen un impacto positivo en la compañía y están más orientados a beneficios estratégicos.

Los beneficios intangibles del sistema de información para la generación de la nómina son:

- 1. Reducción del margen de error en el procesamiento de nómina:** Con la ayuda de un sistema de nómina, se puede reducir considerablemente el margen de error debido a que la información generada proviene de un sistema especializado más preciso.

2. **Fácil generación de informes requeridos:** Con esta nueva herramienta se podrá tener en tiempo real los informes establecidos y necesarios por la gerencia. Lo que permitirá la toma de decisiones en un tiempo óptimo para el giro del negocio, mejorando la planificación estratégica de la empresa.

3. **Aumento de la transparencia, el control y seguridad de la información:** Esto se logra llevando un control de ingreso al sistema por medio de autenticación y asignación de roles de accesos, también garantizar el resguardo y respaldo de la información.

4. **Mejor y mayor control financiero:** Al generar la nómina en el sistema de información permite controlar de una manera más efectiva todos los egresos, provisiones de cuentas acumuladas por pagar entre otras cosas que se relacionan con la parte financiera de la empresa.

ACTORES INVOLUCRADOS

Se le llama actor a toda entidad externa al sistema que guarda una relación con éste y que le demanda una funcionalidad. Cada uno especifica un rol jugado por un usuario o cualquier otro sistema que interactúe con el sistema analizado; esto incluye a los operadores humanos, pero también incluye a todos los sistemas externos, además de entidades abstractas, como el tiempo.

A continuación, se describen los actores involucrados en el sistema de generación de nómina.

ACTORES INTERNOS

Colaborador: Representa a todos los colaboradores contratados por la empresa.

Analista de reclutamiento y selección: Es la colaboradora encargada de gestionar la selección y contratación del personal, así como también resguardar los documentos físicos correspondientes y de reportar las incidencias del personal (altas, bajas y cambios) a la Analista de Nómina.

Jefe de área: Es el encargado de remitir quincenalmente las incidencias de cada subordinado (vacaciones, comisiones, horas extras, viáticos).

Analista de nómina: Es la encargada de elaborar, revisar y validar la nómina quincenalmente.

Gerente de gestión humana: Es la encargada de aprobar el pago de liquidaciones y de la nómina.

Asistente contable: Es el encargado de revisar y validar el reporte de la nómina quincenal.

Contador: Da el visto bueno de la revisión realizada por el asistente contable.

Gerente financiero: Es el encargado de aprobar las revisiones realizadas por el área de contabilidad.

Gerente general: Es el encargado de autorizar el pago de la nómina quincenal.

ACTORES EXTERNOS A LA EMPRESA

INSS: Se registra el ingreso de cada colaborador que este sujeto a la deducción correspondiente con la ley del seguro social obligatorio.

DGI: Se registra el ingreso de cada colaborador que este sujeto a la deducción correspondiente con la ley del fisco (IR).

DEFINICIÓN DE USUARIOS

Los usuarios son las entidades que utilizan el sistema de información para capturar, introducir, validar, transformar y almacenar datos e información; las cuales poseen un conjunto de permisos y de recursos asignados como parte de una red informática, y que bien pueden ser una persona, un programa informático o un computador.

A continuación, se describen los usuarios del sistema de información:

- **Responsable IT:** Persona encargada de proveer soporte a los usuarios, y de otorgar acceso al sistema previamente autorizado por gerencia.
- **Especialista de selección y contratación:** Persona encargada de registrar los datos generales del empleado, cargos, dependencias, estudios académicos, etc., así como también de registrar las altas, bajas y cambios de los colaboradores.
- **Especialista de nómina:** Persona encargada de registrar las compensaciones y deducciones de los colaboradores, también de validar los cálculos relacionados a dicha nómina tal como el pago de INSS, IR, Horas Extras, etc.

1.7. REQUERIMIENTOS FUNCIONALES

Los requisitos funcionales se componen de dos partes: función y comportamiento. La función es lo que hace el sistema (por ejemplo: calcular el INSS laboral). El comportamiento es cómo lo hace el sistema (ejemplo: el sistema calculara el INSS laboral multiplicando el ingreso del colaborador por la tasa impuesta).

Los requerimientos funcionales que cumplirá el sistema serán los siguientes:

- ✓ **RF01 Generar reportes administrativos:** Reporte del aporte del INSS, reporte de comprobantes de pagos, reporte de constancia salarial, reporte de declaración del IR, reporte del depósito de cuentas de nómina, reporte del detalle de la nómina, reporte de la planilla de pago, reporte de la planilla por firma del empleado, reporte del aporte del INATEC.
- ✓ **RF02 Generar reporte de contrato laboral:** una vez registrado el colaborador se podrá imprimir su contrato laboral desde el sistema.
- ✓ **RF03 Administrar perfil del empleado:** se registran y actualizan los datos personales, experiencia laboral, familiares, beneficiarios.
- ✓ **RF04 Administrar dependencias:** se refiere a la configuración de cada una de las áreas compuestas por el organigrama de la empresa.
- ✓ **RF05 Administrar tipos académicos:** debe de permitir agregar y actualizar los títulos académicos de los empleados.
- ✓ **RF06 Administrar sucursales:** se refiere a cada una de las sucursales con las que cuenta la empresa, con el objetivo de agrupar a los empleados en cada una de las sucursales que corresponden.

- ✓ **RF07 Administrar clasificaciones académicas:** se administran los diferentes niveles de escolaridad de los empleados.
- ✓ **RF08 Administrar acreditaciones académicas:** define el grado académico que alcanzó el empleado o la persona a la vacante en la empresa.
- ✓ **RF09 Administrar cargos:** se definen los cargos y puestos que desempeñan los empleados.
- ✓ **RF10 Administrar instituciones educativas y no educativas:** describe las instituciones educativas y no educativas las cuales avalan los cursos y estudios de los candidatos y empleados.
- ✓ **RF11 Administrar parentesco:** se registran y actualiza los datos de los familiares de empleados una vez contratados.
- ✓ **RF12 Administrar municipios:** contiene la clasificación de los municipios por departamento del país.
- ✓ **RF13 Administrar departamentos:** se registra y actualiza los datos de ubicación donde habita el empleado.
- ✓ **RF14 Generar la nómina:** permitir generar la nómina, una vez los datos estén verificados. y tanto las compensaciones y deducciones del periodo hayan sido ingresados.
- ✓ **RF15 Importar deducciones y compensaciones del empleado:** permite importar los datos de forma masiva desde una plantilla de Excel para los datos de las compensaciones y deducciones.

- ✓ **RF16 Registrar compensaciones y deducciones del empleado:** nos permite registrar las compensaciones y deducciones que serán aplicadas empleado.
- ✓ **RF17 Administrar estudios académicos:** define los estudios que contienen al menos un título académico de alguna institución.
- ✓ **RF18 Administrar grupos de compensaciones y deducciones:** nos permitirá agrupar las compensaciones y deducciones por tipo.
- ✓ **RF19 Abrir periodos de nómina:** permite crear el periodo en que va a generarse la nómina.
- ✓ **RF20 Agregar empleado a nómina:** al momento de importar las compensaciones y deducciones se puede agregar un empleado a la nómina, una vez inicializado el proceso de nómina.
- ✓ **RF21 Registrar empleado:** permite ingresar empleados de nuevo ingreso.
- ✓ **RF22 Generar reporte de comprobante de pago:** permite generar colilla de pago una vez aplicada la nómina.
- ✓ **RF23 Generar contrato laboral:** permite generar el contrato laboral de cada empleado que será contratado una vez implementado el sistema.

1.8. REQUERIMIENTOS NO FUNCIONALES

Los requerimientos no funcionales son las restricciones o los requisitos impuestos al sistema.

- ✓ **RNF01 Requerimiento de seguridad:** el sistema debe contar con los roles, permisos y segmentar los módulos con el fin de controlar los accesos de los usuarios.
- ✓ **RNF02 Requerimiento de rendimiento:** el sistema debe ejecutar las tareas de forma eficiente, sin retrasos en la cargas y solicitudes de información. Deberá tener un tiempo de respuesta de al menos 3 segundo.
- ✓ **RNF03 Requerimiento de almacenamiento:** se tendrá definido el lugar en el servidor donde serán almacenado los datos, este espacio tiene que ser mayor a 5GB.
- ✓ **RNF04 Requerimiento de hardware:** deberá contar con un servidor con los requisitos mínimos para ejecutar el sistema sin problemas, los cuales son al menos 4GB de memoria RAM dedicada, procesador Core i5, almacenamiento de al menos 256GB.
- ✓ **RNF05 Requerimiento de red:** se deberá contar con una conexión a Internet para que los usuarios tengan acceso al sistema. La empresa cuenta con una red LAN con acceso a internet, el servicio de internet es provisto por el proveedor IBW con un ancho de banda de 50mbps, de forma simétrica con enlace de fibra óptica dedicado.
- ✓ **RNF06 Requerimiento de interfaz:** el sistema deberá presentar una interfaz simple, agradable visualmente, responsiva, de fácil uso con el fin que el usuario puede explorar y comprender su funcionalidad.

Para los RNF04 y RNF05 la empresa cuenta con su propio servidor, donde será instalado el sistema web, para ver las características revisar a partir de la [página 78](#).

1.9. TABULACIÓN DE LOS REQUERIMIENTOS

A continuación, se muestra la tabulación de los requerimientos tanto funcional como los no funcionales que se contemplan en el sistema, los cuales permitirán definir claramente cada uno de sus atributos como responsables, dependencias, fuentes, descripción, nivel de importancia, estado, entre otros. Para el resto de las ilustraciones revisar a partir de la sección de [Anexo No. 2](#).

1.9.1. TABULACIÓN DE REQUERIMIENTOS FUNCIONALES

RF-0001 - Generar reportes administrativos.

FRQ-0001	Generar reportes administrativos
Versión	1.0 (19/04/2023)
Autores	<ul style="list-style-type: none"> • Responsable TI
Fuentes	<ul style="list-style-type: none"> • Especialista de Nómina • Especialista de selección y contratación
Dependencias	<ul style="list-style-type: none"> • [OBJ-0004] Reducir los márgenes de errores en la generación de la nómina • [FRQ-0019] Aperturar periodos de nómina • [FRQ-0014] Generar nómina
Descripción	El sistema deberá <i>generar reporte del aporte del INSS, reporte de comprobantes de pagos, reporte de constancia salarial, reporte de declaración del IR, reporte del depósito de cuentas de nómina, reporte del detalle de la nómina, reporte de la planilla de pago, reporte de la planilla por firma del empleado, reporte del aporte del INATEC.</i>
Importancia	importante
Urgencia	inmediatamente
Estado	en construcción
Estabilidad	media
Comentarios	Ninguno

Fuente: Elaboración propia.

RF-0002 - Generar reporte de contrato laboral.

FRQ-0002	Generar reporte de contrato laboral
Versión	1.0 (19/04/2023)
Autores	<ul style="list-style-type: none"> • <u>Eddy Sarantes</u> • <u>Harold Solis</u> • <u>Jeferson Cano</u>
Fuentes	<ul style="list-style-type: none"> • <u>Especialista de Nómina</u> • <u>Especialista de selección y contratación</u>
Dependencias	<ul style="list-style-type: none"> • <u>[OBJ-0004] Reducir los márgenes de errores en la generación de la nómina</u> • <u>[OBJ-0001] Generar la nómina</u>
Descripción	El sistema deberá <i>una vez registrado el colaborador se podrá imprimir su contrato laboral desde el sistema.</i>
Importancia	vital
Urgencia	inmediatamente
Estado	pendiente de verificación
Estabilidad	alta
Comentarios	Ninguno

Fuente: Elaboración propia.

1.9.2. TABULACIÓN DE REQUERIMIENTOS NO FUNCIONALES

RNF-0001 - Requerimiento de seguridad.

NFR-0001	Requerimiento de seguridad
Versión	1.0 (22/04/2023)
Autores	<ul style="list-style-type: none"> • <u>Eddy Sarantes</u> • <u>Harold Solis</u> • <u>Jeferson Cano</u>
Fuentes	<ul style="list-style-type: none"> • <u>Responsable TI</u>
Dependencias	• <u>[OBJ-0007] Mantener la integridad, disponibilidad y confidencialidad de la información</u>
Descripción	El sistema deberá <i>contar con los roles, permisos y segmentar los módulos con el fin de controlar los accesos de los usuarios.</i>
Importancia	importante
Urgencia	inmediatamente
Estado	en construcción
Estabilidad	media
Comentarios	Ninguno

Fuente: Elaboración propia.

RNF-0002 - Requerimiento de rendimiento.

NFR-0002	Requerimiento de rendimiento
Versión	1.0 (22/04/2023)
Autores	<ul style="list-style-type: none"> • <u>Eddy Sarantes</u> • <u>Harold Solis</u> • <u>Jeferson Cano</u>
Fuentes	<ul style="list-style-type: none"> • <u>Responsable TI</u>
Dependencias	Ninguno
Descripción	El sistema deberá <i>ejecutar las tareas de forma eficiente, sin retrasos en la cargas y solicitudes de información. Deberá tener un tiempo de respuesta de al menos 3 segundo.</i>
Importancia	importante
Urgencia	puede esperar
Estado	en construcción
Estabilidad	media
Comentarios	Ninguno

Fuente: Elaboración propia.

1.10. MATRIZ DE TRAZABILIDAD

Este documento contiene los requisitos del sistema Automatización para la generación de la nómina de la empresa DIGESA. A continuación, se muestra la matriz de trazabilidad y en ella se pueda visualizar como cada uno de los requerimientos funcionales nos ayudan a lograr los objetivos planteados.

Ilustración 3 : Matriz de trazabilidad.

TRM-0001	OBJ-0001	OBJ-0002	OBJ-0003	OBJ-0004	OBJ-0005	OBJ-0006	OBJ-0007	OBJ-0008
FRQ-0001	-	-	-	↑	-	-	-	-
FRQ-0002	↑	-	-	↑	-	-	-	↑
FRQ-0003	↑	↑	↑	-	↑	-	-	↑
FRQ-0004	↑	-	↑	-	-	-	-	-
FRQ-0005	-	-	-	-	↑	-	-	-
FRQ-0006	-	-	↑	-	↑	-	-	-
FRQ-0007	-	-	↑	-	-	-	-	↑
FRQ-0008	-	-	↑	-	-	-	-	-
FRQ-0009	-	-	↑	-	-	-	-	-
FRQ-0010	-	-	↑	-	-	-	-	-
FRQ-0011	-	-	↑	-	↑	-	-	-
FRQ-0012	-	-	↑	-	-	-	-	-
FRQ-0013	-	-	↑	-	-	-	-	-
FRQ-0014	↑	↑	↑	-	↑	↑	↑	-
FRQ-0015	-	↑	-	-	-	-	-	-
FRQ-0016	-	↑	-	-	-	-	-	-
FRQ-0017	-	-	↑	-	-	-	-	-
FRQ-0018	-	↑	-	-	-	-	-	-
FRQ-0019	↑	↑	↑	-	↑	↑	↑	-
FRQ-0020	↑	-	-	-	-	-	-	-
FRQ-0021	↑	-	-	-	-	-	-	↑
FRQ-0022	-	-	↑	-	-	-	-	-
FRQ-0023	-	-	-	-	-	-	-	↑

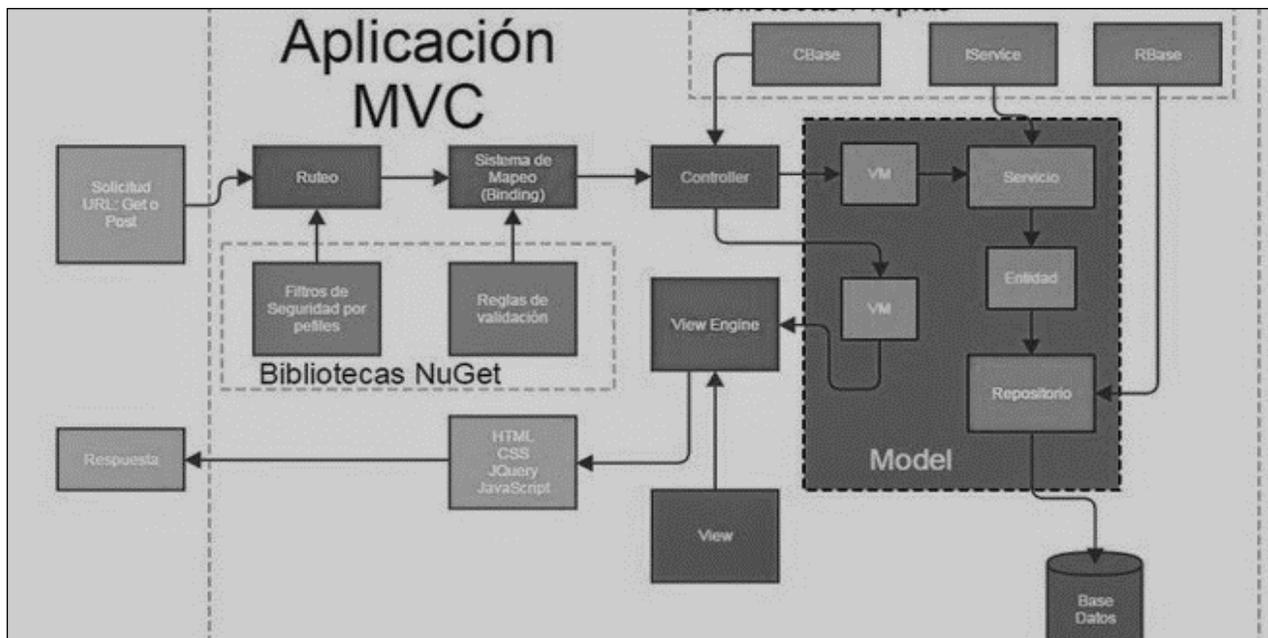
Fuente: Elaboración propia.

CAPÍTULO II DISEÑO DEL SISTEMA

En el siguiente capítulo se abordan temas importantes relacionados con el diseño del sistema de información, tales como los patrones de diseño a implementar que nos muestran cómo va estructurada la codificación con buenas prácticas; como va estructurado el modelo UWE con sus diagramas de casos de uso, modelo de contenido, modelo de navegación, modelo de proceso, diagramas de actividades, modelo de presentación de la interfaz y el diagrama físico de la base de datos creada.

2.1. DISEÑO DE ARQUITECTURA DEL SISTEMA

El diseño de una arquitectura de software utiliza los conocimientos de programación para planear el diseño general del software de modo que puedan agregarse detalles más adelante, lo cual permite a los equipos de software delimitar el panorama general y comenzar a elaborar un prototipo.



Fuente: (Ojeda, 2015)

2.2. PATRÓN DE DISEÑO DE LA ARQUITECTURA

Para el desarrollo del software es esencial implementar buenas prácticas en la codificación, que permita organizar, estructurar y agilizar el proceso de desarrollo, de igual manera que sea de fácil mantenibilidad al momento de integrar nuevos módulos o modificar los existentes con el paso del tiempo.

Con la programación orientada a objetos es requerido considerar algunos patrones que permita cumplir con los fines antes mencionados:

El Patrón de diseño en capas se puede utilizar para estructurar el software a desarrollar, ya que dicho patrón permite descomponer en grupos de subtareas las capas, las cuales se encuentra en un nivel particular de abstracción. Cada capa proporciona servicios a la siguiente capa superior.

Con el uso del patrón de diseño MVC, básicamente es un patrón que tiene como principio establecer una estructura en tres componentes: Modelo, vista y controlador. Y separarlos en diferentes objetos.

Cabe mencionar que este patrón de diseño es uno de los más utilizados en la actualidad para la programación orientada a objetos.

Entonces para la construcción del software se considerará estructurarlo en las siguientes capas:

Capa de Modelo: Contiene la estructura de la base de datos, sus entidades y relaciones, no contiene ninguna lógica que describa como presentar los datos al usuario.

Capa de Vista: Esta capa es la encargada de presentar los datos del modelo al usuario.

Capa de Controlador: Básicamente es la encargada de gestionar las instrucciones que se recibe, atenderlas y procesarla por medio de acciones establecidas en el controlador. Cabe mencionar que por medio de esta capa se establece comunicación entre la capa de modelo y la capa de presentación.

Las tres capas antes mencionadas están interconectadas, la capa de presentación muestra el modelo para el usuario, el controlador acepta las entradas del usuario y actualiza el modelo y debido a esta acción la presentación vuelve a tener un cambio con los datos actualizados.

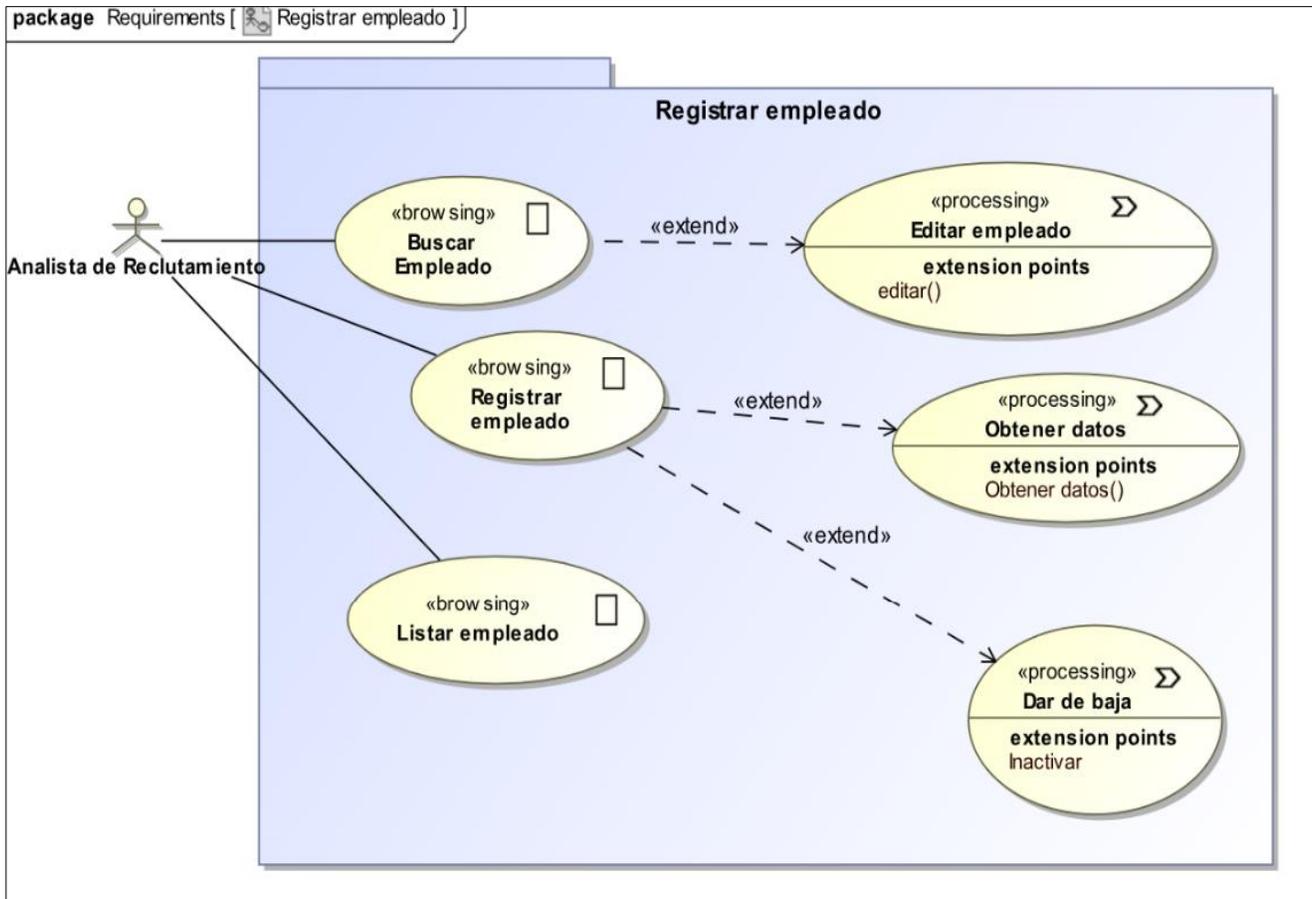
2.3. MODELOS UWE

Es una metodología de la Ingeniería de Software, una extensión del UML para desarrollar aplicaciones WEB, en ella se puede realizar diagramas representando los requisitos del usuario y documentar de mejor manera el funcionamiento que tendrá la aplicación. A continuación, se mostrarán los modelos de diagramas a utilizar para la aplicación.

2.3.1. MODELO DE CASOS DE USO

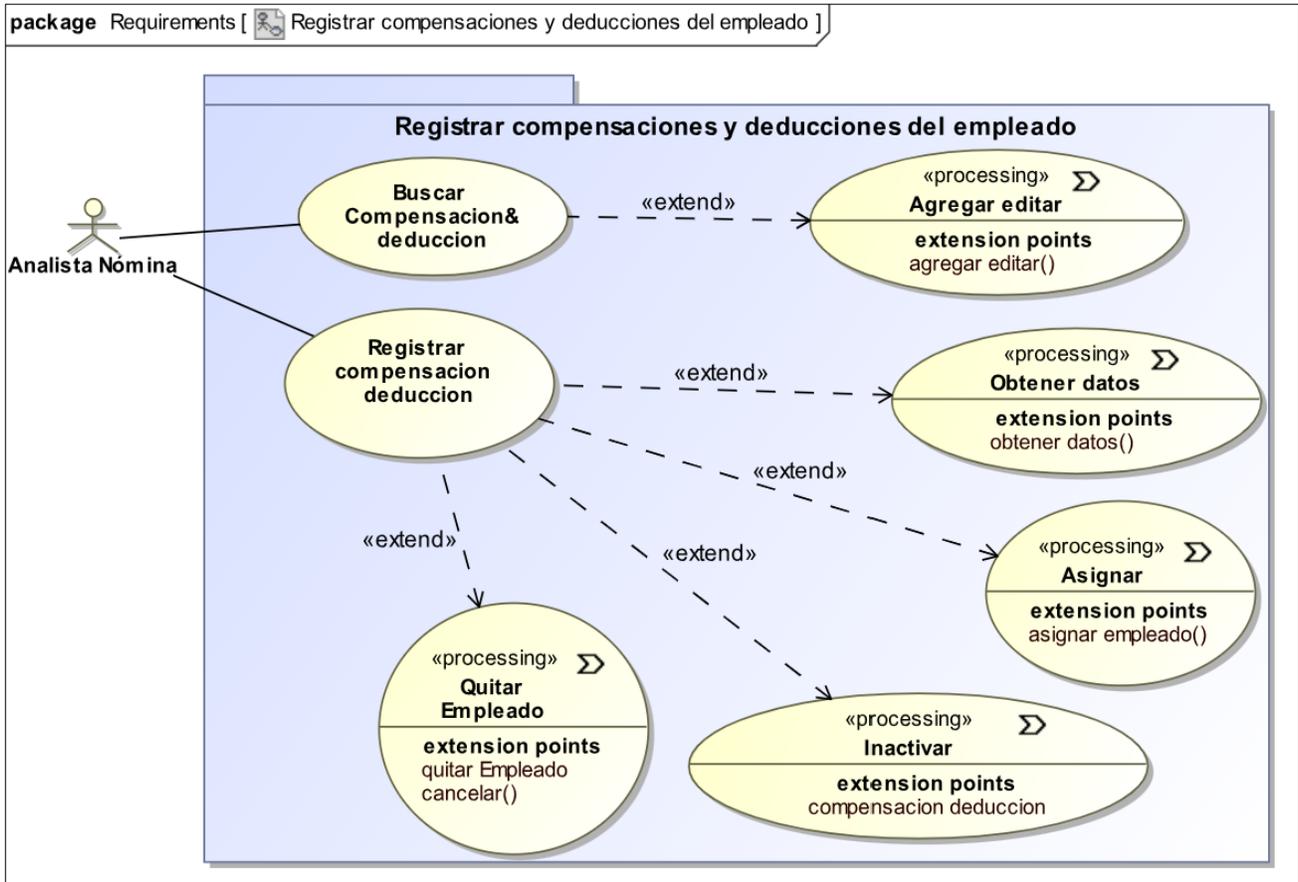
A continuación, se muestran los diagramas de “casos de usos” por cada uno de los requerimientos funcionales del sistema. Para el resto de los diagramas revisar a partir de la sección de [Anexo No. 3](#).

Ilustración 4 : Diagrama de casos de uso - Registrar empleado.



Fuente: Elaboración propia.

Ilustración 5 : Diagrama de casos de uso - Registrar compensaciones y deducciones del empleado.

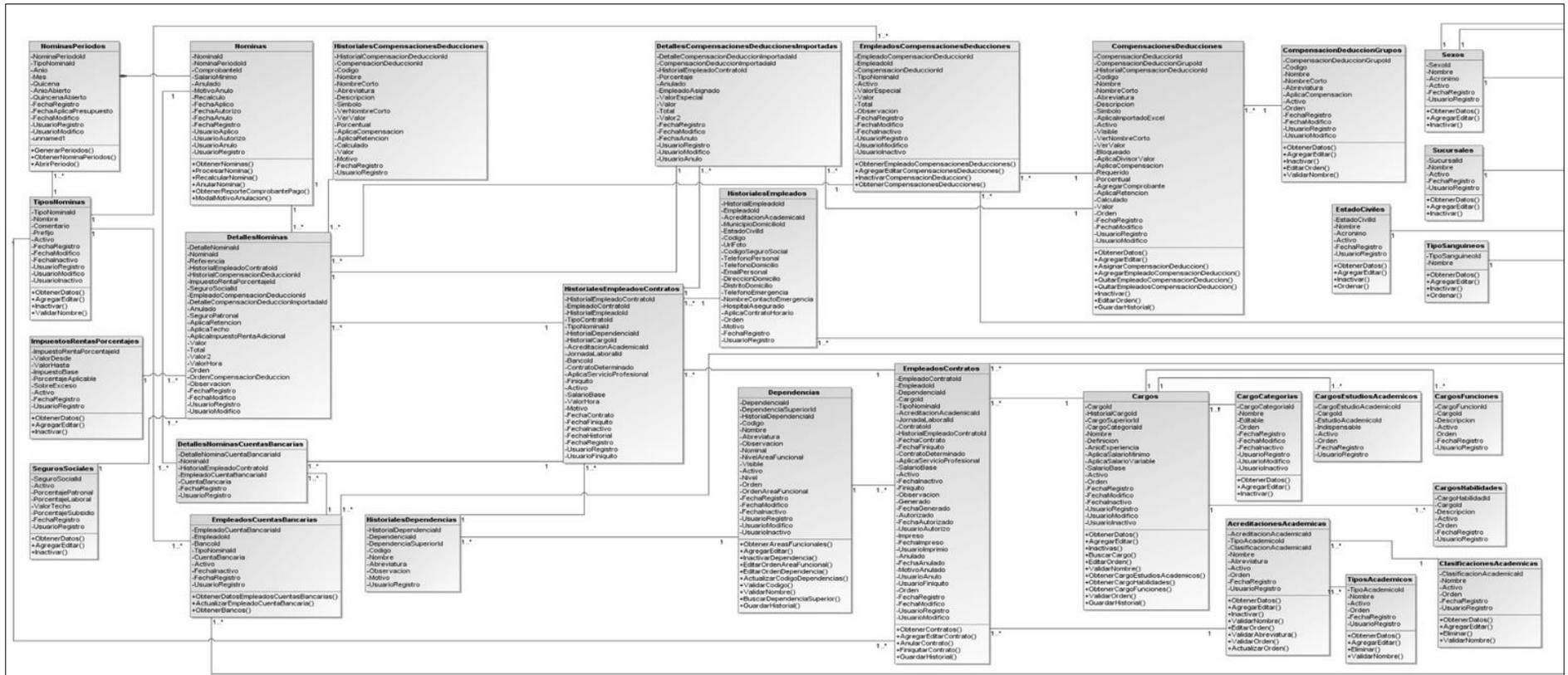


Fuente: Elaboración propia.

2.3.2. MODELO DE CONTENIDO

A continuación, se muestran los diagramas de contenidos del sistema, donde se puede observar la estructura de cada una de las relaciones existentes entre sus objetos. Para el resto de este diagrama revisar completo en sección de [Anexo No. 4](#).

Ilustración 6 : Diagrama de contenido.

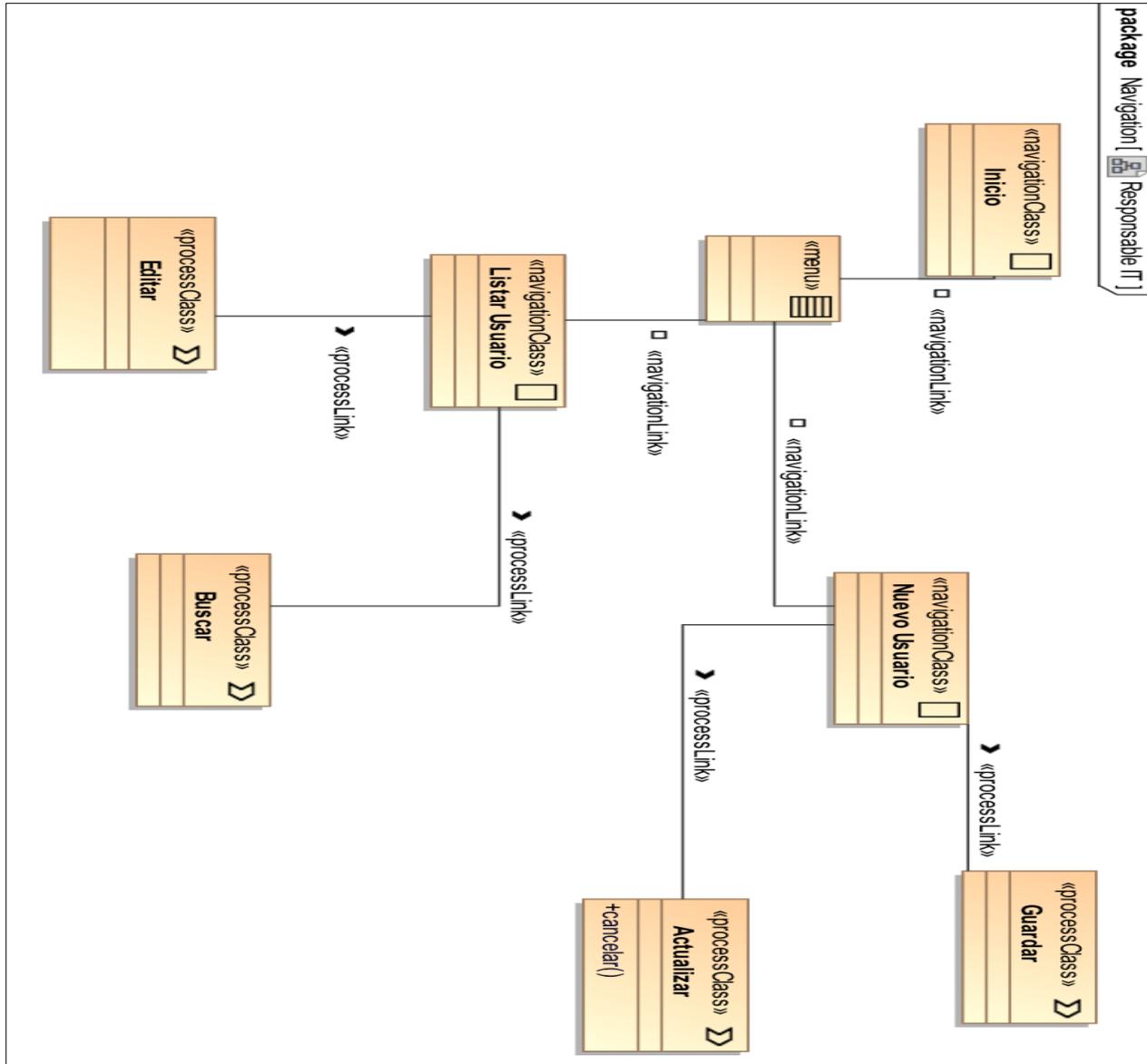


Fuente: Elaboración propia.

2.3.3. MODELO NAVEGACIONAL

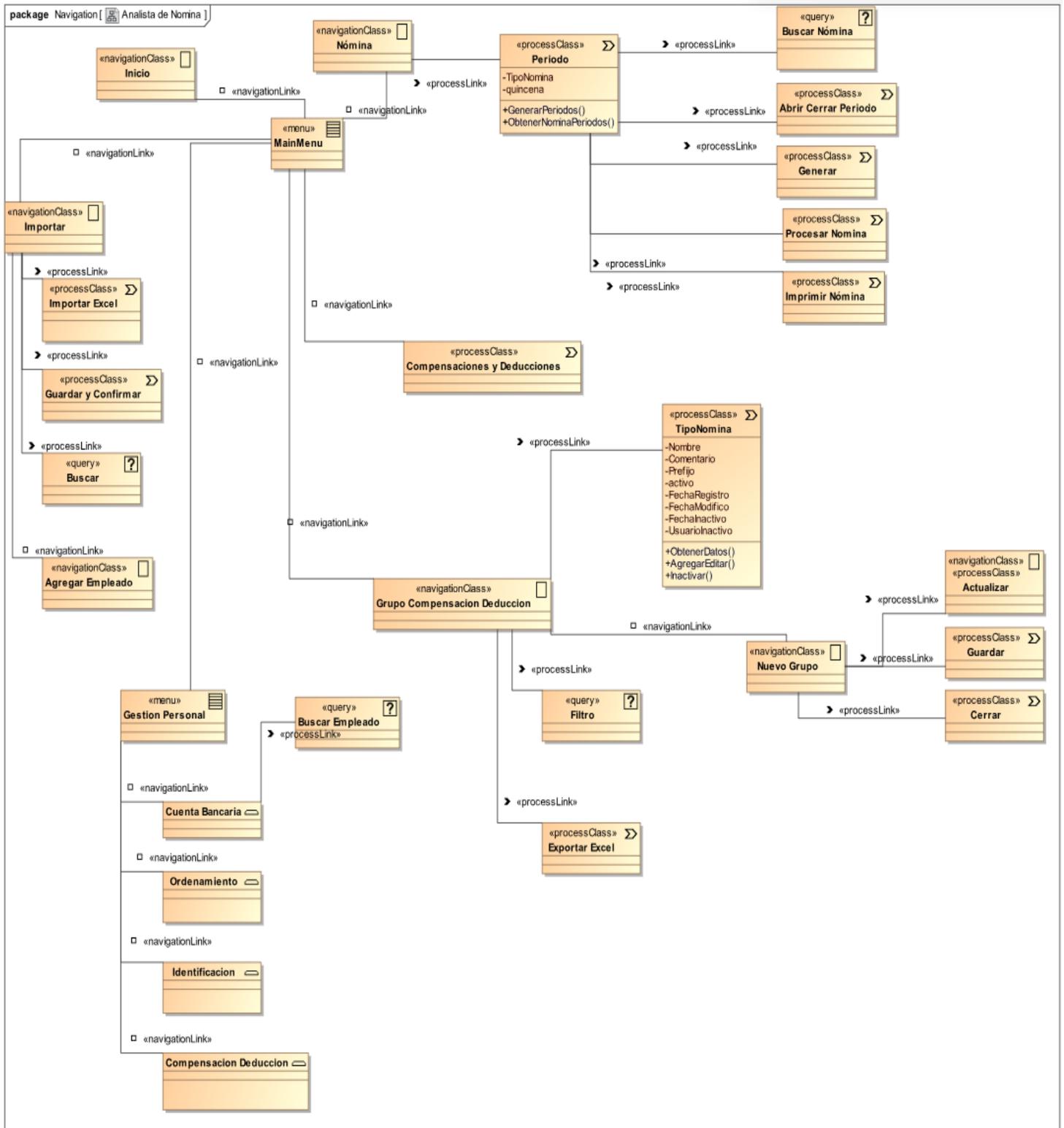
A continuación, se muestran los diagramas de navegación en los cuales se representa la navegación de cada uno de los actores dentro del sistema.

Ilustración 7 : Diagrama de navegación - Responsable IT.



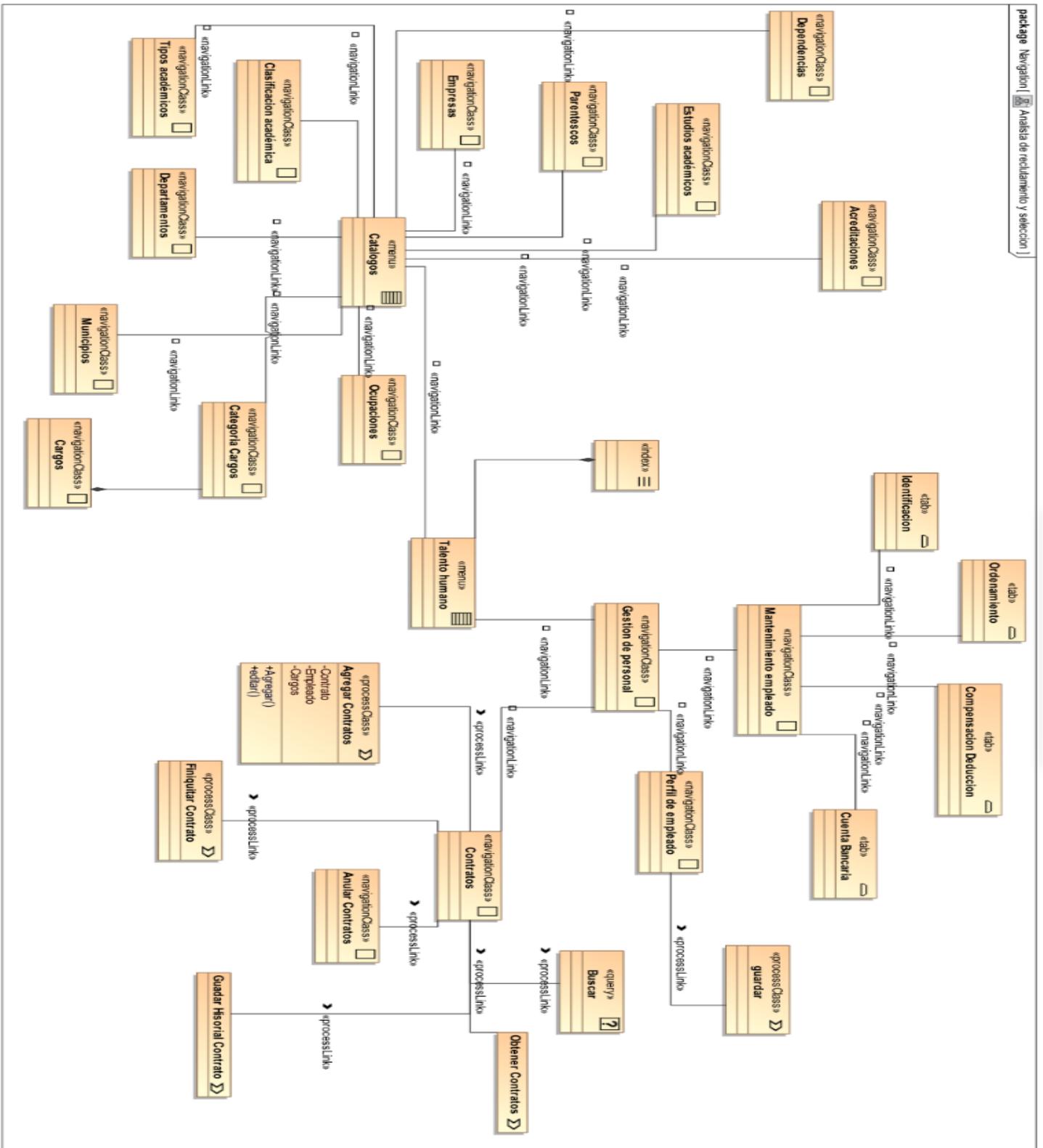
Fuente: Elaboración propia.

Ilustración 8 : Diagrama de navegación - Analista de nómina.



Fuente: Elaboración propia.

Ilustración 9 : Diagrama de navegación - Analista de reclutamiento y selección.

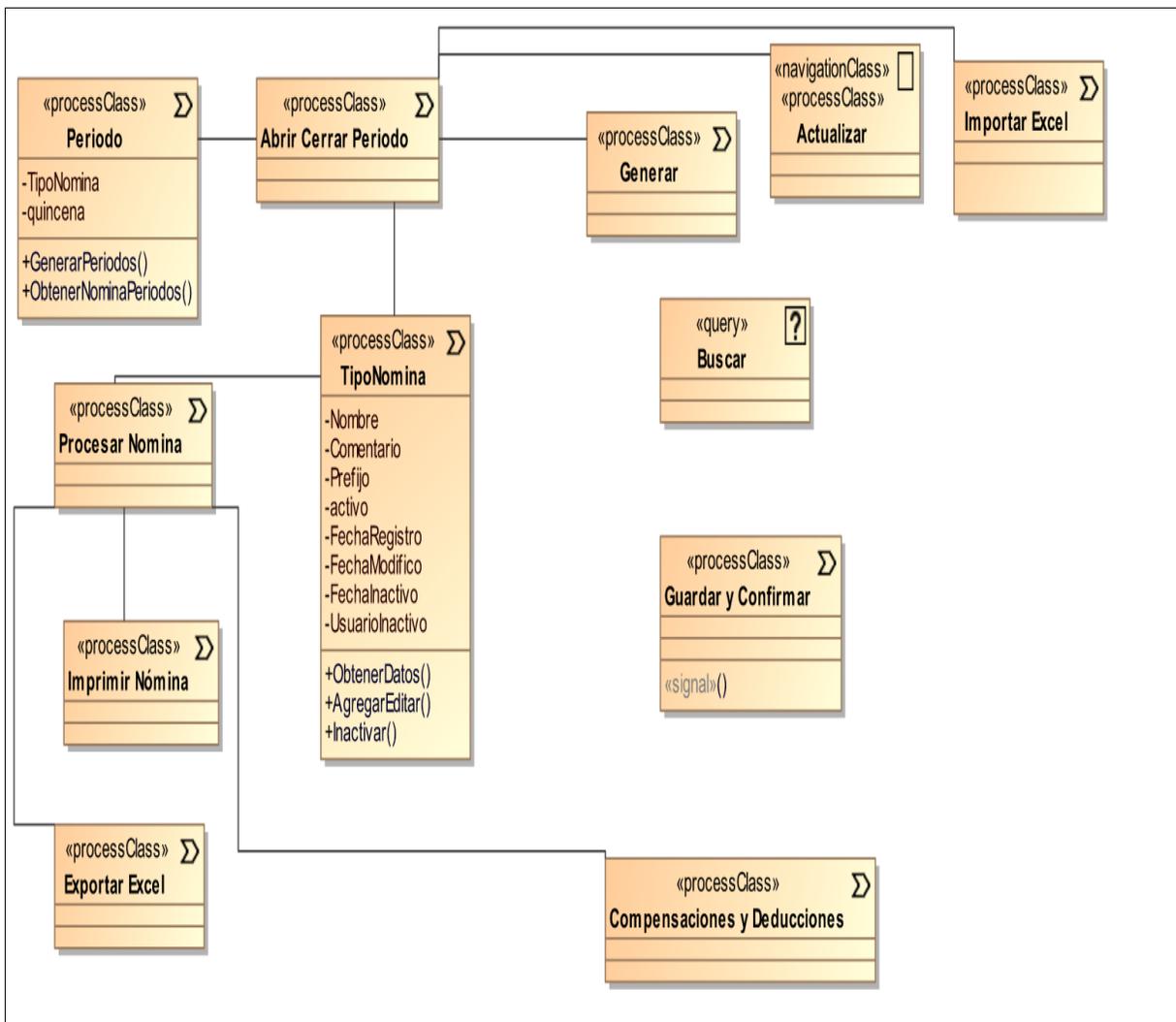


Fuente: Elaboración propia.

2.3.4. MODELO DE PROCESO

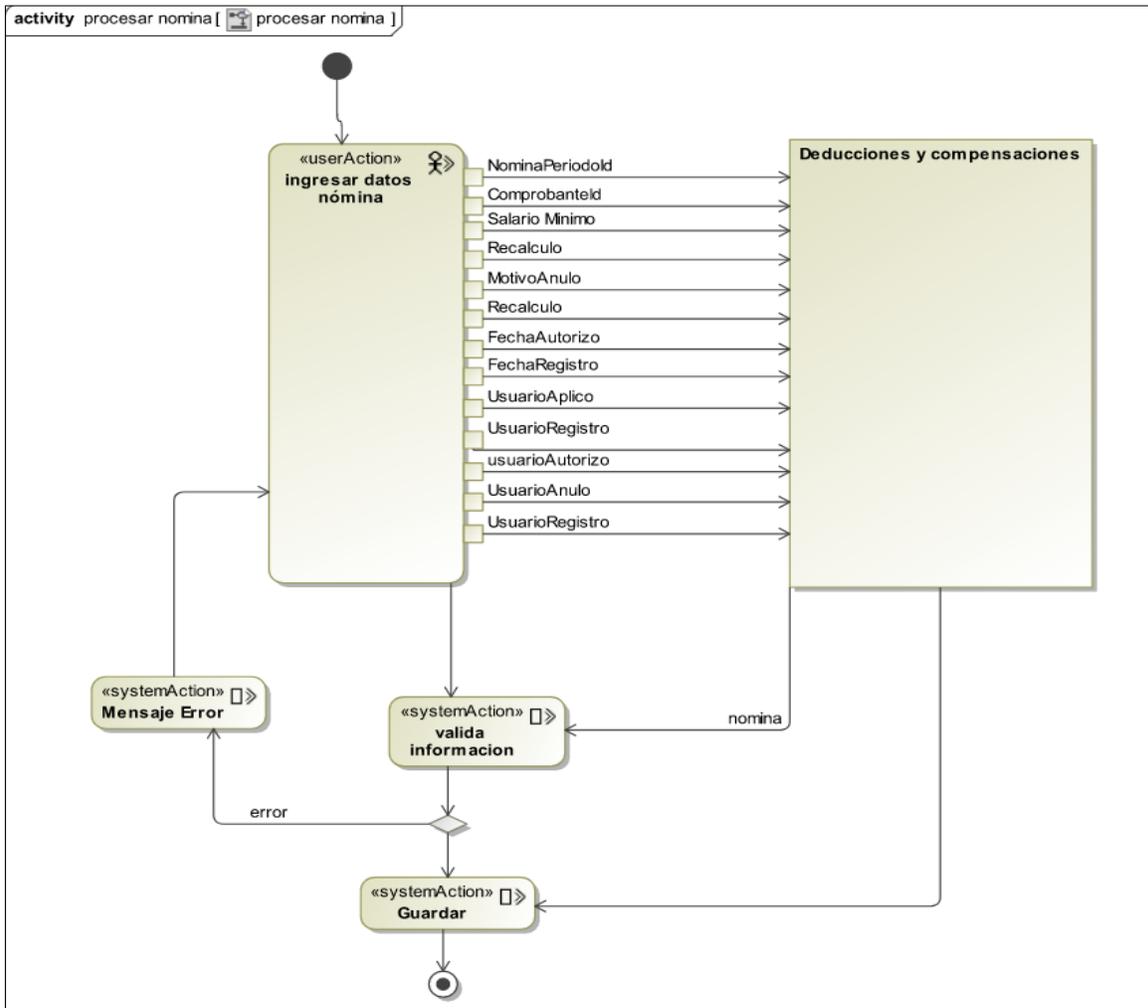
El modelo de proceso se deriva de los diagramas de: clases, flujo de proceso que deriva de los diagramas de actividades que conforman el sistema. Los diagramas de procesos se pueden visualizar a partir de la sección de [Anexo No. 5](#) y los diagramas de actividad a partir del [Anexo No. 6](#).

Ilustración 10 : Diagrama de proceso - generación de nómina.



Fuente: Elaboración propia.

Ilustración 11 : Diagrama de actividad - Procesar nómina.

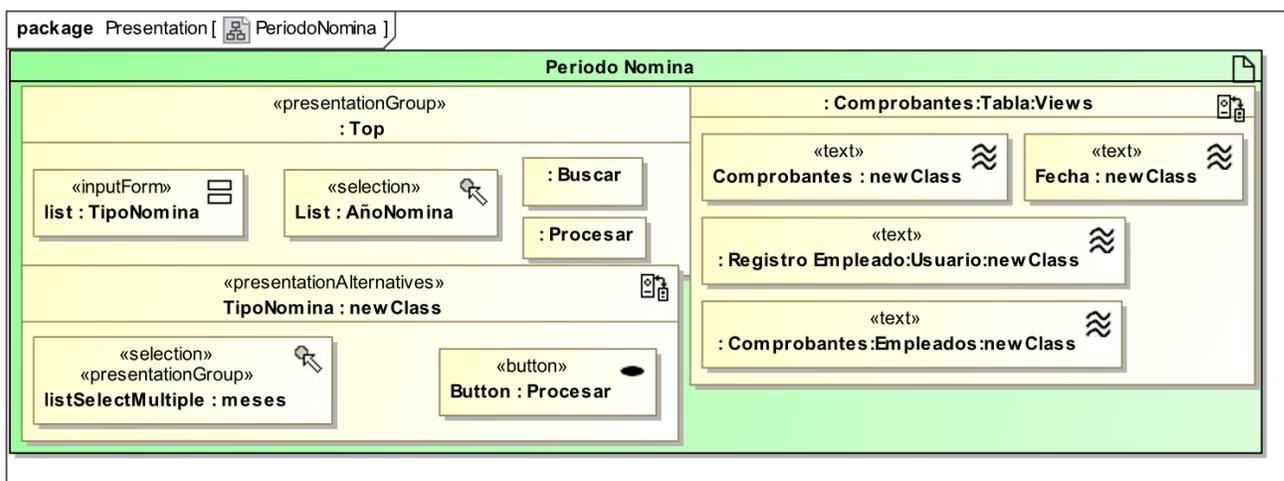


Fuente: Elaboración propia.

2.3.5. MODELO DE PRESENTACIÓN

A continuación, se muestra modelo de presentación, el cual representa las interfaces de usuario por medio de las vistas correspondientes a la presentación establecida en el sistema (botones, interfaz gráfica, etc.). Para el resto de los diagramas revisar a partir de la sección de [Anexo No. 7](#).

Ilustración 12 : Diagrama de presentación - Período nómina.



Fuente: Elaboración propia.

CAPÍTULO III – CONSTRUCCIÓN DEL SISTEMA

En este capítulo se abordan temas medulares para la construcción del sistema, mencionando las tecnologías y herramientas utilizadas tanto de hardware como software, patrones de diseño, clases implementadas, interfaz del sistema, entre otros; las cuales fueron fundamentales para cumplir los objetivos planteados en este presente trabajo monográfico.

3.1. SELECCIÓN DEL HARDWARE PARA LA CONSTRUCCIÓN Y ALOJAMIENTO DEL SISTEMA

Para la construcción del software se utilizará tres computadoras laptop con las siguientes características:

Tabla 3 : Especificaciones de laptop utilizadas.

LAPTOP 1	
Marca	DELL Latitude 3420
Procesador (CPU)	Intel Core i5 2.4 GHz.
Memoria RAM	16 GB
Disco Duro	256 SSD

LAPTOP 2	
Marca	DELL Latitude 5400
Procesador (CPU)	Intel Core i5 1.9 GHz.
Memoria RAM	8 GB
Disco Duro	256 SSD

LAPTOP 3	
Marca	DELL Inspiron 3493
Procesador (CPU)	Intel Core i3 1.20 GHz.
Memoria RAM	16 GB
Disco Duro	256 SSD

La empresa cuenta con su propio servidor donde será instalado el sistema web una vez construido; el servidor tiene las siguientes cualidades:

- Servidor HP ProLiant DL380 gen8.
- Procesador Intel® Xeon® E5-2609 v2 (2.5GHz/4-core/10MB/6.4GT-s QPI/80W).
- Capacidad de 8GB RAM.
- HP Ethernet 10Gb 2-port 530FLR-SFP+ Adapter.
- Almacenamiento SATA de 1tb de capacidad en RAID.

Así mismo la empresa cuenta con un NAS para el respaldo de información y como respaldo ante una posible falla del servidor principal. Las características del NAS son las siguientes:

- CPU AMD Ryzen v1500B.
- 4 GB DDR4 ECC SODIMM x2.
- Almacenamiento en RAID de 16TB.
- Conexión Ethernet de 10GBs.

3.2. SELECCIÓN DEL SOFTWARE PARA LA CONSTRUCCIÓN

Para el desarrollo del software se utilizará el IDE Microsoft Visual Studio 2019 Community. Cabe mencionar que la licencia del IDE a utilizar no requiere de pago y es de libre uso para fines educativos o para proyectos de código abierto. Con dicho IDE se puede editar, compilar, depurar y publicar proyectos en código C#, HTML, JavaScript, etc.

El motor de bases de datos a utilizar será Microsoft SQL Express 2019, el cual es un sistema de administración de base de datos gratuito, que ofrece un almacenamiento de datos de hasta 10GB.

3.3. ESPECIFICACIÓN DE LAS CLASES DEL SISTEMA

A continuación, se muestran cada una de las clases implementadas en el sistema como están conformadas y su respectivo código.

3.3.1. Login

```
15 [Authorize]
16 Oreferencias
17 public class LoginController : Controller
18 {
19     private readonly Context _db = new Context();
20
21     Oreferencias
22     public ActionResult Index()
23     {
24         return View();
25     }
26
27     [HttpPost]
28     [ValidateAntiForgeryToken]
29     Oreferencias
30     public ActionResult VerificarAcceso()...
31
32 }
33 }
```

3.3.2. Administrar Usuarios

```
17 public class UsuariosController : Controller
18 {
19     private readonly Models.Context _db = new Models.Context();
20     private readonly ModelStateValid _ModelStateValid = new ModelStateValid();
21
22     // GET
23     Oreferencias
24     public ActionResult Index()...
25
26     // POST: TalentoHumano/Usuarios/ObtenerDatos
27     [HttpGet]
28     Oreferencias
29     public JsonResult ObtenerDatos()...
30
31     // POST: TalentoHumano/Usuarios/AgregarEditar
32     [ValidateAntiForgeryToken]
33     Oreferencias
34     public JsonResult AgregarEditar([Bind(Include =
35         "NombreCompleto,Cargo,NombreUsuario,Contraseña,Correo")] Usuario usuario, int usuarioId)...
36
37     // POST: TalentoHumano/Usuarios/Inactivar
38     [HttpPost]
39     [ValidateAntiForgeryToken]
40     Oreferencias
41     public ActionResult Inactivar(short id)...
42
43 }
```

3.3.3. Administrar Acreditaciones Académicas

```
11 public class AcreditacionesAcademicasController : Controller
12 {
13     private readonly Context _context = new Context();
14     private readonly ModelStateValid _ModelStateValid = new ModelStateValid();
15
16     // GET: TalentoHumano/AcreditacionesAcademicas
17     Oreferencias
18     public ActionResult Index()...
19
20
21
22
23
24     // GET: TalentoHumano/AcreditacionesAcademicas/ObtenerDatos
25     [HttpGet]
26     Oreferencias
27     public ActionResult ObtenerDatos()...
28
29
30
31
32
33
34
35
36
37
38
39
40     // GET: TalentoHumano/AcreditacionesAcademicas/AgregarEditar
41     [HttpGet]
42     Oreferencias
43     public ActionResult AgregarEditar(byte id)...
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63     // POST: TalentoHumano/AcreditacionesAcademicas/AgregarEditar
64     // Para protegerse de ataques de publicación excesiva, habilite las propiedades específicas a las
65     // que desea enlazarse. Para obtener
66     // más información vea https://go.microsoft.com/fwlink/?LinkId=317598.
67     [HttpPost]
68     [ValidateAntiForgeryToken]
69     Oreferencias
70     public ActionResult AgregarEditar([Bind(Include =
71     : "Nombre,TipoAcademicoId,ClasificacionAcademicaId,Abreviatura")]
72     : AcreditacionAcademica aAcademica, short acreditacionAcademicaId)...
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164     // POST: TalentoHumano/AcreditacionesAcademicas/Inactivar
165     [HttpPost]
166     [ValidateAntiForgeryToken]
167     Oreferencias
168     public ActionResult Inactivar(short id)...
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199     // GET: TalentoHumano/AcreditacionesAcademicas/EditarOrden
200     [HttpGet]
201     Oreferencias
202     public ActionResult EditarOrden(byte id)...
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248     // POST: TalentoHumano/AcreditacionesAcademicas/EditarOrden
249     [HttpPost]
250     [ValidateAntiForgeryToken]
251     Oreferencias
252     public ActionResult EditarOrden(byte id, byte orden)...
```

3.3.4. Administrar Cargos

```
19 public class CargosController : Controller
20 {
21     private readonly Context _context = new Context();
22     private readonly ModelStateValid _ModelStateValid = new ModelStateValid();
23
24     // GET: TalentoHumano/Cargos
25     0 referencias
26     public ActionResult Index()...
27
28
29     // Obtener cargos
30     // GET: TalentoHumano/Cargos/ObtenerDatos
31     [HttpGet]
32     0 referencias
33     public ActionResult ObtenerDatos()...
34
35
36     // Buscar Cargo
37     // GET: TalentoHumano/Cargos/BuscarCargo
38     [HttpGet]
39     0 referencias
40     public JsonResult BuscarCargo(short id)...
41
42
43     private int _historialCargoId;
44
45     // POST: TalentoHumano/Cargos/CreateEdit
46     // Para protegerse de ataques de publicación excesiva, habilite las propiedades específicas a las
47     // que desea enlazarse. Para obtener
48     // más información vea https://go.microsoft.com/fwlink/?LinkId=317598.
49     [HttpPost]
50     [ValidateAntiForgeryToken]
51     0 referencias
52     public async Task<ActionResult> AgregarEditar(
53         [Bind(Include =
54             "Nombre,CargoClasificacionId,CargoCategoriaId,CargoSuperiorId,AnioExperiencia,AplicaSalarioMinimo,AplicaSa
55             larioVariable,SalarioBasico,Definicion")]
56         Cargo cargo, short cargoId, bool guardarHistorial, string motivo)...
57
58
59     //detalle del cargo
60     1 referencia
61     private void DetalleCargo(short cargoId)...
62
63
64     // POST: TalentoHumano/Cargos/Inactivar
65     [HttpPost]
66     [ValidateAntiForgeryToken]
67     0 referencias
68     public ActionResult Inactivar(short id)...
69
70
71     // GET: TalentoHumano/Cargos/EditarOrden
72     [HttpGet]
73     0 referencias
74     public ActionResult EditarOrden(short id)...
```

3.3.5. Administrar Departamento

```
11 public class DepartamentosController : Controller
12 {
13     private readonly Context _context = new Context();
14     private readonly ModelStateValid _ModelStateValid = new ModelStateValid();
15
16     // GET: Catalogo/Departamentos
17     // Referencias
18     public ActionResult Index()...
19
20
21
22
23
24
25
26
27
28     // Obtener registros
29     // GET: Catalogo/Departamentos/ObtenerDatos
30     [HttpGet]
31     // Referencias
32     public ActionResult ObtenerDatos(byte id)...
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47     // Editar registros
48     // GET: Catalogo/Departamentos/AgregarEditar
49     [HttpGet]
50     // Referencias
51     public JsonResult AgregarEditar(short id)...
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77     // Agregar o editar registro en la base de datos
78     // POST: Catalogo/Departamentos/CreateEdit
79     // Para protegerse de ataques de publicación excesiva, habilite las propiedades específicas a las
80     // que desea enlazarse. Para obtener
81     // más información vea https://go.microsoft.com/fwlink/?LinkId=317598.
82     [HttpPost]
83     [ValidateAntiForgeryToken]
84     // Referencias
85     public ActionResult AgregarEditar([Bind(Include = "Codigo,Nombre,PaisId")]
86     : Departamento departamento, short departamentoId)...
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117     // POST: Catalogo/Departamentos/Inactivar/5
118     [HttpPost]
119     [ValidateAntiForgeryToken]
120     // Referencias
121     public ActionResult Inactivar(short id)...
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226     // Editar orden
227     // GET: Catalogo/Departamentos/EditarOrden
228     [HttpGet]
229     // Referencias
230     public ActionResult EditarOrden(short id)...
```

3.3.6. Administrar Municipios

```
11 public class MunicipiosController : Controller
12 {
13     private readonly Context _context = new Context();
14     private readonly ModelStateValid _ModelStateValid = new ModelStateValid();
15
16     // GET: Catalogo/Municipios
17     // Referencias
18     public ActionResult Index()...
19
20
21
22
23     // Obtener registros
24     // GET: Catalogo/Municipios/ObtenerDatos
25     [HttpGet]
26     // Referencias
27     public ActionResult ObtenerDatos(byte paisId, short departamentoId)...
28
29
30
31
32     // Editar registros
33     // GET: Catalogo/Municipios/AgregarEditar
34     [HttpGet]
35     // Referencias
36     public JsonResult AgregarEditar(short id)...
37
38
39
40
41     // Agregar o editar registro en la base de datos
42     // POST: Catalogo/Municipios/AgregarEditar
43     // Para protegerse de ataques de publicación excesiva, habilite las propiedades específicas a las
44     // que desea enlazarse. Para obtener
45     // más información vea https://go.microsoft.com/fwlink/?LinkId=317598.
46     [HttpPost]
47     [ValidateAntiForgeryToken]
48     // Referencias
49     public ActionResult AgregarEditar([Bind(Include = "Codigo,Nombre,DepartamentoId")]
50     : Municipio municipio, short municipioId)...
51
52
53
54
55
56
57
58
59
60
61
62     // Eliminar registro
63     // POST: Catalogo/Municipios/Inactivar/5
64     [HttpPost]
65     [ValidateAntiForgeryToken]
66     // Referencias
67     public ActionResult Inactivar(short id)...
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
```

3.3.7. Administrar Dependencias

```
16 public class DependenciasController : Controller
17 {
18     private readonly Context _context = new Context();
19     private readonly ModelStateValid _ModelStateValid = new ModelStateValid();
20
21     // GET: TalentoHumano/Dependencias
22     // Referencias
23     public ActionResult Index()...
24
25     // Obtener Dependencias Areas Funcionales
26     // GET: TalentoHumano/Dependencias/ObtenerAreasFuncionales
27     [HttpGet]
28     // Referencias
29     public async Task<ActionResult> ObtenerAreasFuncionales()...
30
31     // GET: TalentoHumano/Dependencias/AgregarEditar
32     [HttpGet]
33     // Referencias
34     public async Task<ActionResult> AgregarEditar(short id)...
35
36     private int _historialDependenciaId;
37
38     // POST: TalentoHumano/Dependencias/AgregarEditar
39     [HttpPost]
40     [ValidateAntiForgeryToken]
41     // Referencias
42     public ActionResult AgregarEditar([Bind(Include =
43     "DependenciaSuperiorId,DependenciaGrupoId,DependenciaAmbitoId,Nombre,NivelAreaFuncional,Nominal,Abreviatura
44     Observacion")]
45     : Dependencia dependencia, short dependenciaId, bool guardarHistorial, string motivo)...
46
47     // POST: TalentoHumano/Dependencias/InactivarDependencia
48     [HttpPost]
49     [ValidateAntiForgeryToken]
50     // Referencias
51     public ActionResult InactivarDependencia(short id)...
52
53     [HttpGet]
54     // Referencias
55     public ActionResult EditarOrdenAreaFuncional(short id)...
56
57     // POST: TalentoHumano/Dependencias/EditarOrdenAreaFuncional
58     [HttpPost]
59     [ValidateAntiForgeryToken]
60     // Referencias
61     public ActionResult EditarOrdenAreaFuncional(short id, short orden)...
62
63     // GET: TalentoHumano/Dependencias/EditarOrdenDependencia
64     [HttpGet]
65     // Referencias
66     public ActionResult EditarOrdenDependencia(short id)...
```

3.3.8. Administrar Estudios Académicos

```
13 public class EstudiosAcademicosController : Controller
14 {
15     private readonly Context _context = new Context();
16     private readonly ModelStateValid _ModelStateValid = new ModelStateValid();
17
18     // GET: TalentoHumano/EstudiosAcademicos
19     // O referencias
20     public ActionResult Index()...
21
22     // Obtener registros
23     // GET: TalentoHumano/EstudiosAcademicos/ObtenerDatos
24     [HttpGet]
25     // O referencias
26     public ActionResult ObtenerDatos()...
27
28     // GET: TalentoHumano/EstudiosAcademicos/AgregarEditar
29     [HttpGet]
30     // O referencias
31     public JsonResult AgregarEditar(short id)...
32
33     // Agregar o editar registro en la base de datos
34     // POST: TalentoHumano/EstudiosAcademicos/AgregarEditar
35     // Para protegerse de ataques de publicación excesiva, habilite las propiedades específicas a las que desea enla
36     // más información vea https://go.microsoft.com/fwlink/?LinkId=317598.
37     [HttpPost]
38     [ValidateAntiForgeryToken]
39     // O referencias
40     public ActionResult AgregarEditar([Bind(Include = "Nombre,TipoAcreditacionAcademicaId,AcreditacionAcademicaId")]
41     : EstudioAcademico eAcademico, short estudioAcademicoId)...
42
43     // POST: TalentoHumano/EstudiosAcademicos/Inactivar/5
44     [HttpPost]
45     [ValidateAntiForgeryToken]
46     // O referencias
47     public ActionResult Inactivar(short id)...
48
49     // Editar orden
50     // GET: TalentoHumano/EstudiosAcademicos/EditarOrden
51     [HttpGet]
52     // O referencias
53     public ActionResult EditarOrden(short id)...
54
55     // POST: TalentoHumano/EstudiosAcademicos/EditarOrden
56     [HttpPost]
57     [ValidateAntiForgeryToken]
58     // O referencias
59     public ActionResult EditarOrden(short id, short orden)...
```

3.3.9. Administrar Grupo de Compensaciones y Deducciones

```
11 public class CompensacionesDeduccionesGruposController : Controller
12 {
13     private readonly Context _context = new Context();
14     private readonly ModelStateValid _ModelStateValid = new ModelStateValid();
15
16     // GET: TalentoHumano/CompensacionesDeduccionesGrupos
17     // Referencias
18     public ActionResult Index()...
19
20
21
22
23
24
25
26
27
28     // GET: TalentoHumano/CompensacionesDeduccionesGrupos/ObtenerDatos
29     [HttpGet]
30     // Referencias
31     public ActionResult ObtenerDatos()...
32
33
34
35
36
37
38
39
40
41
42
43     // Editar registros
44     // GET: TalentoHumano/CompensacionesDeduccionesGrupos/AgregarEditar
45     [HttpGet]
46     // Referencias
47     public JsonResult AgregarEditar(byte id)...
48
49
50
51
52
53
54
55
56
57
58
59
60
61     // POST: TalentoHumano/CompensacionesDeduccionesGrupos/AgregarEditar
62     [HttpPost]
63     [ValidateAntiForgeryToken]
64     // Referencias
65     public ActionResult AgregarEditar([Bind(Include =
66         "Codigo,Nombre,NombreCorto,Abreviatura,TipoId")] CompensacionDeducccionGrupo cDeducccionGrupo,
67         byte compensacionDeducccionGrupoId)...
68
69
70
71
72
73
74
75
76
77
78
79
80
81     // POST: TalentoHumano/CompensacionesDeduccionesGrupos/Inactivar
82     [HttpPost]
83     [ValidateAntiForgeryToken]
84     // Referencias
85     public ActionResult Inactivar(byte id)...
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101     // GET: TalentoHumano/CompensacionesDeduccionesGrupos/EditarOrden
102     [HttpGet]
103     // Referencias
104     public ActionResult EditarOrden(byte id)...
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131     // POST: TalentoHumano/CompensacionesDeduccionesGrupos/EditarOrden
132     [HttpPost]
133     [ValidateAntiForgeryToken]
134     // Referencias
135     public ActionResult EditarOrden(byte id, byte orden)...
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301     // Validar si el registro existe
302     // Get: TalentoHumano/CompensacionesDeduccionesGrupos/ValidarNombre
303     [HttpGet]
304     // Referencias
305     public JsonResult ValidarNombre(string text, byte id)...
```

3.3.10. Administrar Empresas

```
16 public class EmpresasController : Controller
17 {
18     private readonly Context _context = new Context();
19     private readonly ModelStateValid _modelStateValid = new ModelStateValid();
20
21     // GET: TalentoHumano/Empresas
22     // Referencias
23     public ActionResult Index()...
24
25
26
27
28     // GET: TalentoHumano/Empresas/ObtenerDatos
29     [HttpGet]
30     // Referencias
31     public ActionResult ObtenerDatos()...
32
33
34
35
36     // Agregar o editar registro
37     // GET: TalentoHumano/Empresas/AgregarEditar
38     [HttpGet]
39     // Referencias
40     public async Task<ActionResult> AgregarEditar(int id)...
41
42
43
44
45     // Agregar o editar registro en la base de datos ...
46     [HttpPost]
47     [ValidateAntiForgeryToken]
48     // Referencias
49     public async Task<ActionResult> AgregarEditar([Bind(Include =
50     "PaisId,DepartamentoId,MunicipioId,EmpresaClasificacionId,EmpresaTipoId,Nombre,Acronimo,Correo,Telefono,Distrito",
51     "Direccion,OfreceBien,OfreceServicio")]
52     Empresa empresa, int empresaId)...
53
54
55
56
57     // Eliminar Registro
58     // POST: TalentoHumano/Empresas/Inactivar
59     [HttpPost]
60     [ValidateAntiForgeryToken]
61     // Referencias
62     public ActionResult Inactivar(short id)...
63
64
65
66
67     // Valida si el registro existe
68     // Get: TalentoHumano/Empresas/ValidarNombre
69     [HttpGet]
70     // Referencias
71     public JsonResult ValidarNombre(string text, int id, byte paisId)...
```

3.3.11. Administrar Perfil del Empleado

```
20 public class EmpleadosController : Controller
21 {
22     private readonly Context _context = new Context();
23     private readonly ModelStateValid _ModelStateValid = new ModelStateValid();
24
25     // GET: TalentoHumano/Empleados
26     Oreferencias
27     public async Task<ActionResult> Index()...
28
29     // Mostrar información personal del empleado ...
30     [HttpGet]
31     Oreferencias
32     public async Task<ActionResult> InformacionPersonal(string id)...
33
34     // Obtener identificacion del empleado ...
35     [HttpGet]
36     Oreferencias
37     public async Task<JsonResult> ObtenerEmpleadosTiposIdentificaciones(int id)...
38
39     [HttpPost]
40     Oreferencias
41     public ActionResult FilesUpload(HttpPostedFileBase file)...
42
43     private int _historialEmpleadoId;
44
45     // POST: TalentoHumano/Empleados/AgregarEditarInformacionPersonal ...
46     [HttpPost]
47     [ValidateAntiForgeryToken]
48     Oreferencias
49     public async Task<ActionResult> AgregarEditarInformacionPersonal(
50         [Bind(Include =
```

```
490     // POST: TalentoHumano/Empleados/GenerarCodigoEmpleado
491     [HttpPost]
492     [ValidateAntiForgeryToken]
493     Oreferencias
494     public ActionResult GenerarCodigoEmpleado(int empleadoId)...
495
496     // GET: TalentoHumano/Empleados/EstudiosAcademicos
497     [HttpGet]
498     Oreferencias
499     public async Task<ActionResult> EstudiosAcademicos(string id)...
500
501     // GET: TalentoHumano/Empleados/ExperienciasLaborales
502     [HttpGet]
503     Oreferencias
504     public async Task<ActionResult> ExperienciasLaborales(string id)...
505
506     // GET: TalentoHumano/Empleados/Familiares
507     [HttpGet]
508     Oreferencias
509     public async Task<ActionResult> Familiares(string id)...
```

3.3.12. Administrar Contratos del Empleado

```
20 public class EmpleadosContratosController : Controller
21 {
22     private readonly Context _context = new Context();
23     private readonly ModelStateValid _ModelStateValid = new ModelStateValid();
24
25     // GET: TalentoHumano/EmpleadosContratos
26     0 referencias
27     public async Task<ActionResult> Index()...
28
29     // Obtener datos
30     // Get: TalentoHumano/EmpleadosContratos/ObtenerContratos
31     [HttpGet]
32     0 referencias
33     public async Task<JsonResult> ObtenerContratos(int id)...
34
35     // GET: TalentoHumano/EmpleadosContratos/AgregarEditarContrato
36     [HttpGet]
37     0 referencias
38     public async Task<ActionResult> AgregarEditarContrato(int empleadoContratoId, byte? tipoContratoId)...
39
40     // POST: TalentoHumano/EmpleadosContratos/AgregarEditarContrato
41     [HttpPost]
42     1 referencia
43     public async Task<ActionResult> AgregarEditarContrato(
44         [Bind(Include =
45             Codes.Consultas.EmpleadoContrato.StrParametroAgregarEditarEmpleadoContrato)]
46         EmpleadoContrato empleadoContrato, int empleadoContratoId, bool guardarHistorial, string motivo)...
47
48     // Obtener tipos de contratos
49     // Get: TalentoHumano/EmpleadosContratos/ObtenerBancos
50     [HttpGet]
51     0 referencias
52     public async Task<JsonResult> ObtenerBancos(int id)...
53
54     // Obtener datos del Cargo
55     // Get: TalentoHumano/EmpleadosContratos/ObtenerDatosCargo
56     [HttpGet]
57     0 referencias
58     public async Task<JsonResult> ObtenerDatosCargo(short id)
59     {
60         var data = await Task.Run(() => Cargo.BuscarCargo(id));
61         return Json(new { data, exito = true }, JsonRequestBehavior.AllowGet);
62     }
63 }
```

3.3.13. Generar Nomina

```
18 public class NominasPeriodosController : Controller
19 {
20     private readonly Context _context = new Context();
21
22     // GET: TalentoHumano/NominasPeriodos
23     // 0 referencias
24     public async Task<ActionResult> Index()...
25
26     // Obtener periodos
27     // GET: TalentoHumano/NominaPeriodos/ObtenerNominaPeriodos
28     [HttpGet]
29     // 0 referencias
30     public async Task<ActionResult> ObtenerNominaPeriodos(byte tipoNominaId, int anio)...
31
32     // Obtener comprobantes
33     // GET: TalentoHumano/NominaPeriodos/ObtenerNominas
34     [HttpGet]
35     // 0 referencias
36     public async Task<ActionResult> ObtenerNominas(int nominaPeriodoId)...
37
38     // POST: TalentoHumano/NominasPeriodos/GenerarPeriodos ...
39     [HttpPost]
40     [ValidateAntiForgeryToken]
41     // 0 referencias
42     public ActionResult GenerarPeriodos(byte tipoNomina, int anio)...
43
44     private const string StrCampoContratoFijo =
45     "EmpleadoContratoId, HistorialEmpleadoContratoId, EmpleadoId,Codigo, SalarioBase, OrdenAreaFuncional, OrdenEmpleado, Or
46     denContrato";
47
48     private const string StrCampoEmpleadoCompensacionDeducccion =
49     "EmpleadoCompensacionDeducccionId, EmpleadoId, CompensacionDeducccionId, HistorialCompensacionDeducccionId, Valor, Total,
50     AplicaRetencion, AplicaCompensacion, Calculado, AgregarComprobante, OrdenCompensacionDeducccion, Porcentual";
51
52     // POST: TalentoHumano/NominasPeriodos/GenerarPeriodos
53     // Para protegerse de ataques de publicación excesiva, habilite las propiedades específicas a las que
54     // desea enlazarse. Para obtener
55     // más información vea https://go.microsoft.com/fwlink/?LinkId=317598.
56     [HttpPost]
57     [ValidateAntiForgeryToken]
58     // 0 referencias
59     public async Task<ActionResult> ProcesarNomina(int nominaPeriodoId)...
```

```

150 public async Task<ActionResult> ProcesarNomina(int nominaPeriodoId)...
594
595 // POST: TalentoHumano/NominasPeriodos/RecalcularNomina
596 [HttpPost]
597 [ValidateAntiForgeryToken]
598 0 referencias
598 public JsonResult RecalcularNomina(int nominaComprobanteId)...
1006
1007 // POST: TalentoHumano/NominasPeriodos/AgregarEditar
1008 // Para protegerse de ataques de publicación excesiva, habilite las propiedades específicas a las que
1009 // desea enlazarse. Para obtener
1009 // más información vea https://go.microsoft.com/fwlink/?LinkId=317598.
1010 [HttpPost]
1011 [ValidateAntiForgeryToken]
1012 0 referencias
1012 public ActionResult AbrirPeriodo([Bind(Include = "MesAbierto")] NominaPeriodo nominaPeriodo, int
1012 | nominaPeriodoId)...
1038
1039 // GET: TalentoHumano/NominasPeriodos/ModalContent
1040 [HttpGet]
1041 0 referencias
1041 public ActionResult ModalContent()...
1045
1046 // GET: TalentoHumano/NominasPeriodos/ModalMotivoAnulacion
1047 [HttpGet]
1048 0 referencias
1048 public ActionResult ModalMotivoAnulacion()...
1059
1060 //Anular comprobante de pago
1061 // POST: TalentoHumano/NominasPeriodos/AnularComprobantePago
1062 [HttpPost]
1063 [ValidateAntiForgeryToken]
1063 0 referencias
1064 public JsonResult AnularNomina(int nominaComprobanteId, string motivoAnulo)...

```

3.4. IMPLEMENTACIÓN DE PATRONES DE DISEÑO

Aplicación del patrón de diseño MVC.

Se utilizó el patrón de diseño MVC el cual divide la aplicación en tres módulos: modelo, vistas y controlador. Esto va permitir tener la información actualizada siempre, las modificaciones a las vistas no afectaran los otros módulos, entre otros beneficios que nos ofrece este patrón de diseño.

3.4.1. Modelos: Los modelos están contenidos en la carpeta Models seccionados en dos carpetas: Administración y Talento Humano.

A continuación, se muestra la estructura del Modelo:

Tabla 4 : Estructura del modelo.

Administración	Talento Humano
▷ C# AccesoUsuarioControlador.cs	▷ C# AcreditacionAcademica.cs
▷ C# AccesoUsuarioReporte.cs	▷ C# Banco.cs
▷ C# Bitacora.cs	▷ C# Barrio.cs
▷ C# Controlador.cs	▷ C# Calendario.cs
▷ C# Menu.cs	▷ C# Cargo.cs
▷ C# Reporte.cs	▷ C# CargoCategoria.cs
▷ C# TipoAccion.cs	▷ C# CargoEstudioAcademico.cs
▷ C# Usuario.cs	▷ C# CargoFuncion.cs
	▷ C# CargoHabilidades.cs
	▷ C# CargoHistorial.cs
	▷ C# CategoriaLicenciaConducir.cs
	▷ C# ClasificacionAcademica.cs
	▷ C# Departamento.cs
	▷ C# Dependencia.cs
	▷ C# Distritio.cs
	▷ C# Empresa.cs
	▷ C# EmpresaClasificacion.cs
	▷ C# EmpresaGiro.cs
	▷ C# EmpresaTipo.cs
	▷ C# EstadoCivil.cs
	▷ C# EstudioAcademico.cs
	▷ C# HistorialCargo.cs
	▷ C# Mes.cs
	▷ C# Moneda.cs

Fuente: Elaboración propia.

3.4.2. Vista: Para el desarrollo de las vistas cabe mencionar que la programación de las hojas de estilos en cascada (CSS) y del código JavaScript se ha separado del lenguaje de etiquetado (HTML), y organizado de tal forma que al mostrar el contenido de un módulo únicamente se descargue los scripts correspondientes.

Para cada controlador se incluye una carpeta que contiene las vistas correspondientes (código HTML, CSS, JS). Las vistas están seccionadas en áreas: Administración y Talento Humano.

A continuación, se muestra la estructura de las vistas:

Tabla 5 : Estructura de las vistas.

Administración	Talento Humano
<ul style="list-style-type: none"> ▲ Areas <ul style="list-style-type: none"> ▲ Administracion <ul style="list-style-type: none"> ▸ Controllers ▲ Views <ul style="list-style-type: none"> ▸ AccesosUsuariosMenu ▸ AccesosUsuariosReportes ▸ Menus ▸ Reportes ▸ Usuarios [@] _ViewStart.cshtml web.config ▸ C# AdministracionAreaRegistration.cs 	<ul style="list-style-type: none"> ▲ Areas <ul style="list-style-type: none"> ▸ Administracion ▲ TalentoHumano <ul style="list-style-type: none"> ▸ Controllers ▲ Views <ul style="list-style-type: none"> ▸ AcreditacionesAcademicas ▸ Barrios ▸ CargoCategorias ▸ Cargos ▸ ClasificacionesAcademicas ▸ CompensacionesDeducciones ▸ CompensacionesDeduccionesGrupos ▸ CompensacionesDeduccionesImportadas ▸ Departamentos ▸ Dependencias ▸ Empleados ▸ EmpleadosBeneficiarios ▸ EmpleadosCompensacionesDeducciones ▸ EmpleadosContratos ▸ EmpleadosEstudiosAcademicos

Fuente: Elaboración propia.

3.4.3. Controlador: Desde la clase controlador se gestionan todas las acciones correspondientes a cada módulo, para las operaciones genéricas y consultas a la base de datos se consideró separar por medio de clases públicas, en el cual se creó una carpeta llamada Clases Publicas.

A continuación, se muestra la estructura del Controlador:

Tabla 6 : Estructura del controlador.

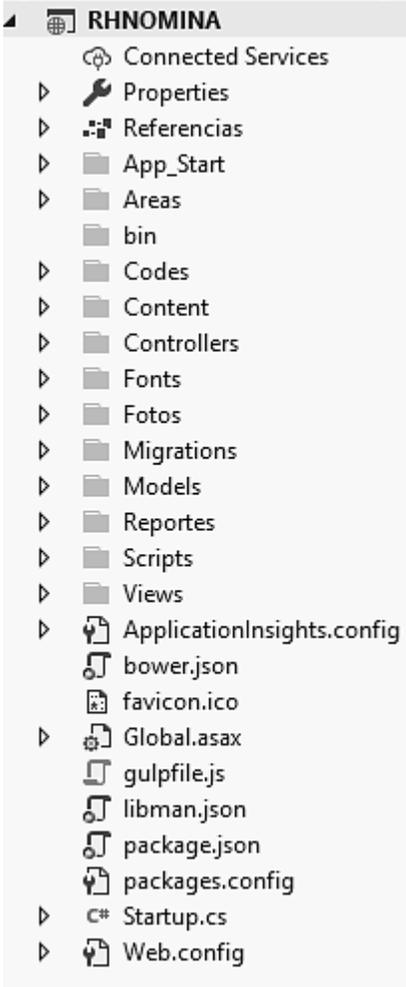
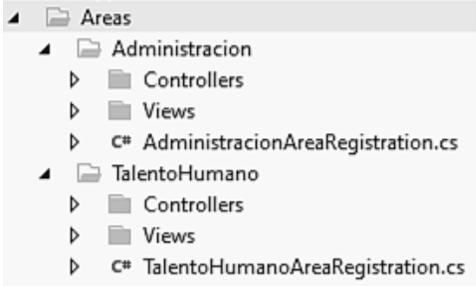
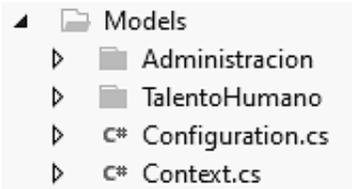
Administración	Talento Humano
<ul style="list-style-type: none"> ▲ Areas <ul style="list-style-type: none"> ▲ Administracion <ul style="list-style-type: none"> ▲ Controllers <ul style="list-style-type: none"> ▶ C# AccesosUsuariosMenuController.cs ▶ C# AccesosUsuariosReportesController.cs ▶ C# AgrupadosReportesController.cs ▶ C# GenerarReportesController.cs ▶ C# MenusController.cs ▶ C# ReportesController.cs ▶ C# UsuariosController.cs 	<ul style="list-style-type: none"> ▲ Areas <ul style="list-style-type: none"> ▶ Administracion ▲ TalentoHumano <ul style="list-style-type: none"> ▲ Controllers <ul style="list-style-type: none"> ▶ C# AcreditacionesAcademicasController.cs ▶ C# BarriosController.cs ▶ C# CargoCategoriasController.cs ▶ C# CargosController.cs ▶ C# ClasificacionesAcademicasController.cs ▶ C# CompensacionesDeduccionesController.cs ▶ C# CompensacionesDeduccionesGruposController.cs ▶ C# CompensacionesDeduccionesImportadasController.cs ▶ C# DepartamentosController.cs ▶ C# DependenciasController.cs

Fuente: Elaboración propia.

3.5. ORGANIZACIÓN DE LOS ARCHIVOS DEL PROYECTO

A continuación, se muestra la estructura de donde y como están almacenados los archivos del sistema como son: los controladores, imágenes, fuentes de letras, reportes, las vistas de cada uno de los módulos, entre otros.

Tabla 7 : Organización de archivos del proyecto.

Principal	Áreas
	<p>En la carpeta de áreas se integra los controladores y las vistas (páginas de presentación) de cada módulo.</p> 
	Modelos
	<p>En la carpeta Modelos se agregan las clases correspondiente al modelado de la base de datos.</p> 

Fuente: Elaboración propia.

3.6. DISEÑO DE LA INTERFAZ DEL SISTEMA

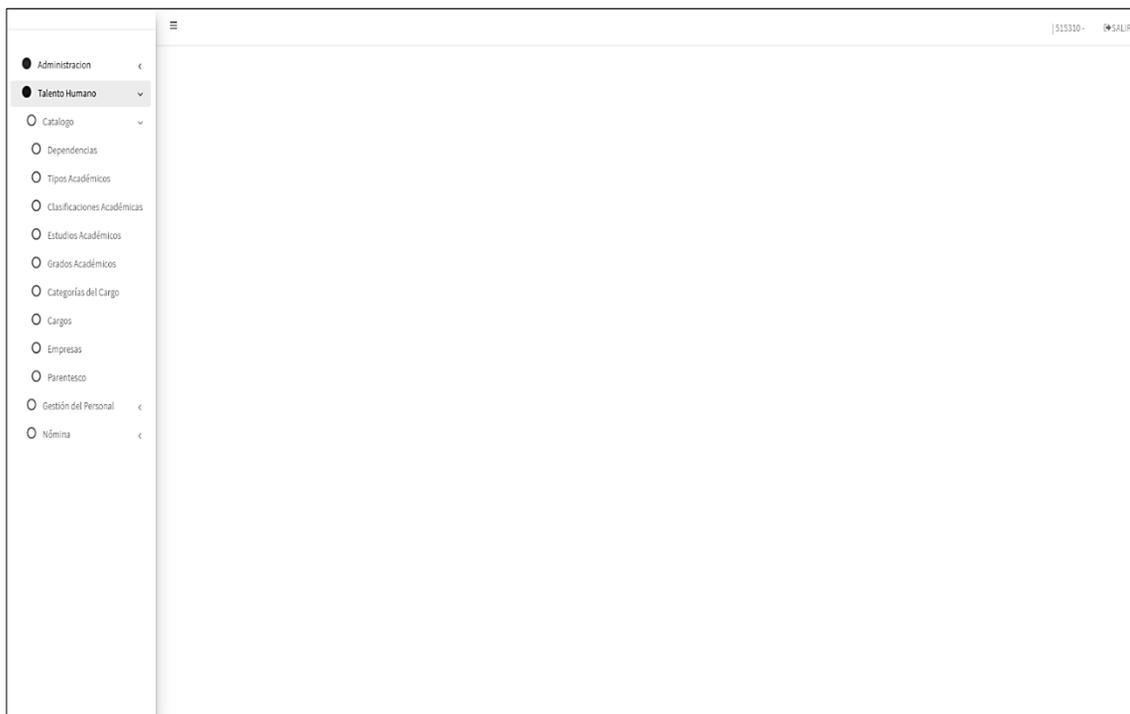
En la interfaz de usuario se incluyó un diseño fácil de utilizar, responsive que se pueda visualizar desde cualquier dispositivo. Se utilizaron colores básicos de la empresa más otros colores alternativos para los errores y otros para las notificaciones. Se incluyó también logo institucional en las pantallas.

LOGIN



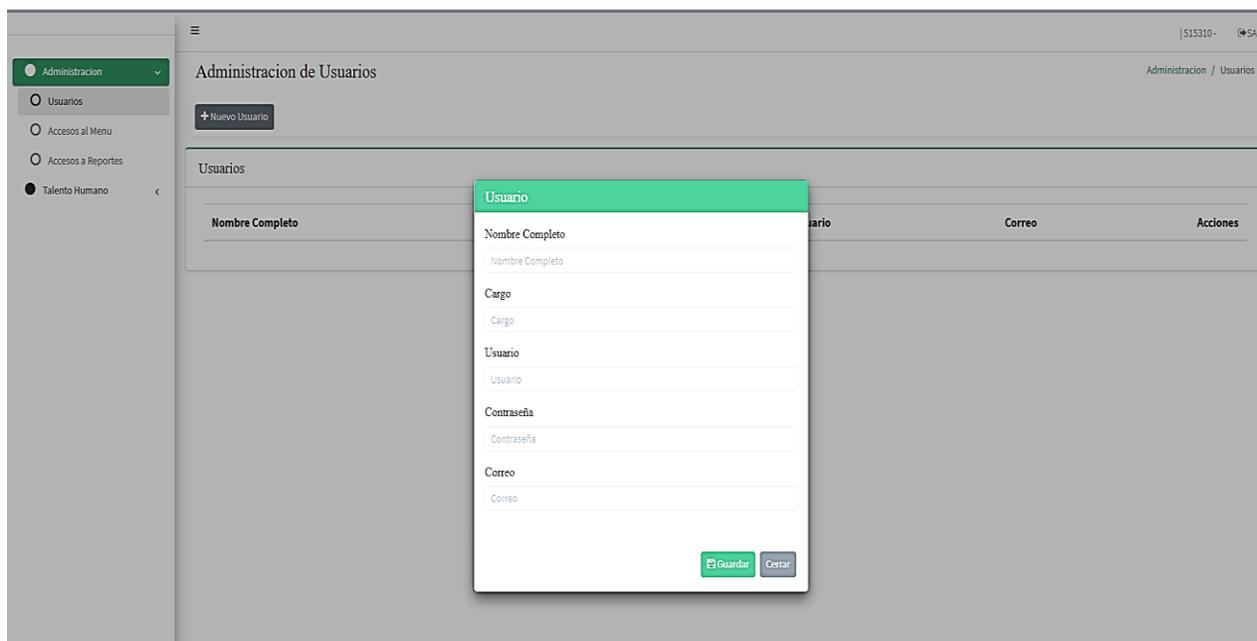
Descripción:	Venta de autenticación de usuario del sistema
Acciones / eventos:	Inicio de sesión de usuario
Elementos gráficos que la componen:	Título de ventana, logo institucional, campo de recepción del usuario y de la contraseña con su imagen representativa, botón de "Ingresar".
Comparación con los modelos de presentación:	Acorde al modelo propuesto.

INICIO



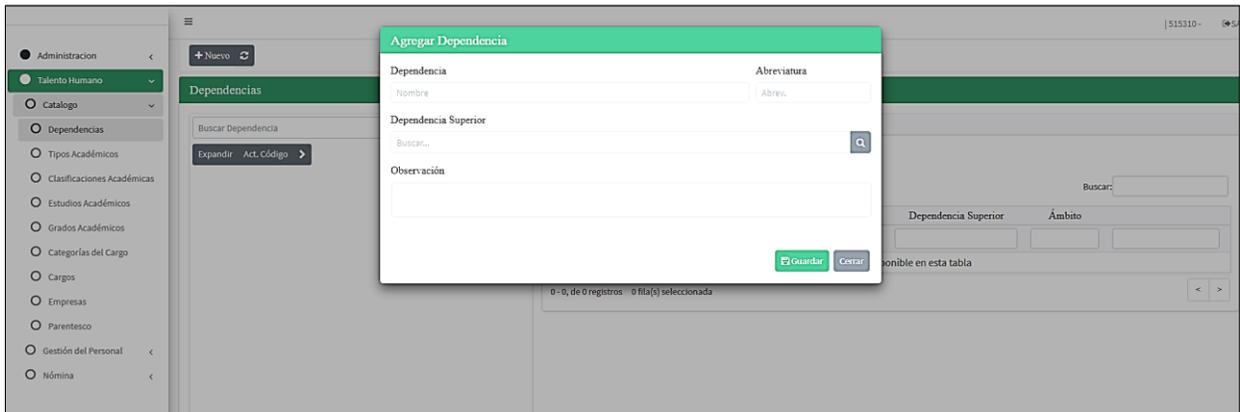
Descripción:	Ventana de inicio de trabajo en sistema, panel de selección en base al perfil de usuario.
Acciones / eventos:	Ingreso a opciones de trabajo según perfil de usuario.
Elementos gráficos que la componen:	Casillas de selección según área seleccionada, botones de "selección".
Comparación con los modelos de presentación:	Acorde al modelo propuesto.

Administrar Usuarios



Descripción:	Ventana de administración de los nuevos usuarios del sistema.
Acciones / eventos:	Agregar usuarios nuevos al sistema.
Elementos gráficos que la componen:	Título de ventana, campos de recepción de datos del usuario como son: nombre completo, cargo, usuario, contraseña y correo. También botón de “cerrar” y “guardar”.
Comparación con los modelos de presentación:	Acorde al modelo propuesto.

Administrar Dependencias



Descripción:	Venta donde se administran las áreas de la empresa según el organigrama de la misma, con su respectiva abreviación e instancia superior.
Acciones / eventos:	Agregar dependencias.
Elementos gráficos que la componen:	Título de ventana, campos de recepción de datos como son: Dependencia, Abreviatura, Dependencia superior, Observación, botones “Guardar” y “Cerrar”.
Comparación con los modelos de presentación:	Acorde al modelo propuesto.

Administrar Tipos Académicos

Descripción:	Ventana donde se agregan los títulos académicos de los empleados.
Acciones / eventos:	Administrar los títulos académicos de los empleados.
Elementos gráficos que la componen:	Título de ventana, campos de recepción de datos Tipo académico y Buscar. También tabla de resultado y botón “Guardar”.
Comparación con los modelos de presentación:	Acorde al modelo propuesto.

Administrar Clasificación Académica

The screenshot displays a web application interface for managing academic classifications. On the left is a sidebar menu with options like 'Administración', 'Talentos Humanos', 'Catalogo', 'Dependencias', 'Tipos Académicos', 'Clasificaciones Académicas', 'Estudios Académicos', 'Grados Académicos', 'Categorías del Cargo', 'Cargos', 'Empresas', 'Parentesco', 'Gestión del Personal', and 'Nómina'. The main area is titled 'Clasificaciones Académicas' and contains a form with a 'Clasificación Académica*' label, an input field, and a 'Guardar' button. Below the form is a table with a search bar and a message: 'No se encontró nada - UML. 0 fila(s) seleccionada'.

Descripción:	Ventana donde se administran los diferentes niveles de escolaridad de los empleados.
Acciones / eventos:	Administrar niveles de escolaridad de los empleados.
Elementos gráficos que la componen:	Título de ventana, campos de recepción de datos Clasificación académica y Buscar. También tabla de resultado y botón “Guardar”.
Comparación con los modelos de presentación:	Acorde al modelo propuesto.

Administrar Parentesco

Descripción:	Ventana donde se agregan y editan toda la información de los familiares que cada empleado proporciona al momento de su contratación.
Acciones / eventos:	Administrar los datos de los familiares de los empleados.
Elementos gráficos que la componen:	Título de ventana, campos de recepción de datos Clasificación académica y Buscar. También tabla de resultado y botón “Guardar”.
Comparación con los modelos de presentación:	Acorde al modelo propuesto.

Administrar Departamentos

Descripción:	Ventana donde se registra y actualiza la información de ubicación de permanencia del empleado.
Acciones / eventos:	Administrar los datos de ubicación de los empleados.
Elementos gráficos que la componen:	Título de ventana, campos de recepción de datos como: Código, Departamento, País, Selecciones por país, Buscar. También tabla de resultado y botón de “Guardar” y “Buscar”.
Comparación con los modelos de presentación:	Acorde al modelo propuesto.

VII CONCLUSIONES

Una vez finalizado el presente trabajo que consistió en la automatización de la generación de la nómina para la empresa Distribuidores Generales S, A. y habiendo cumplido con los objetivos planteados, se pudieron precisar las siguientes conclusiones:

Se analizó la situación actual y el modelo de negocios de la empresa por medio del cual se delimitó la aplicación del presente estudio específicamente al área de recursos humanos en la automatización de la generación de la nómina y se logró identificar los procesos principales del mismo, determinando de esta forma 23 requerimientos funcionales y 6 no funcionales, los cuales fueron la base para el diseño del sistema y el modelo se basó en la metodología UWE.

Se realizaron estudios preliminares iniciando por el estudio del modelo de negocios los cuales permitieron conocer los procesos principales de la empresa con el fin de obtener los requerimientos funcionales y no funcionales para lograr todos los objetivos planteados.

Se desarrolló el sistema planteado haciendo uso de las tecnologías propuestas en base al modelo de diseño y se consideraron elementos gráficos que ofrecen al usuario una experiencia agradable y dan al sistema usabilidad. Se procuró que el sistema ofrezca la mejor experiencia al usuario haciéndolo responsivo y flexible a los diferentes medios donde podrá ser usado; todo esto permitirá la implementación de nuevos módulos a futuro.

El sistema de información ofrece una mejor gestión en el proceso de elaboración de la nómina mejorando los tiempos de respuesta y disminuyendo los errores frecuentes, lo cual a su vez se ve reflejado en los servicios brindados a sus colaboradores, incrementando sus niveles de calidad y mostrando una imagen corporativa en el mercado.

VIII RECOMENDACIONES

En base a los resultados obtenidos de este trabajo monográfico que se enfocó en el proceso de la generación quincenal de la nómina para el departamento de Gestión Humana de la empresa “Distribuidora General S.A.” (DIGESA), se consideran pertinentes las siguientes recomendaciones:

- ✓ Que el departamento de gestión humana y todos los involucrados utilice el sistema de información con el fin de cumplir los objetivos del sistema planteados, como reducir tiempos de gestión en la generación de la nómina.
- ✓ Dar continuidad al sistema, aportando nuevos módulos como es integrar las liquidaciones salariales, el control de vacaciones, liquidación de comisiones, contrataciones por servicios profesionales, entre otros; debido a que el sistema realizado solo abarca meramente la realización de la nómina quincenal, generación del contrato laboral, gestión del personal y la generación de reportes administrativos.
- ✓ Para mejorar el funcionamiento del sistema es necesarios que la empresa establezca políticas internas para todos los usuarios del sistema, donde se oriente que tienen que retroalimentar periódicamente respecto al funcionamiento del sistema.
- ✓ Evaluar otros departamentos que conforman la empresa y que no fueron el foco principal en este estudio.

IX BIBLIOGRAFÍA

- Anderson, R. (2016). *ASP.NET MVC 6 Documentation*.
- Arias, Á. (2015). *Aprende sobre la Ingeniería de Software* (Vol. 2da Edición). CreateSpace Independent Publishing Platform.
- Ceballos Sierra, J. (2014). *Microsoft C#. Curso de Programación. 2ª edición*. RA-MA.
- Chiavenato, I. (2009). *Administración de Recursos Humanos- El capital humano de las organizaciones*. MC Graw Hill Educacion.
- CSS mozilla. (8 de ene de 2023). <https://developer.mozilla.org/es/docs/Web/CSS>. mozilla: <https://developer.mozilla.org/es/docs/Web/CSS>
- DGI. (01 de 01 de 2013). *Dirección General de Ingresos*. Dirección General de Ingresos: https://www.dgi.gob.ni/FAQ/index.html?periodo_fiscal_y_tarifa_del_im.htm
- DIGESA. (2023). *REGLAMENTO INTERNO DE LA EMPRESA*.
- DOM Mozilla. (29 de nov de 2022). <https://developer.mozilla.org/es/docs/Glossary/DOM>. <https://developer.mozilla.org/es/docs/Glossary/DOM>: <https://developer.mozilla.org/es/docs/Glossary/DOM>
- Engineering, R. U. (10 de 08 de 2016). *UWE – UML-based Web Engineering*. UWE – UML-based Web Engineering: <https://uwe.pst.ifi.lmu.de/teachingTutorialRequirementsSpanish.html>
- Engineering, R. U. (10 de 08 de 2016). *UWE – UML-based Web Engineering*. UWE – UML-based Web Engineering: <https://uwe.pst.ifi.lmu.de/teachingTutorialContentSpanish.html>
- Freund, J., & Rucker, B. (2014). *BPMN 2.0 Manual de Referencia y Guía Práctica*. Santiago de Chile.
- Gutiérrez González, Á., & López Goytia, J. (2017). *DESARROLLO Y PROGRAMACION EN ENTORNOS WEB*. MÉXICO: ALFAOMEGA.
- html mozilla org. (6 de Dec de 2022). developer.mozilla.org. HTML: <https://developer.mozilla.org/es/docs/Glossary/HTML>

javascript mozilla. (24 de enero de 2023).

<https://developer.mozilla.org/es/docs/Web/JavaScript>.

<https://developer.mozilla.org/es/docs/Web/JavaScript>:

<https://developer.mozilla.org/es/docs/Web/JavaScript>

javascript Mozilla MDN. (25 de nov de 2022).

<https://developer.mozilla.org/es/docs/Web/Guide/AJAX>.

<https://developer.mozilla.org/es/docs/Web/Guide/AJAX>:

<https://developer.mozilla.org/es/docs/Web/Guide/AJAX>

Jquery.com. (2023). <https://jquery.com/>. <https://jquery.com/>: <https://jquery.com/>

KENDALL, K., & KENDALL, J. (2011). *ANÁLISIS Y DISEÑO DE SISTEMAS* (8va Edición ed.). MÉXICO: MC GRAW-HILL.

Nieves-Guerrero, C. G., Ucán-Pech, J. P., & Menéndez-Domínguez, V. H. (2014). UWE en Sistema de Recomendación de Objetos de Aprendizaje. *Revista Latinoamericana de Ingeniería de Software*, 137.

Ojeda, j. (12 de marzo de 2015). *Arquitectura de software en aspnet MVC*.

[msaspnetmvc.blogspot.com: msaspnetmvc.blogspot.com/2015/03/la-arquitectura-de-mis-proyectos-mvc.html](https://msaspnetmvc.blogspot.com/2015/03/la-arquitectura-de-mis-proyectos-mvc.html)

Rigoberto Navas S. (2016). *Código del trabajo y procesal laboral*. Managua: Editorial Juridica.

Senso., J. (septiembre de 2016). *blog.ugr.es*. <https://blogs.ugr.es/tecweb/los-sistemas-informacion-basados-la-web/>

Simona Bernardi, L. D. (2020). *Sistemas de información para la dirección* (2da ed.). España: Centro Universitario de la Defensa.

Sommerville, I. (2011). *INGENIERIA DE SOFTWARE*. México: Pearson Educacion.

Souza, k. J., & Oz, E. (2016). *Administración de los Sistemas de Información*. México: Cengage.

sql Universidad Europea. (04 de marzo de 2022). <https://universidadeuropea.com/>.

<https://universidadeuropea.com/>:

<https://universidadeuropea.com/blog/lenguaje-programacion-sql/>

Stair, R., & Reynolds, G. (2015). *PRINCIPIOS DE SISTEMAS DE INFORMACION*. México: Cengage.

Standards Development organization (OMG). (diciembre de 2010). *Business Process Model and Notation*. Business Process Model And Notation:

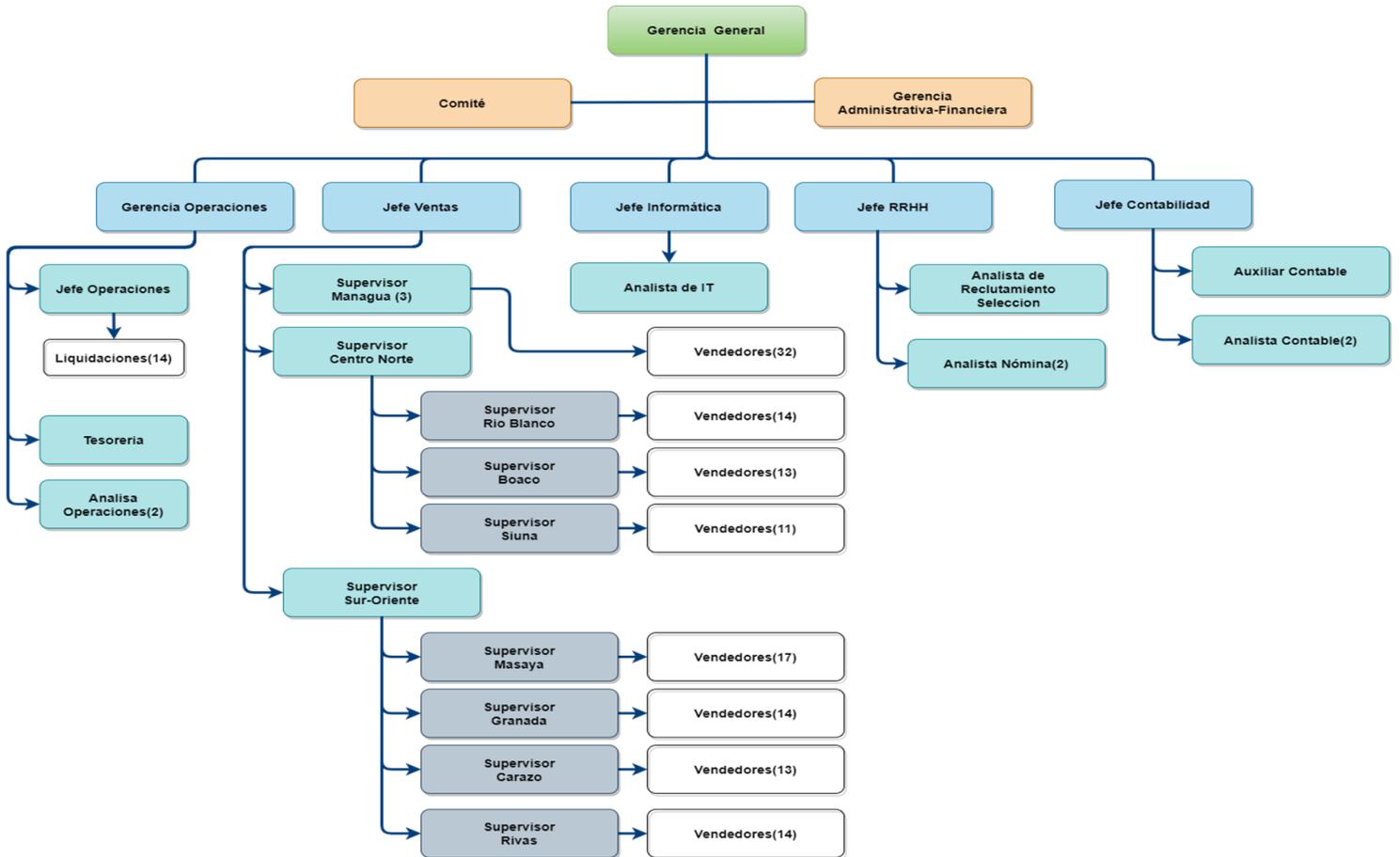
<https://www.omg.org/spec/BPMN/2.0>

Whitten, J. L., & Bentley, L. D. (2008). *ANÁLISIS DE SISTEMAS :DISEÑO Y MÉTODOS*. México: Mc Graw-Hill.

Whitten, J., & Bentley, L. (2008). *Analisis de Sistemas y Diseño de Metodos* (Septima edición ed.). Mexico, Mexico: Pearson Educación.

X ANEXOS

Anexo 1 : Estructura organizacional de la empresa.



Fuente: Elaboración propia.

Anexo 2 : Tabulación de requerimientos.

RF-0003 - Administrar perfil del empleado.

FRQ-0003	Administrar perfil del empleado
Versión	1.0 (19/04/2023)
Autores	<ul style="list-style-type: none">• <u>Eddy Sarantes</u>• <u>Harold Solis</u>• <u>Jeferson Cano</u>
Fuentes	<ul style="list-style-type: none">• <u>Especialista de selección y contratación</u>
Dependencias	<ul style="list-style-type: none">• <u>[OBJ-0008] Crear Imagen Corporativa de la Empresa</u>• <u>[OBJ-0001] Generar la nómina</u>• <u>[OBJ-0005] Generación de reportes gerenciales para análisis y toma de decisión</u>• <u>[OBJ-0003] Generación de información administrativa y contable</u>• <u>[OBJ-0002] Control de deducciones</u>
Descripción	El sistema deberá <i>registrar y actualizan los datos personales, experiencia laboral, familiares, beneficiarios.</i>
Importancia	vital
Urgencia	inmediatamente
Estado	en construcción
Estabilidad	media
Comentarios	Ninguno

Fuente: Elaboración propia.

RF-0004 - Administrar dependencias.

FRQ-0004	Administrar dependencias
Versión	1.0 (19/04/2023)
Autores	<ul style="list-style-type: none">• <u>Responsable TI</u>
Fuentes	<ul style="list-style-type: none">• <u>Especialista de selección y contratación</u>
Dependencias	<ul style="list-style-type: none">• <u>[OBJ-0003] Generación de información administrativa y contable</u>• <u>[OBJ-0001] Generar la nómina</u>
Descripción	El sistema deberá <i>permitir configurar cada una de las dependencias o areas segun el organigrama de la empresa.</i>
Importancia	importante
Urgencia	hay presión
Estado	en construcción
Estabilidad	media
Comentarios	Ninguno

Fuente: Elaboración propia.

RF-0005 - Administrar tipos académicos.

FRQ-0005	Administrar tipos académicos
Versión	1.0 (20/04/2023)
Autores	<ul style="list-style-type: none"> • <u>Eddy Sarantes</u> • <u>Harold Solis</u> • <u>Jeferson Cano</u>
Fuentes	<ul style="list-style-type: none"> • <u>Especialista de selección y contratación</u>
Dependencias	<ul style="list-style-type: none"> • <u>[OBJ-0005] Generación de reportes gerenciales para análisis y toma de decisión</u>
Descripción	El sistema deberá <i>de permitir agregar y actualizar los títulos académicos de los empleados.</i>
Importancia	vital
Urgencia	puede esperar
Estado	en construcción
Estabilidad	media
Comentarios	Ninguno

Fuente: Elaboración propia.

RF-0006 - Administración sucursales.

FRQ-0006	Administrar sucursales
Versión	1.0 (20/04/2023)
Autores	<ul style="list-style-type: none"> • <u>Eddy Sarantes</u> • <u>Harold Solis</u> • <u>Jeferson Cano</u>
Fuentes	<ul style="list-style-type: none"> • <u>Especialista de selección y contratación</u>
Dependencias	<ul style="list-style-type: none"> • <u>[OBJ-0005] Generación de reportes gerenciales para análisis y toma de decisión</u> • <u>[OBJ-0003] Generación de información administrativa y contable</u>
Descripción	El sistema deberá <i>agrupar cada uno de los empleados en sus diferentes sucursales.</i>
Importancia	vital
Urgencia	inmediatamente
Estado	en construcción
Estabilidad	media
Comentarios	Ninguno

Fuente: Elaboración propia.

RF-0007 - Administrar clasificaciones académicas.

FRQ-0007	Administrar clasificaciones académicas
Versión	1.0 (20/04/2023)
Autores	<ul style="list-style-type: none"> • <u>Eddy Sarantes</u> • <u>Harold Solis</u> • <u>Jeferson Cano</u>
Fuentes	• <u>Especialista de selección y contratación</u>
Dependencias	• <u>[OBJ-0003] Generación de información administrativa y contable</u>
Descripción	El sistema deberá <i>administrar los diferentes niveles de escolaridad de los empleados.</i>
Importancia	quedaría bien
Urgencia	inmediatamente
Estado	en construcción
Estabilidad	baja
Comentarios	Ninguno

Fuente: Elaboración propia.

RF-0008 - Administrar acreditaciones académicas.

FRQ-0008	Administrar acreditaciones académicas
Versión	1.0 (20/04/2023)
Autores	<ul style="list-style-type: none"> • <u>Eddy Sarantes</u> • <u>Harold Solis</u> • <u>Jeferson Cano</u>
Fuentes	• <u>Especialista de selección y contratación</u>
Dependencias	• <u>[OBJ-0003] Generación de información administrativa y contable</u>
Descripción	El sistema deberá <i>definir el grado académico que alcanzó el empleado o la persona a la vacante en la empresa.</i>
Importancia	vital
Urgencia	puede esperar
Estado	en construcción
Estabilidad	media
Comentarios	Ninguno

Fuente: Elaboración propia.

RF-0009 - Administrar cargos.

FRQ-0009	Administrar cargos
Versión	1.0 (20/04/2023)
Autores	<ul style="list-style-type: none"> • <u>Eddy Sarantes</u> • <u>Harold Solis</u> • <u>Jeferson Cano</u>
Fuentes	<ul style="list-style-type: none"> • <u>Especialista de selección y contratación</u>
Dependencias	• <u>[OBJ-0003] Generación de información administrativa y contable</u>
Descripción	El sistema deberá <i>definir los cargos y puestos que desempeñan los empleados.</i>
Importancia	vital
Urgencia	puede esperar
Estado	en construcción
Estabilidad	media
Comentarios	Ninguno

Fuente: Elaboración propia.

RF-0010 - Administrar instituciones educativas y no educativas.

FRQ-0010	Administrar instituciones educativas y no educativas
Versión	1.0 (20/04/2023)
Autores	<ul style="list-style-type: none"> • <u>Eddy Sarantes</u> • <u>Harold Solis</u> • <u>Jeferson Cano</u>
Fuentes	<ul style="list-style-type: none"> • <u>Especialista de selección y contratación</u>
Dependencias	• <u>[OBJ-0003] Generación de información administrativa y contable</u>
Descripción	El sistema deberá <i>describir las instituciones educativas y no educativas las cuales avalan los cursos y estudios de los candidatos y empleados.</i>
Importancia	vital
Urgencia	puede esperar
Estado	en construcción
Estabilidad	media
Comentarios	Ninguno

Fuente: Elaboración propia.

RF-0011 - Administrar parentesco.

FRQ-0011	Administrar parentesco
Versión	1.0 (20/04/2023)
Autores	<ul style="list-style-type: none"> • <u>Eddy Sarantes</u> • <u>Harold Solis</u> • <u>Jeferson Cano</u>
Fuentes	<ul style="list-style-type: none"> • <u>Especialista de selección y contratación</u>
Dependencias	<ul style="list-style-type: none"> • <u>[OBJ-0003] Generación de información administrativa y contable</u> • <u>[OBJ-0005] Generación de reportes gerenciales para análisis y toma de decisión</u>
Descripción	El sistema deberá <i>registrar y actualiza los datos de los familiares de empleados una vez contratados.</i>
Importancia	importante
Urgencia	inmediatamente
Estado	en construcción
Estabilidad	media
Comentarios	Ninguno

Fuente: Elaboración propia.

RF-0012 - Administrar municipios.

FRQ-0012	Administrar municipios
Versión	1.0 (20/04/2023)
Autores	<ul style="list-style-type: none"> • <u>Eddy Sarantes</u> • <u>Harold Solis</u> • <u>Jeferson Cano</u>
Fuentes	<ul style="list-style-type: none"> • <u>Especialista de selección y contratación</u>
Dependencias	<ul style="list-style-type: none"> • <u>[OBJ-0003] Generación de información administrativa y contable</u>
Descripción	El sistema deberá <i>contener la clasificación de los municipios por departamento del país.</i>
Importancia	importante
Urgencia	puede esperar
Estado	en construcción
Estabilidad	media
Comentarios	Ninguno

Fuente: Elaboración propia.

RF-0013 - Administrar departamentos.

FRQ-0013	Administrar departamentos
Versión	1.0 (20/04/2023)
Autores	<ul style="list-style-type: none"> • <u>Eddy Sarantes</u> • <u>Harold Solis</u> • <u>Jeferson Cano</u>
Fuentes	<ul style="list-style-type: none"> • <u>Especialista de selección y contratación</u>
Dependencias	• <u>[OBJ-0003] Generación de información administrativa y contable</u>
Descripción	El sistema deberá <i>registrar y actualiza los datos de ubicación donde habita el empleado.</i>
Importancia	vital
Urgencia	puede esperar
Estado	en construcción
Estabilidad	media
Comentarios	Ninguno

Fuente: Elaboración propia.

RF-0014 - Generar nómina.

FRQ-0014	Generar nómina
Versión	1.0 (20/04/2023)
Autores	<ul style="list-style-type: none"> • <u>Eddy Sarantes</u> • <u>Harold Solis</u> • <u>Jeferson Cano</u>
Fuentes	<ul style="list-style-type: none"> • <u>Especialista de Nómina</u>
Dependencias	<ul style="list-style-type: none"> • <u>[OBJ-0001] Generar la nómina</u> • <u>[OBJ-0002] Control de deducciones</u> • <u>[OBJ-0003] Generación de información administrativa y contable</u> • <u>[OBJ-0005] Generación de reportes gerenciales para análisis y toma de decisión</u> • <u>[OBJ-0007] Mantener la integridad, disponibilidad y confidencialidad de la información</u> • <u>[OBJ-0006] Reducir el tiempo y el esfuerzo en la elaboración de los informes</u>
Descripción	El sistema deberá <i>generar la nómina, una vez los datos estén verificados. y tanto las compensaciones y deducciones del periodo hayan sido ingresados.</i>
Importancia	vital
Urgencia	inmediatamente
Estado	en construcción
Estabilidad	alta
Comentarios	Ninguno

Fuente: Elaboración propia.

RF-0015 - Importar deducciones y compensaciones del empleado.

FRQ-0015	Importar deducciones y compensaciones del empleado
Versión	1.0 (20/04/2023)
Autores	<ul style="list-style-type: none"> • Eddy Sarantes • Harold Solis • Jeferson Cano
Fuentes	<ul style="list-style-type: none"> • Especialista de Nómina
Dependencias	<ul style="list-style-type: none"> • [OBJ-0002] Control de deducciones • [FRQ-0001] Generar reportes administrativos
Descripción	El sistema deberá <i>permitir importar los datos de forma masiva desde una plantilla de excel para los datos de las compensaciones y deducciones.</i>
Importancia	vital
Urgencia	inmediatamente
Estado	en construcción
Estabilidad	media
Comentarios	Ninguno

Fuente: Elaboración propia.

RF-0016 - Registrar compensaciones y deducciones del empleado.

FRQ-0016	Registrar compensaciones y deducciones del empleado
Versión	1.0 (21/04/2023)
Autores	<ul style="list-style-type: none"> • Eddy Sarantes • Harold Solis • Jeferson Cano
Fuentes	<ul style="list-style-type: none"> • Especialista de Nómina
Dependencias	<ul style="list-style-type: none"> • [OBJ-0002] Control de deducciones • [FRQ-0001] Generar reportes administrativos
Descripción	El sistema deberá <i>registrar las compensaciones y deducciones que serán aplicadas empleado.</i>
Importancia	vital
Urgencia	inmediatamente
Estado	en construcción
Estabilidad	media
Comentarios	Ninguno

Fuente: Elaboración propia.

RF-0017 - Administrar estudios académicos.

FRQ-0017	Administrar estudios académicos
Versión	1.0 (21/04/2023)
Autores	<ul style="list-style-type: none"> • <u>Eddy Sarantes</u> • <u>Harold Solis</u> • <u>Jeferson Cano</u>
Fuentes	<ul style="list-style-type: none"> • <u>Especialista de selección y contratación</u>
Dependencias	• <u>[OBJ-0003] Generación de información administrativa y contable</u>
Descripción	El sistema deberá <i>definir los estudios que contienen al menos un título académico de alguna institución.</i>
Importancia	vital
Urgencia	inmediatamente
Estado	en construcción
Estabilidad	baja
Comentarios	Ninguno

Fuente: Elaboración propia.

RF-0018 - Administrar grupos de compensaciones y deducciones.

FRQ-0018	Administrar grupos de compensaciones y deducciones
Versión	1.0 (21/04/2023)
Autores	<ul style="list-style-type: none"> • <u>Eddy Sarantes</u> • <u>Harold Solis</u> • <u>Jeferson Cano</u>
Fuentes	<ul style="list-style-type: none"> • <u>Especialista de Nómina</u>
Dependencias	<ul style="list-style-type: none"> • <u>[FRQ-0001] Generar reportes administrativos</u> • <u>[OBJ-0002] Control de deducciones</u>
Descripción	El sistema deberá <i>permitir agrupar las compensaciones y deducciones por tipo.</i>
Importancia	vital
Urgencia	inmediatamente
Estado	en construcción
Estabilidad	media
Comentarios	Ninguno

Fuente: Elaboración propia.

RF-0019 - Abrir periodos de nómina.

FRQ-0019	Abrir periodos de nómina
Versión	1.0 (21/04/2023)
Autores	<ul style="list-style-type: none"> • Eddy Sarantes • Harold Solis • Jeferson Cano
Fuentes	<ul style="list-style-type: none"> • Especialista de Nómina
Dependencias	<ul style="list-style-type: none"> • [OBJ-0003] Generación de información administrativa y contable • [OBJ-0007] Mantener la integridad, disponibilidad y confidencialidad de la información • [OBJ-0006] Reducir el tiempo y el esfuerzo en la elaboración de los informes • [OBJ-0005] Generación de reportes gerenciales para análisis y toma de decisión • [OBJ-0002] Control de deducciones • [OBJ-0001] Generar la nómina
Descripción	El sistema deberá <i>permitira crear el periodo en que va a generarse la nómina.</i>
Importancia	vital
Urgencia	hay presión
Estado	en construcción
Estabilidad	media
Comentarios	Ninguno

Fuente: Elaboración propia.

RF-0020 - Agregar empleado a nómina.

FRQ-0020	Agregar empleado a nómina
Versión	1.0 (21/04/2023)
Autores	<ul style="list-style-type: none"> • Eddy Sarantes • Harold Solis • Jeferson Cano
Fuentes	<ul style="list-style-type: none"> • Especialista de Nómina
Dependencias	<ul style="list-style-type: none"> • [FRQ-0004] Administrar dependencias • [OBJ-0001] Generar la nómina
Descripción	El sistema deberá <i>al momento de importar las compensaciones y deducciones se puede agregar un empleado a la nómina, una vez inicializado el proceso de nómina.</i>
Importancia	importante
Urgencia	inmediatamente
Estado	en construcción
Estabilidad	media
Comentarios	Ninguno

Fuente: Elaboración propia.

RF-0021 - Registrar empleado.

FRQ-0021	Registrar empleado
Versión	1.0 (21/04/2023)
Autores	<ul style="list-style-type: none"> • <u>Eddy Sarantes</u> • <u>Harold Solis</u> • <u>Jeferson Cano</u>
Fuentes	<ul style="list-style-type: none"> • <u>Especialista de selección y contratación</u>
Dependencias	<ul style="list-style-type: none"> • <u>[FRQ-0001] Generar reportes administrativos</u> • <u>[FRQ-0003] Administrar perfil del empleado</u> • <u>[OBJ-0001] Generar la nómina</u>
Descripción	El sistema deberá <i>permitir ingresar empleados de nuevo ingreso.</i>
Importancia	importante
Urgencia	hay presión
Estado	pendiente de verificación
Estabilidad	media
Comentarios	Ninguno

Fuente: Elaboración propia.

RF-0022 - Generar reporte de comprobante de pago.

FRQ-0022	Generar reporte de comprobante de pago
Versión	1.0 (21/04/2023)
Autores	<ul style="list-style-type: none"> • <u>Eddy Sarantes</u> • <u>Harold Solis</u> • <u>Jeferson Cano</u>
Fuentes	<ul style="list-style-type: none"> • <u>Especialista de Nómina</u>
Dependencias	<ul style="list-style-type: none"> • <u>[OBJ-0003] Generación de información administrativa y contable</u> • <u>[FRQ-0001] Generar reportes administrativos</u>
Descripción	El sistema deberá <i>generar colilla de pago una vez aplicada la nómina.</i>
Importancia	vital
Urgencia	inmediatamente
Estado	en construcción
Estabilidad	media
Comentarios	Ninguno

Fuente: Elaboración propia.

RF-0022 - Generar reporte de comprobante de pago.

FRQ-0023	Generar contrato laboral
Versión	1.0 (08/09/2023)
Autores	<ul style="list-style-type: none"> • Eddy Sarantes • Harold Solis • Jeferson Cano
Fuentes	• Especialista de selección y contratación
Dependencias	<ul style="list-style-type: none"> • [FRQ-0001] Generar reportes administrativos • [OBJ-0008] Generación del contrato laboral
Descripción	El sistema deberá <i>generar el contrato laboral de cada empleado que será contratado una vez implementado el sistema.</i>
Importancia	vital
Urgencia	hay presión
Estado	pendiente de validación
Estabilidad	media
Comentarios	Ninguno

Fuente: Elaboración propia.

RNF-0003 - Requerimiento de almacenamiento.

NFR-0003	Requerimiento de almacenamiento
Versión	1.0 (22/04/2023)
Autores	<ul style="list-style-type: none"> • Eddy Sarantes • Harold Solis • Jeferson Cano
Fuentes	• Responsable TI
Dependencias	Ninguno
Descripción	El sistema deberá <i>tener definido el lugar en el servidor donde serán almacenado los datos, este espacio tiene que ser mayor a 5GB.</i>
Importancia	importante
Urgencia	puede esperar
Estado	en construcción
Estabilidad	media
Comentarios	Ninguno

Fuente: Elaboración propia.

RNF-0004 - Requerimiento de hardware.

NFR-0004	Requerimiento de hardware
Versión	1.0 (22/04/2023)
Autores	<ul style="list-style-type: none"> • Eddy Sarantes • Harold Solis • Jeferson Cano
Fuentes	<ul style="list-style-type: none"> • Responsable TI
Dependencias	Ninguno
Descripción	El sistema deberá contar con un servidor con los requisitos mínimos para ejecutar el sistema sin problemas, los cuales son al menos 4gb de memoria RAM dedicada, procesador Core i5, almacenamiento de al menos 256GB.
Importancia	importante
Urgencia	puede esperar
Estado	en construcción
Estabilidad	media
Comentarios	Ninguno

Fuente: Elaboración propia.

RNF-0005 - Requerimiento de red.

NFR-0005	Requerimiento de red
Versión	1.0 (22/04/2023)
Autores	<ul style="list-style-type: none"> • Eddy Sarantes • Harold Solis • Jeferson Cano
Fuentes	<ul style="list-style-type: none"> • Responsable TI
Dependencias	Ninguno
Descripción	El sistema deberá contar con una conexión a Internet para que los usuarios tengan acceso al sistema. La empresa cuenta con una red LAN con acceso a internet, el servicio de internet es provisto por el proveedor IBW con un ancho de banda de 50mbps, de forma simétrica con enlace de fibra óptica dedicado.
Importancia	vital
Urgencia	puede esperar
Estado	en construcción
Estabilidad	media
Comentarios	Ninguno

Fuente: Elaboración propia.

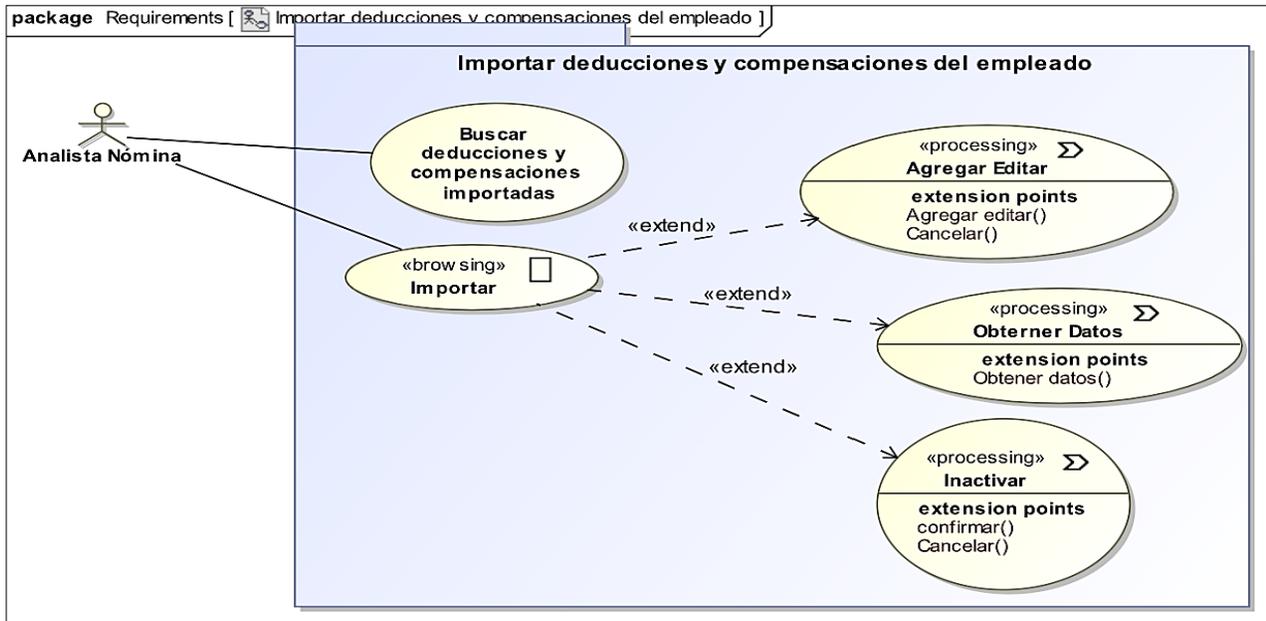
RNF-0006 - Requerimiento de interfaz.

NFR-0006	Requerimiento de interfaz
Versión	1.0 (22/04/2023)
Autores	<ul style="list-style-type: none">• Eddy Sarantes• Harold Solis• Jeferson Cano
Fuentes	<ul style="list-style-type: none">• Responsable TI
Dependencias	Ninguno
Descripción	El sistema deberá <i>presentar una interfaz simple, agradable visualmente, responsiva, de fácil uso con el fin que el usuario puede explorar y comprender su funcionalidad.</i>
Importancia	importante
Urgencia	hay presión
Estado	en construcción
Estabilidad	media
Comentarios	Ninguno

Fuente: Elaboración propia.

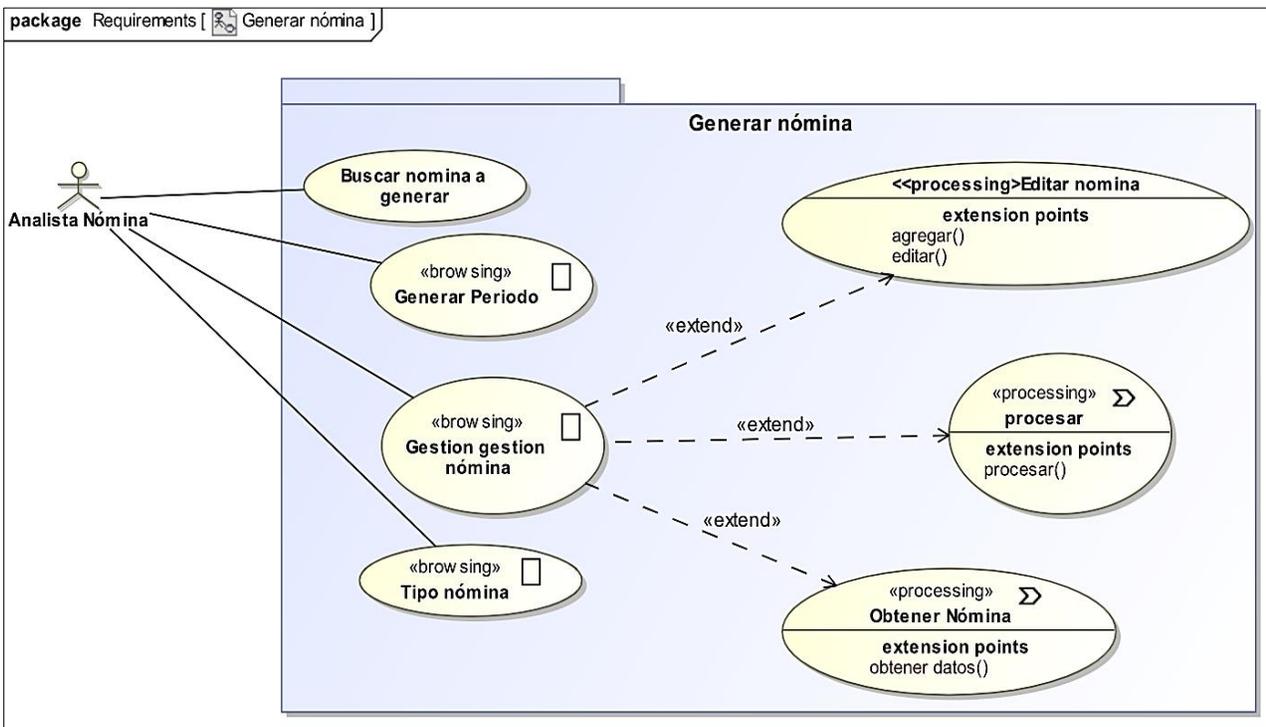
Anexo 3 : Diagramas de casos de uso.

Diagrama de casos de uso - Importar deducciones y compensaciones del empleado.



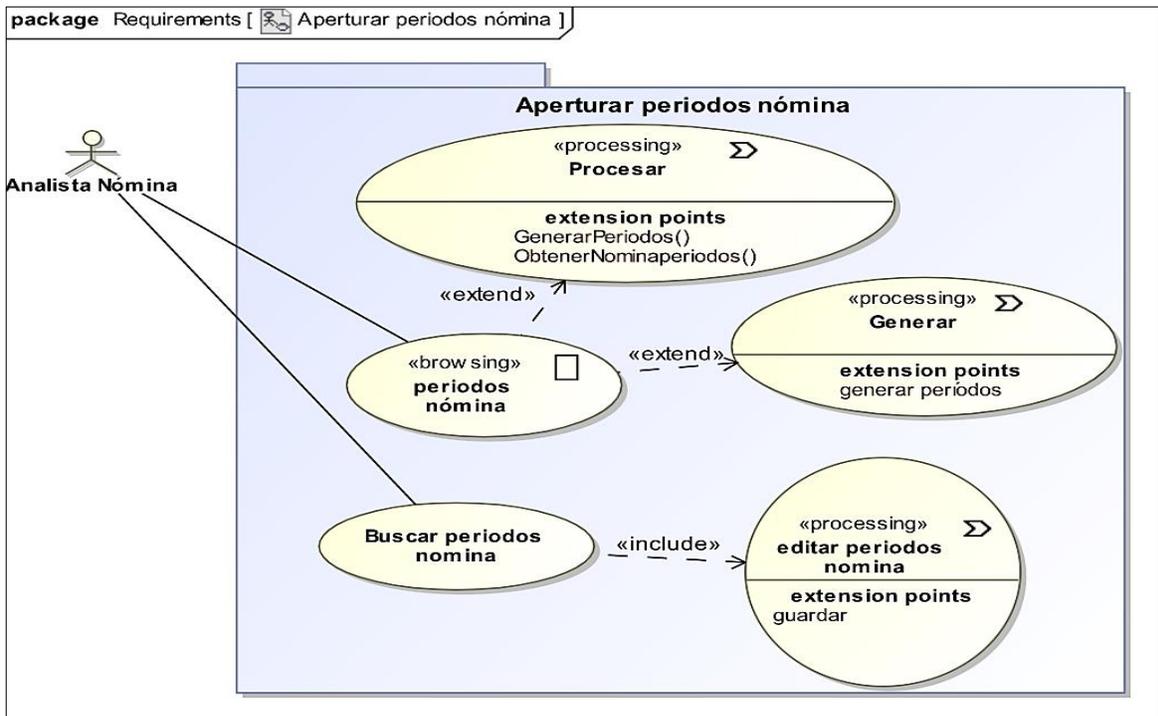
Fuente: Elaboración propia.

Diagrama de casos de uso - Generar nómina.



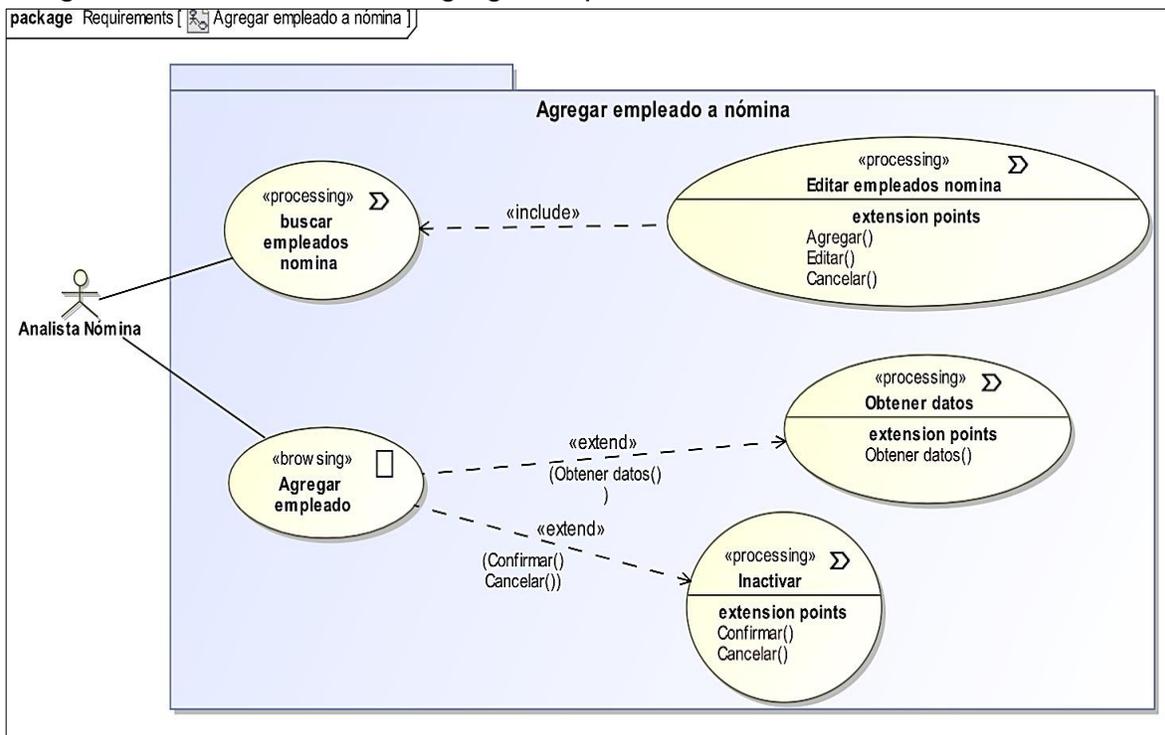
Fuente: Elaboración propia.

Diagrama de casos de uso - Aperturar periodos nómina.



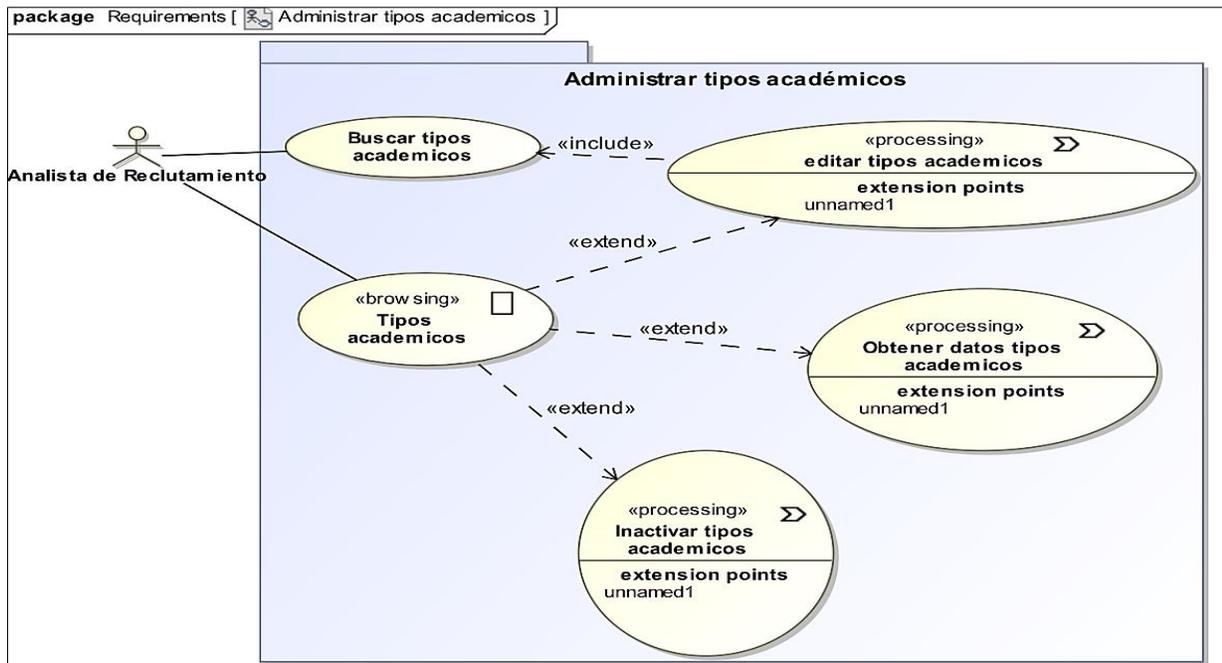
Fuente: Elaboración propia.

Diagrama de casos de uso - Agregar empleado a nómina.



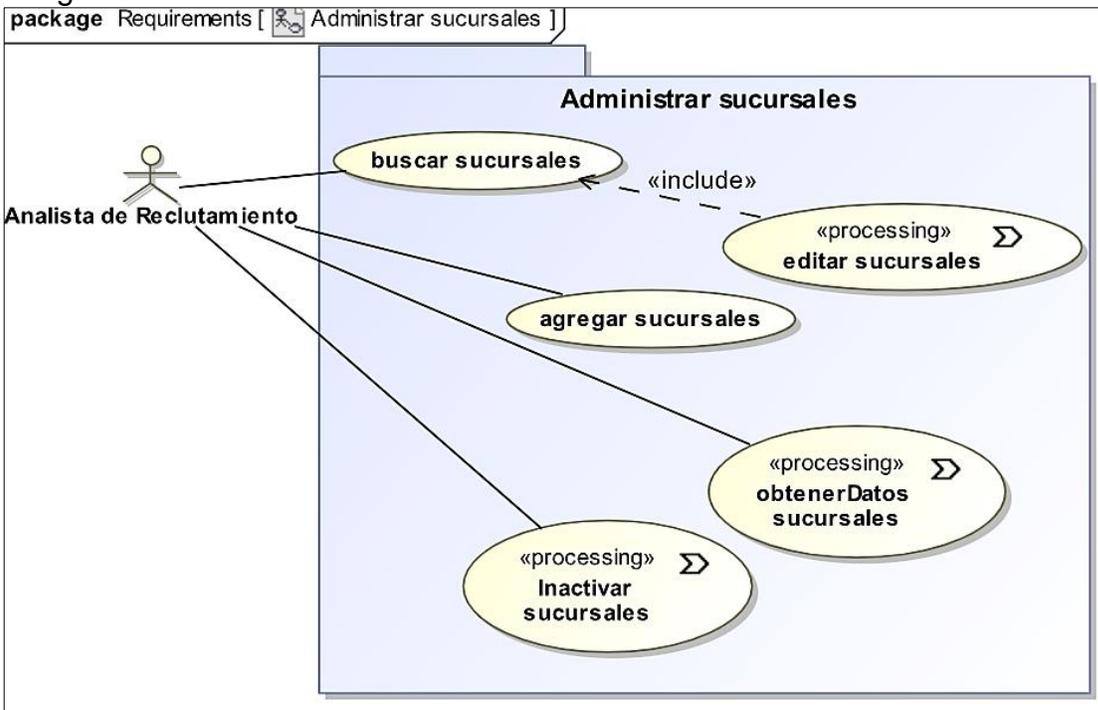
Fuente: Elaboración propia.

Diagrama de casos de uso - Administrar tipos académicos.



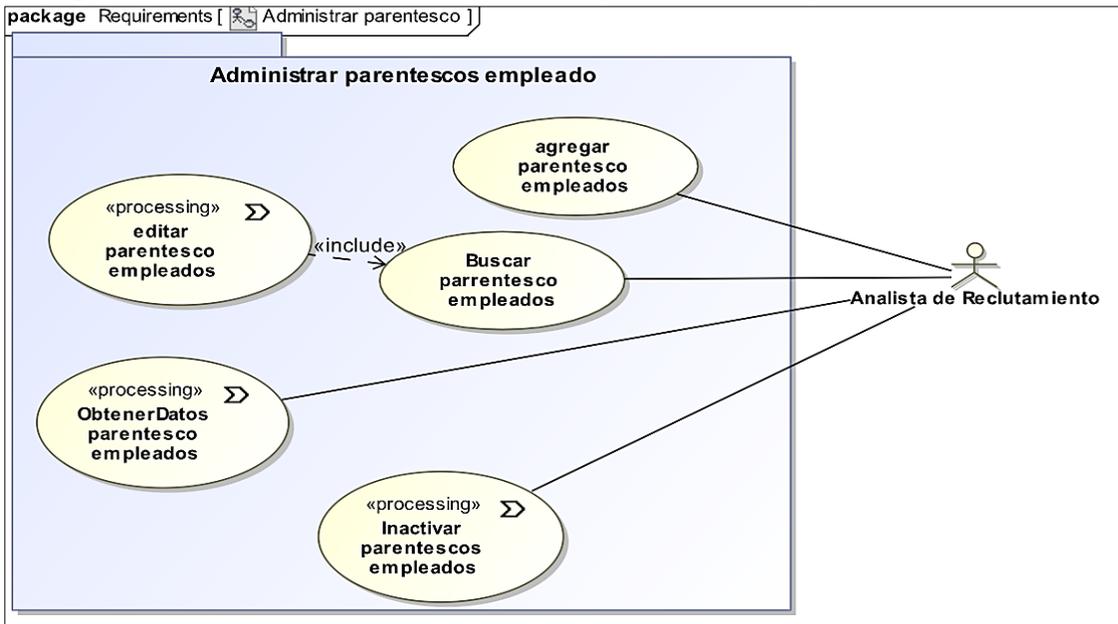
Fuente: Elaboración propia.

Diagrama de casos de uso - Administrar sucursales.



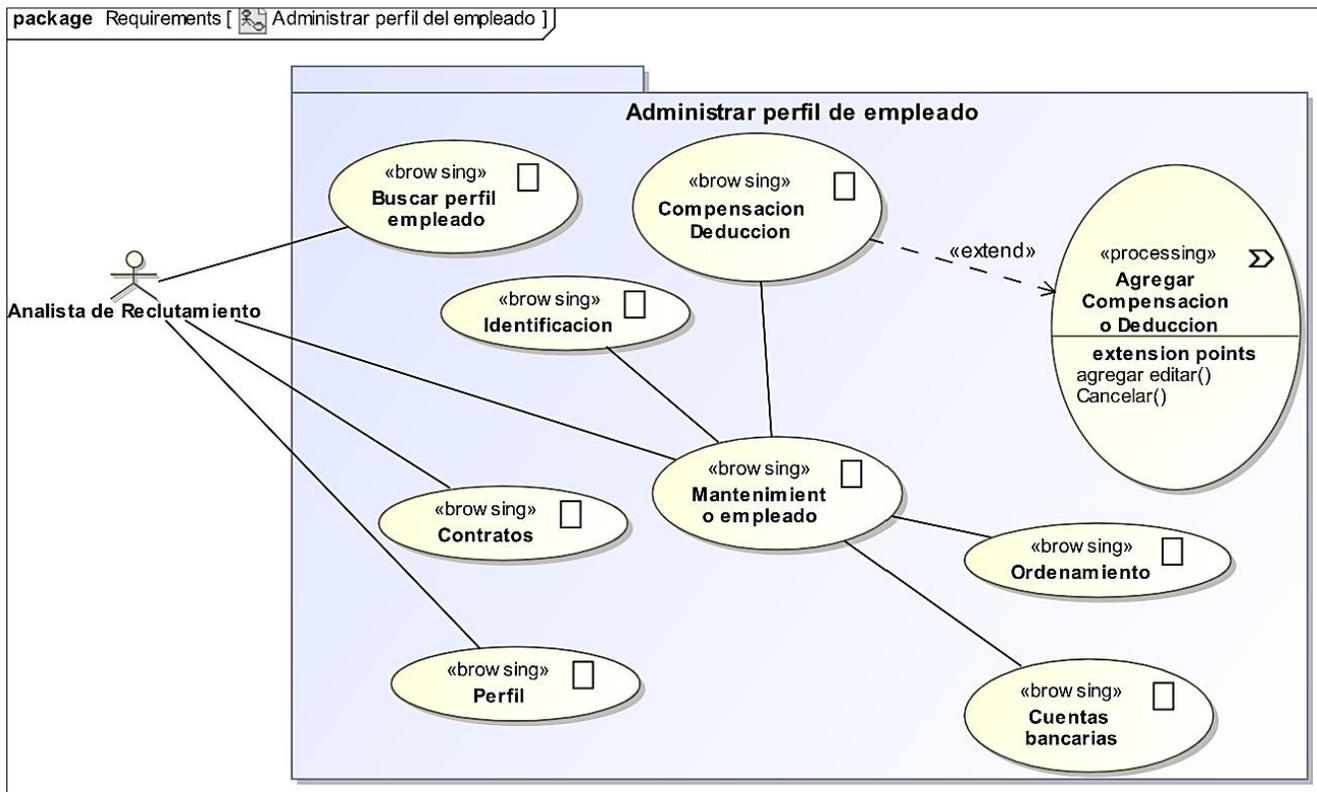
Fuente: Elaboración propia.

Diagrama de casos de uso - Administrar parentesco empleado.



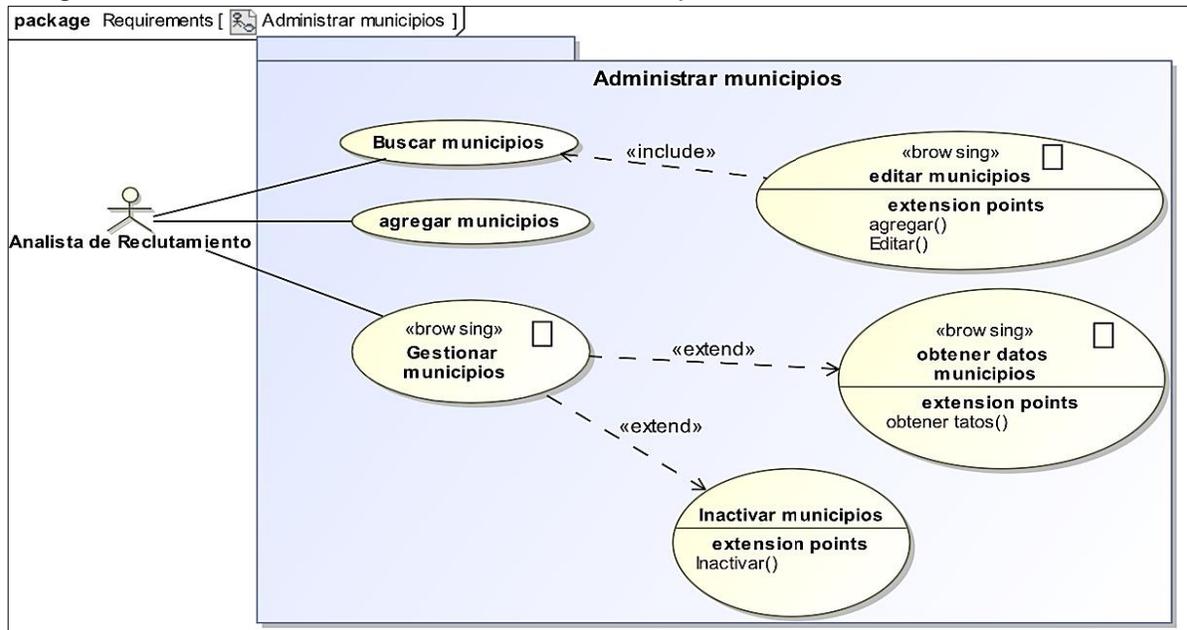
Fuente: Elaboración propia.

Diagrama de casos de uso - Administrar perfil del empleado.



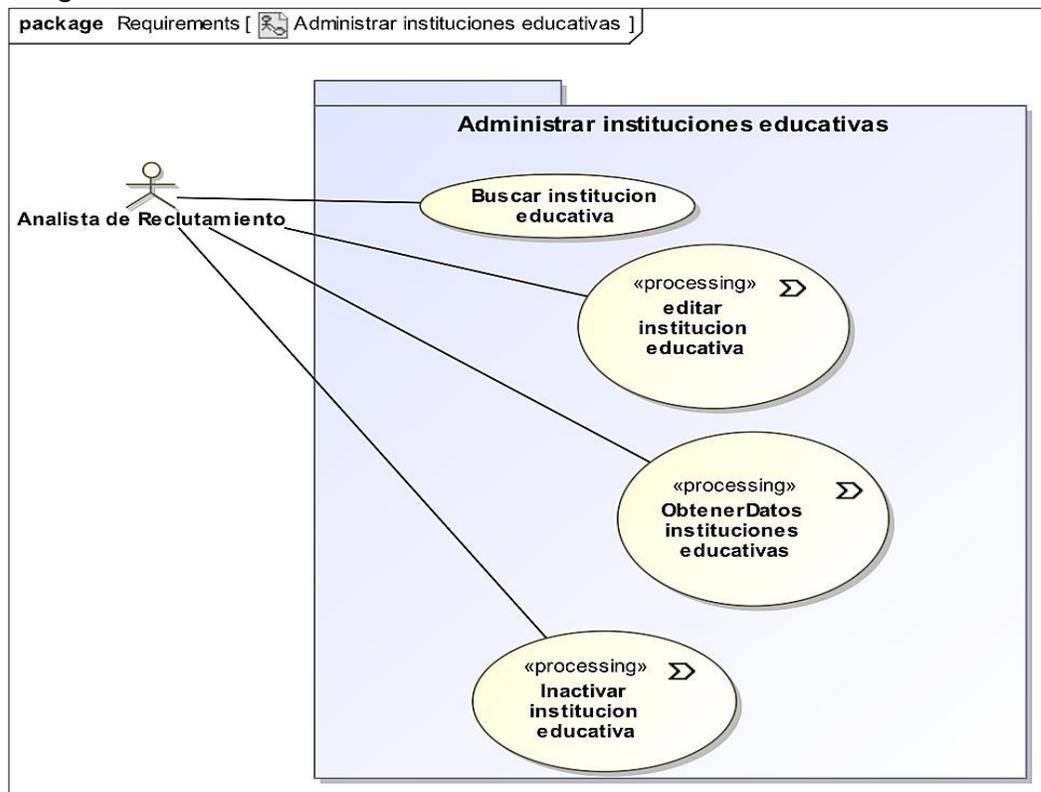
Fuente: Elaboración propia.

Diagrama de casos de uso - Administrar municipios.



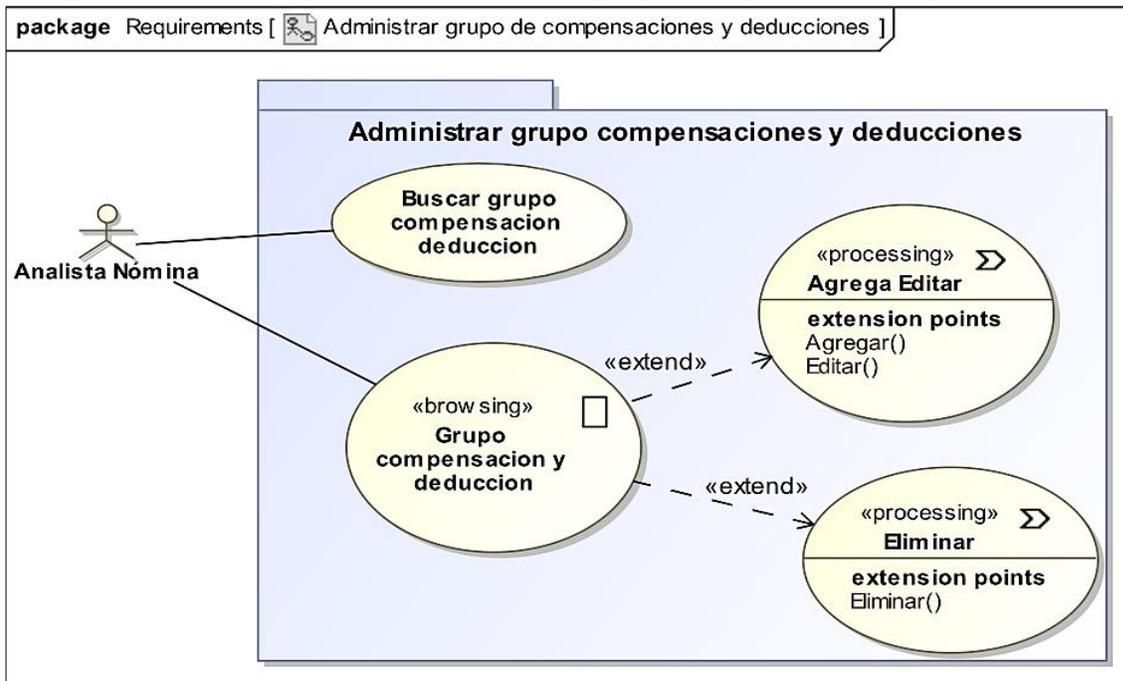
Fuente: Elaboración propia.

Diagrama de casos de uso - Administrar instituciones educativas.



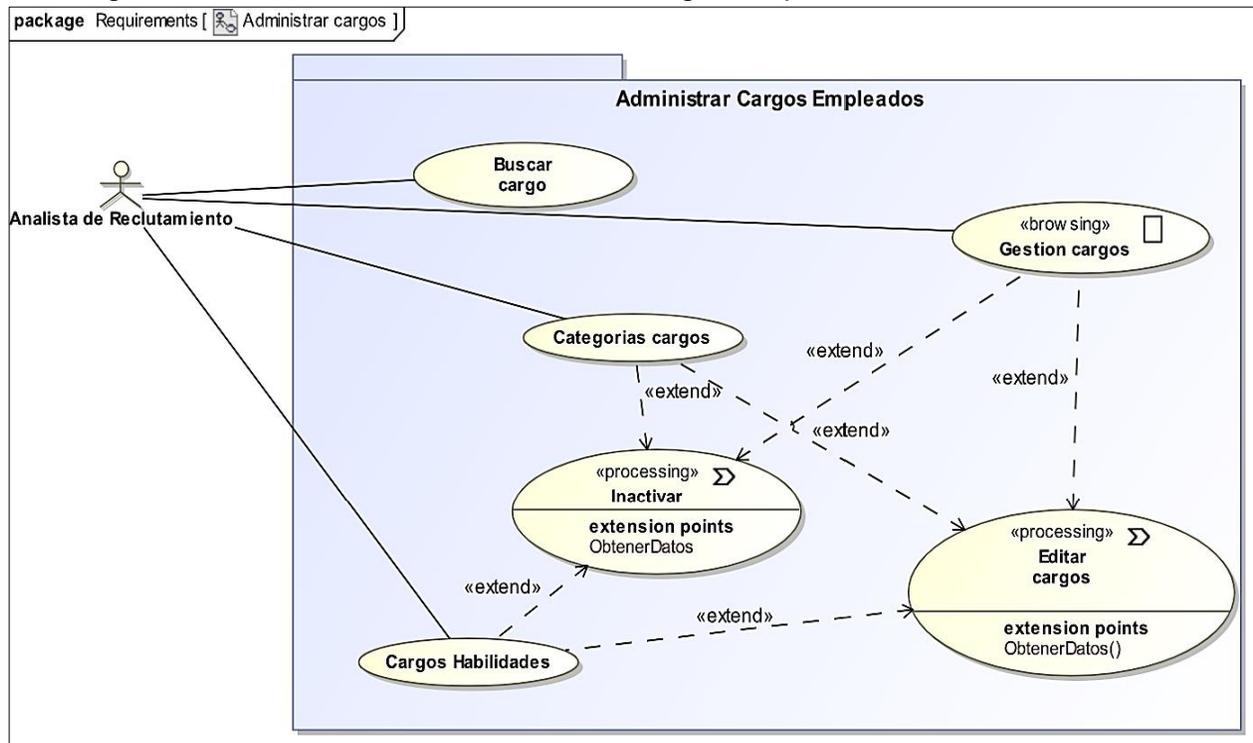
Fuente: Elaboración propia.

Diagrama de casos de uso - Administrar grupo compensaciones y deducciones.



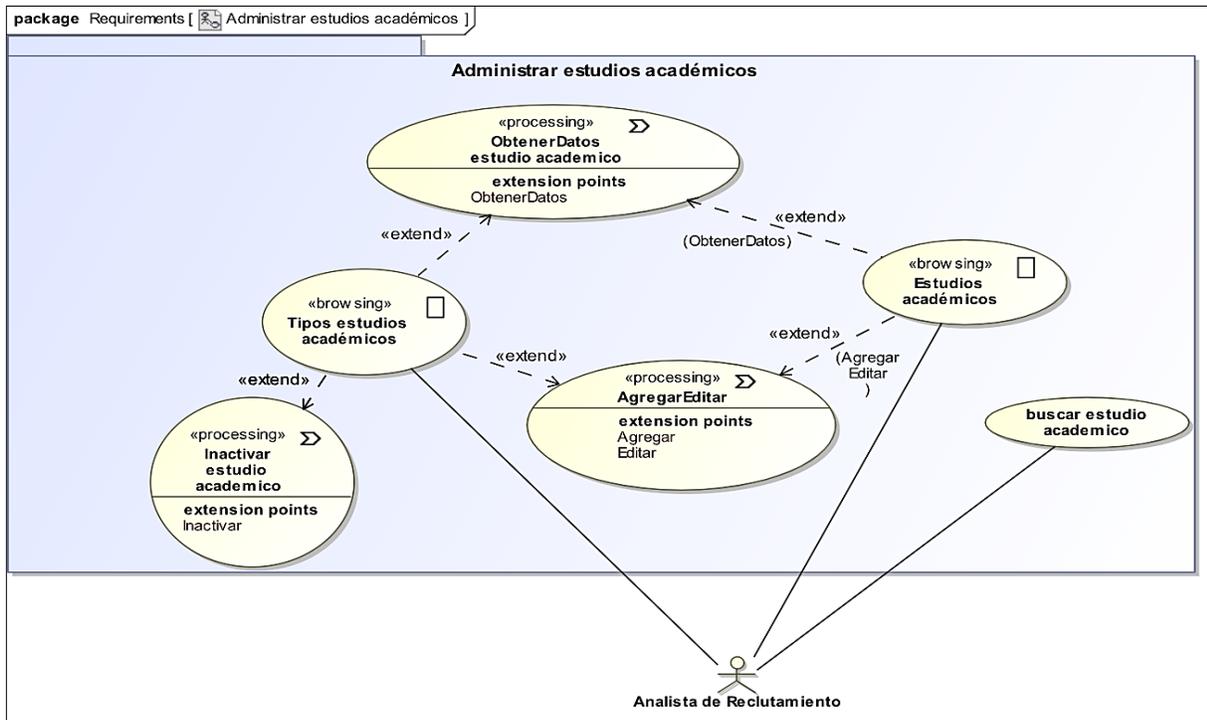
Fuente: Elaboración propia.

Diagrama de casos de uso - Administrar cargos empleados.



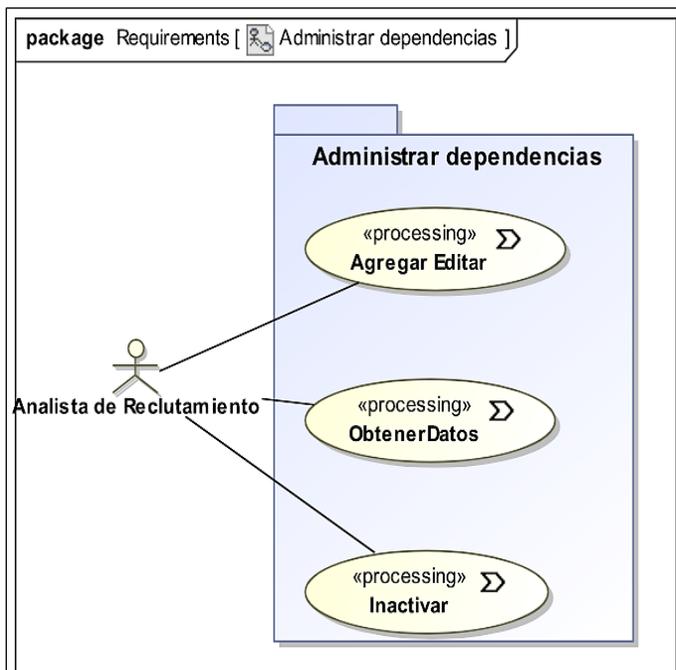
Fuente: Elaboración propia.

Diagrama de casos de uso - Administrar estudios académicos.



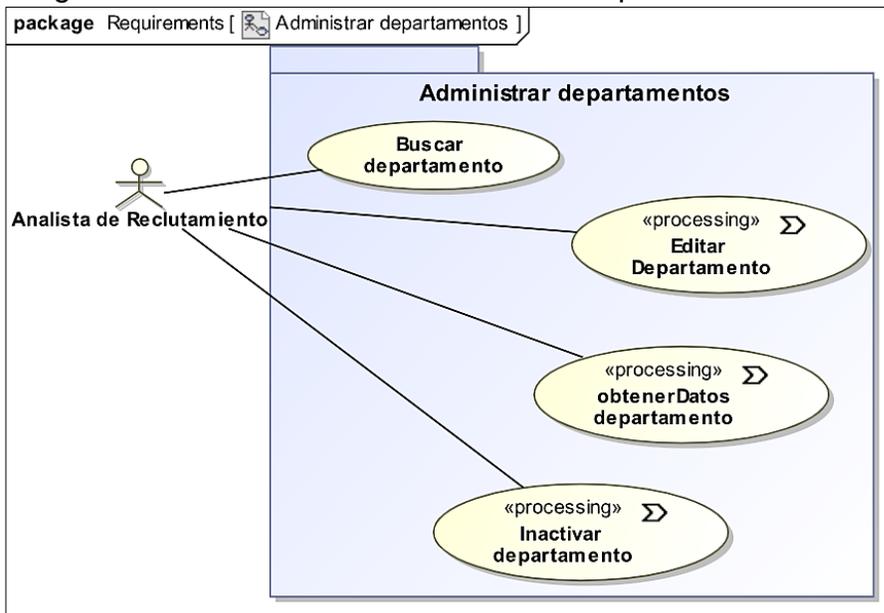
Fuente: Elaboración propia.

Diagrama de casos de uso - Administrar dependencias.



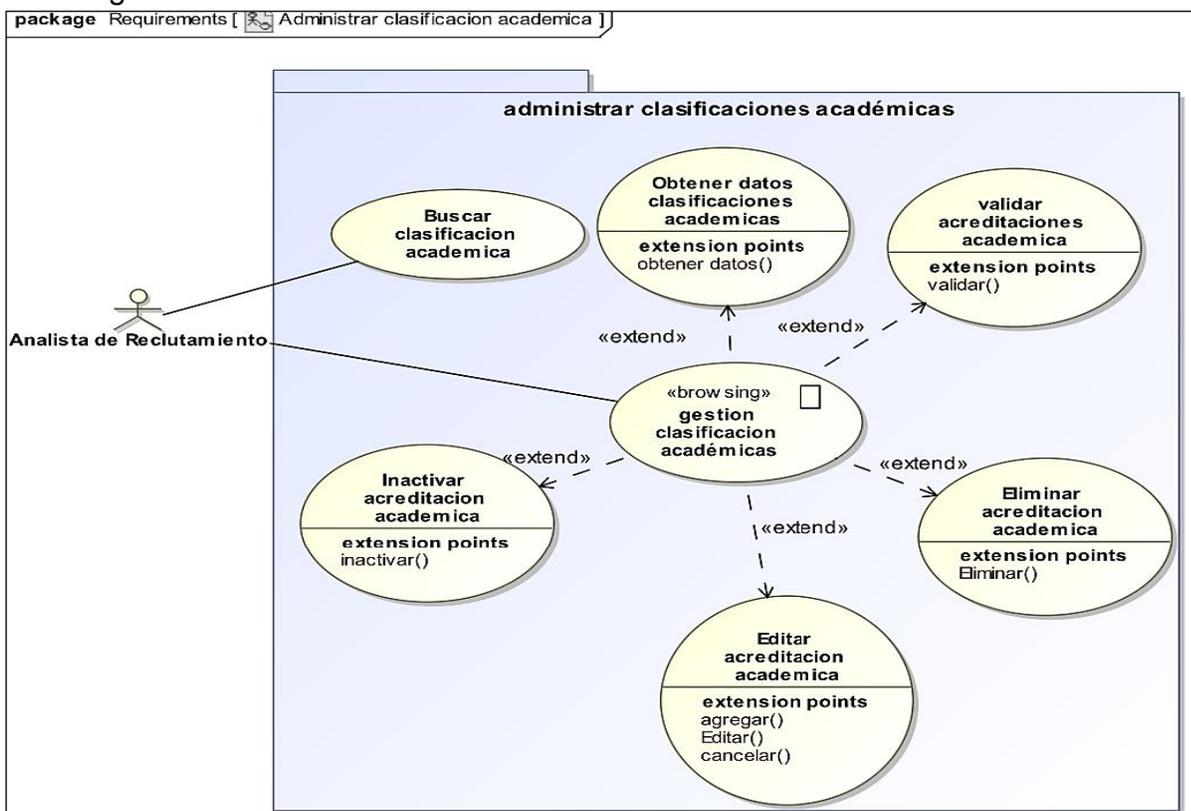
Fuente: Elaboración propia.

Diagrama de casos de uso - Administrar departamentos.



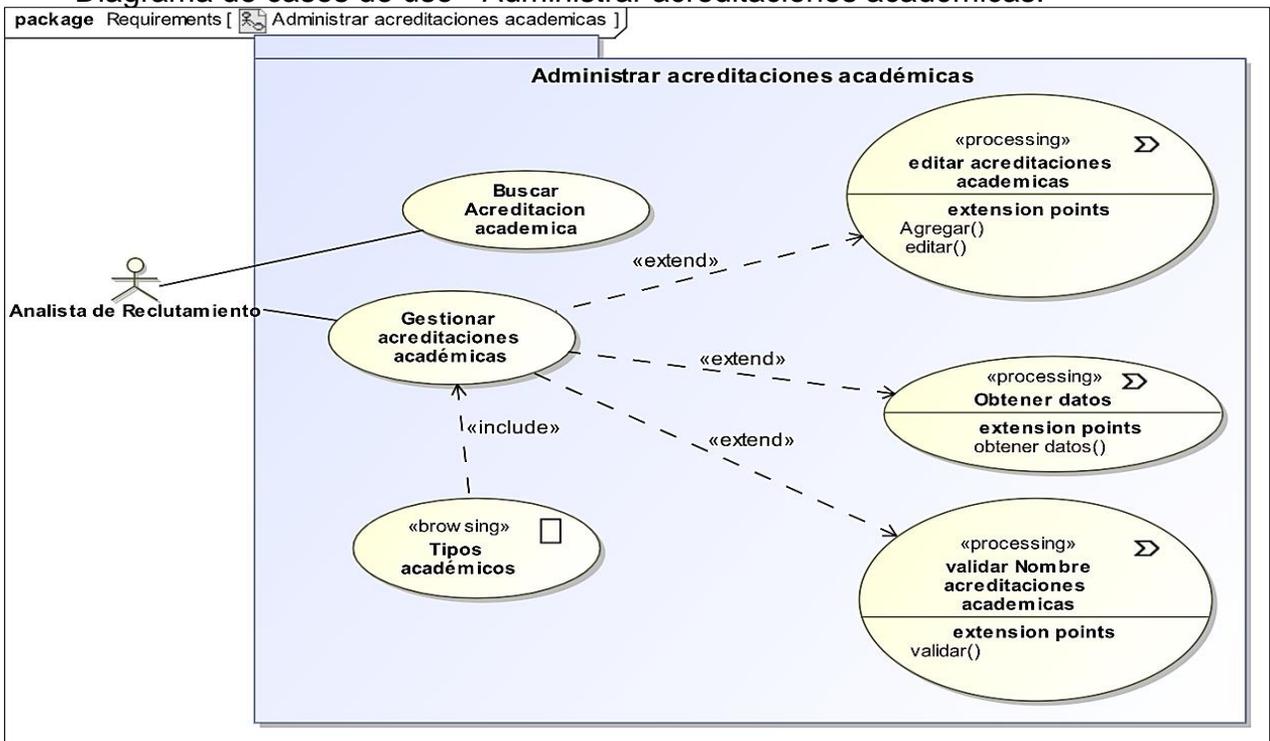
Fuente: Elaboración propia.

Diagrama de casos de uso - Administrar clasificaciones académicas.



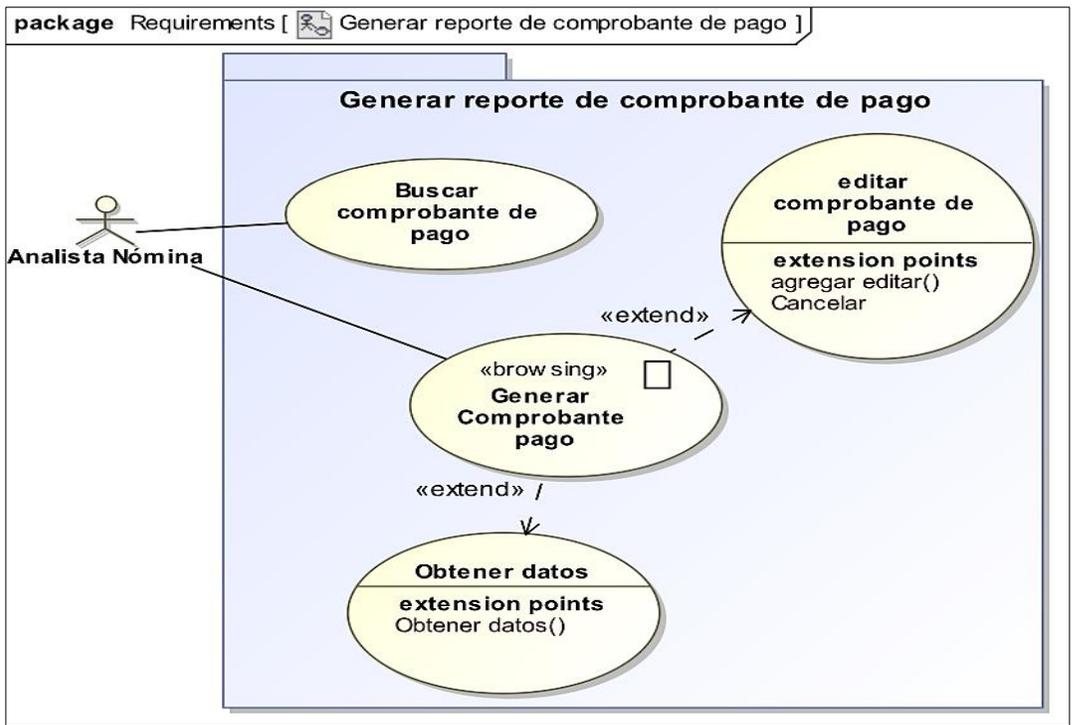
Fuente: Elaboración propia.

Diagrama de casos de uso - Administrar acreditaciones académicas.



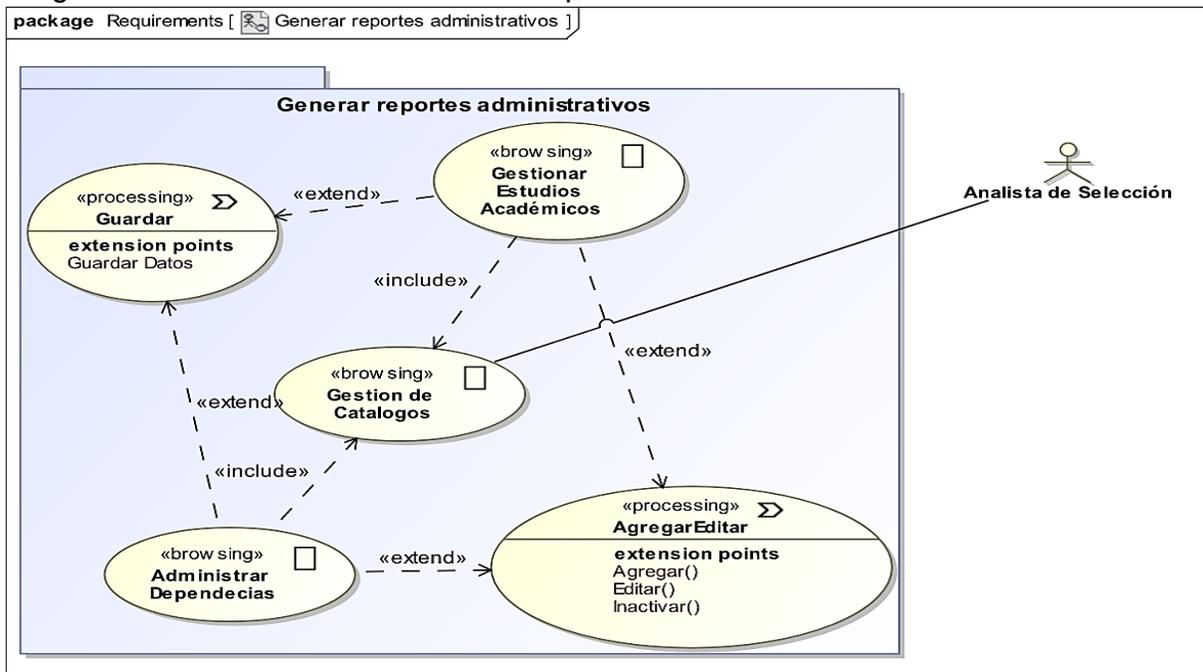
Fuente: Elaboración propia.

Diagrama de casos de uso - Generar reporte de comprobante de pago.



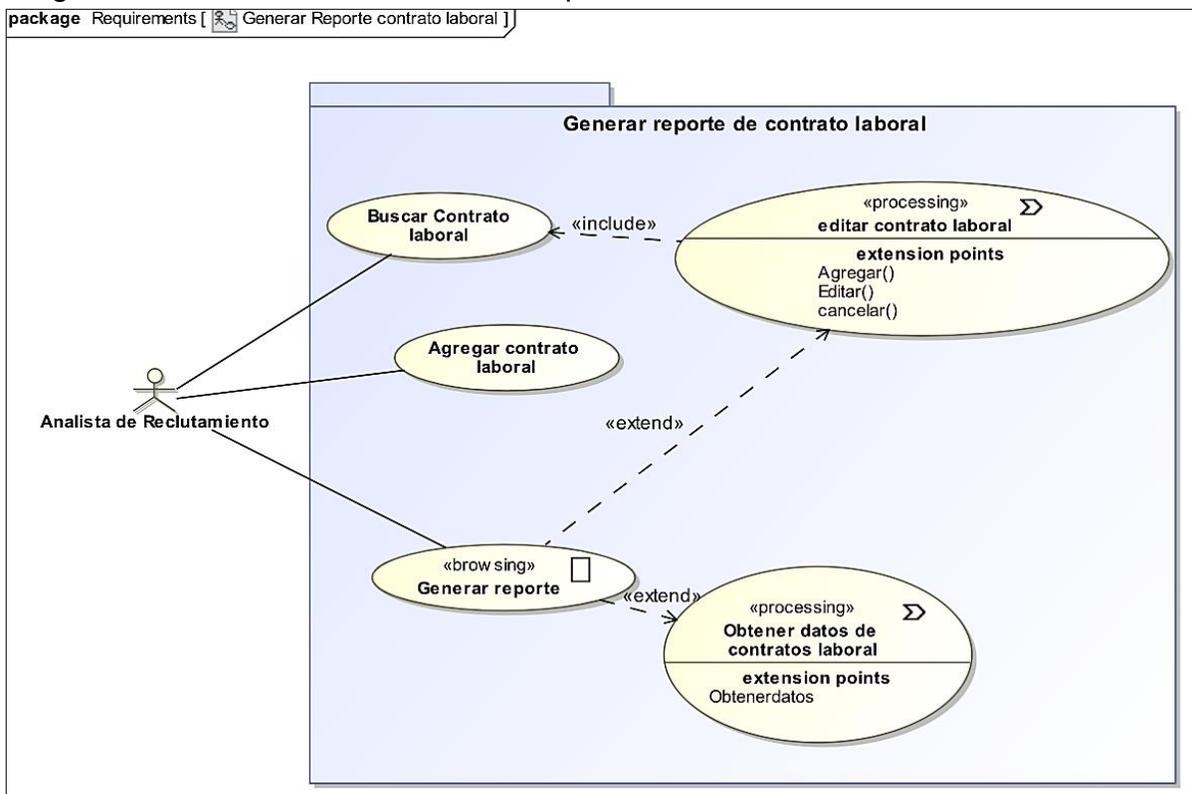
Fuente: Elaboración propia.

Diagrama de casos de uso - Generar reportes administrativos.



Fuente: Elaboración propia.

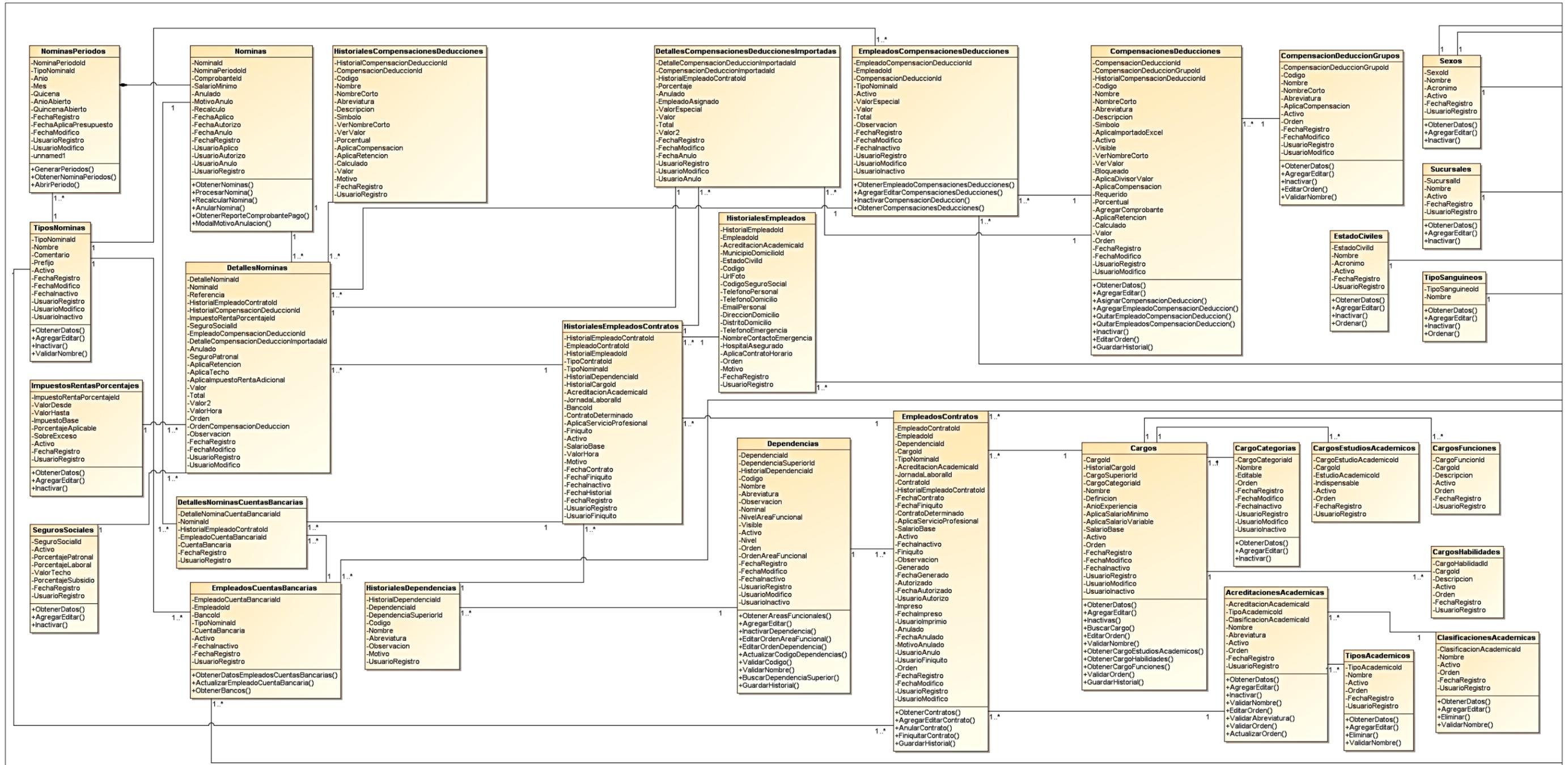
Diagrama de casos de uso - Generar reporte de contrato laboral.



Fuente: Elaboración propia.

Anexo 4: Diagrama de contenido.

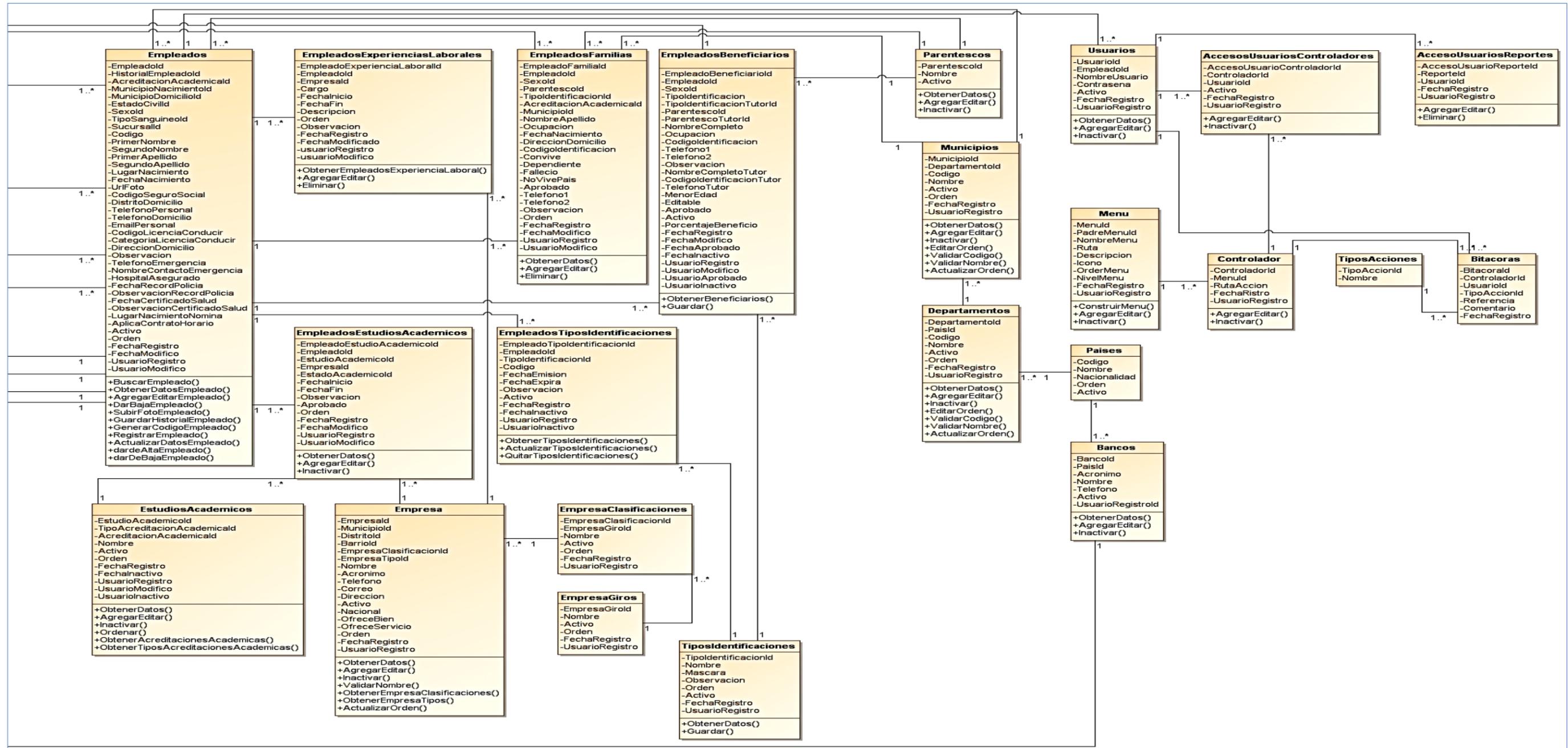
(Parte 1).



Fuente: Elaboración propia.

Anexo 4: Diagrama de contenido.

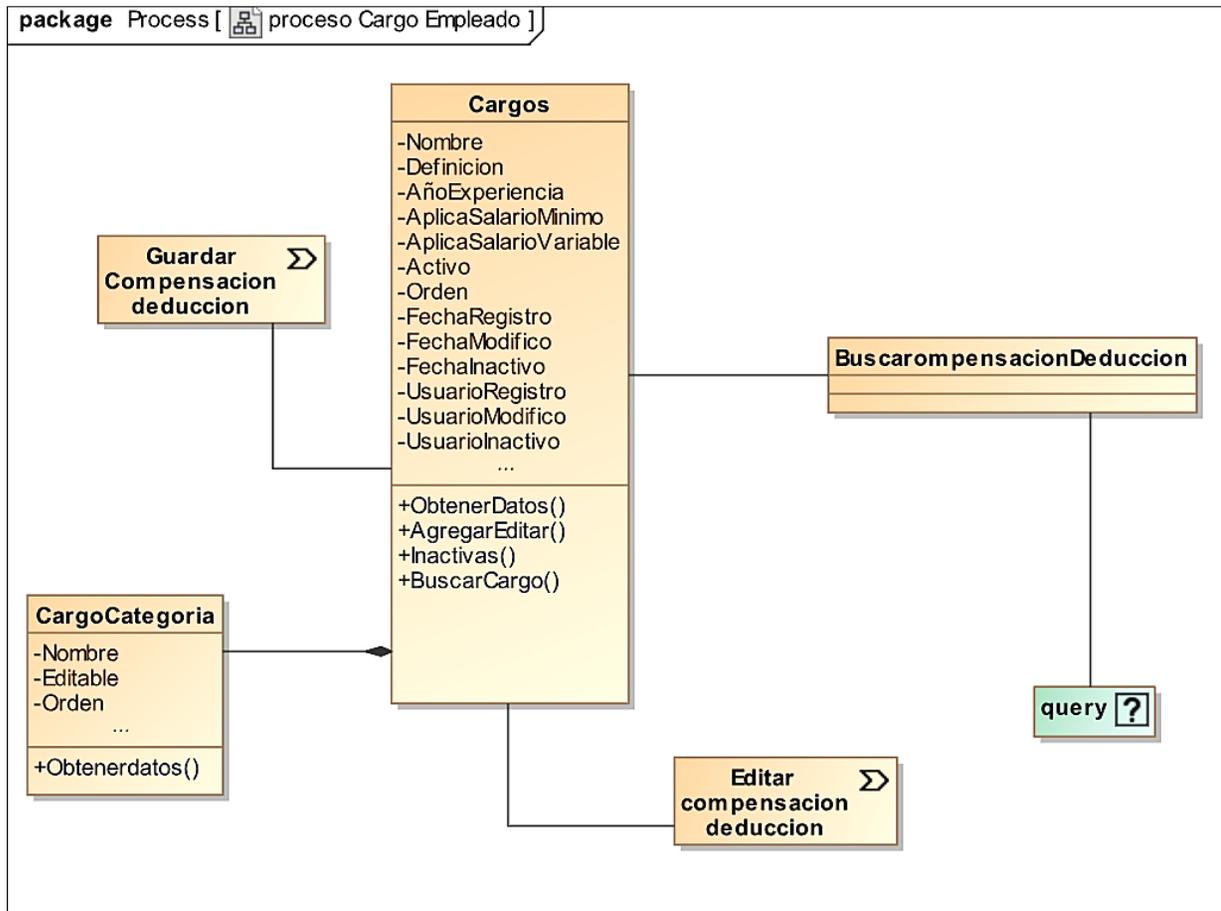
(Parte 2).



Fuente: Elaboración propia.

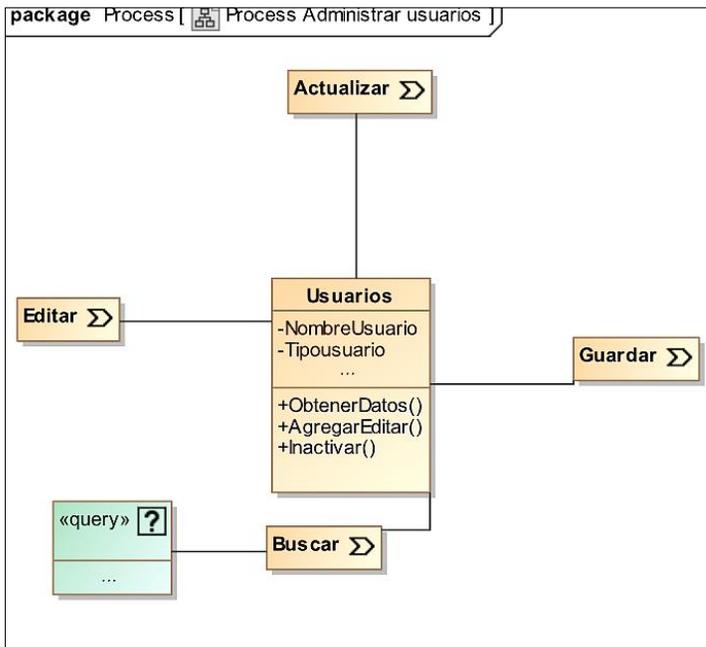
Anexo 5 : Modelo de proceso.

Diagrama de proceso - Cargo empleado.



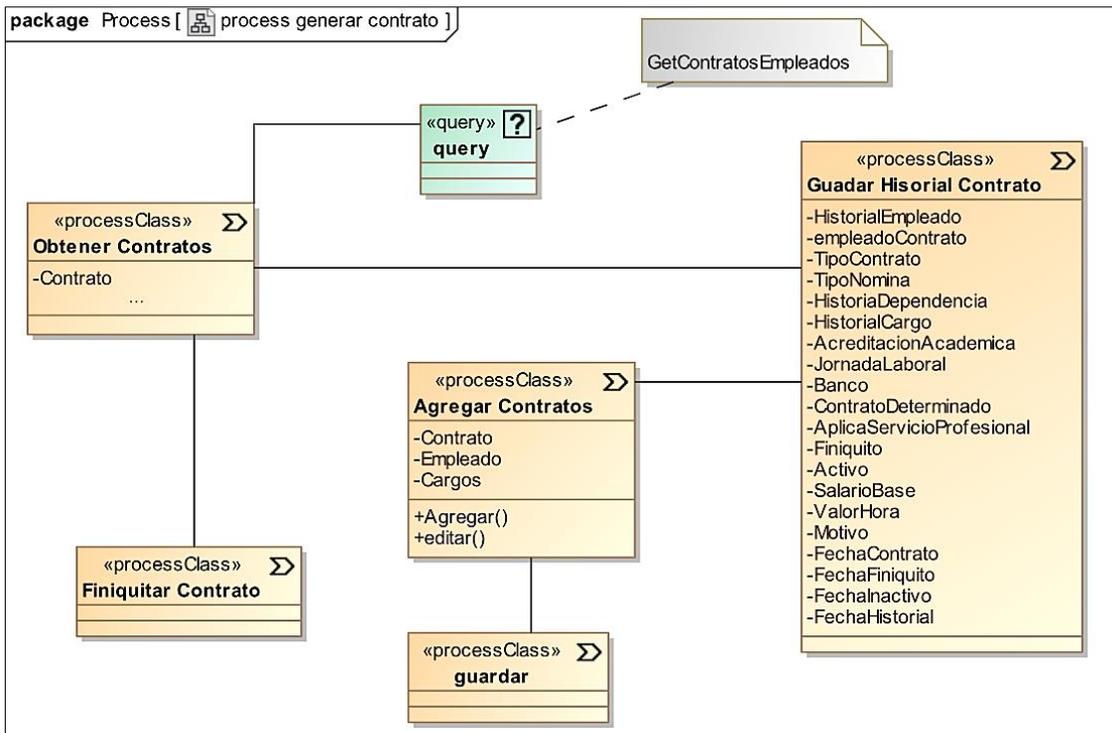
Fuente: Elaboración propia.

Diagrama de proceso - Administrar usuarios.



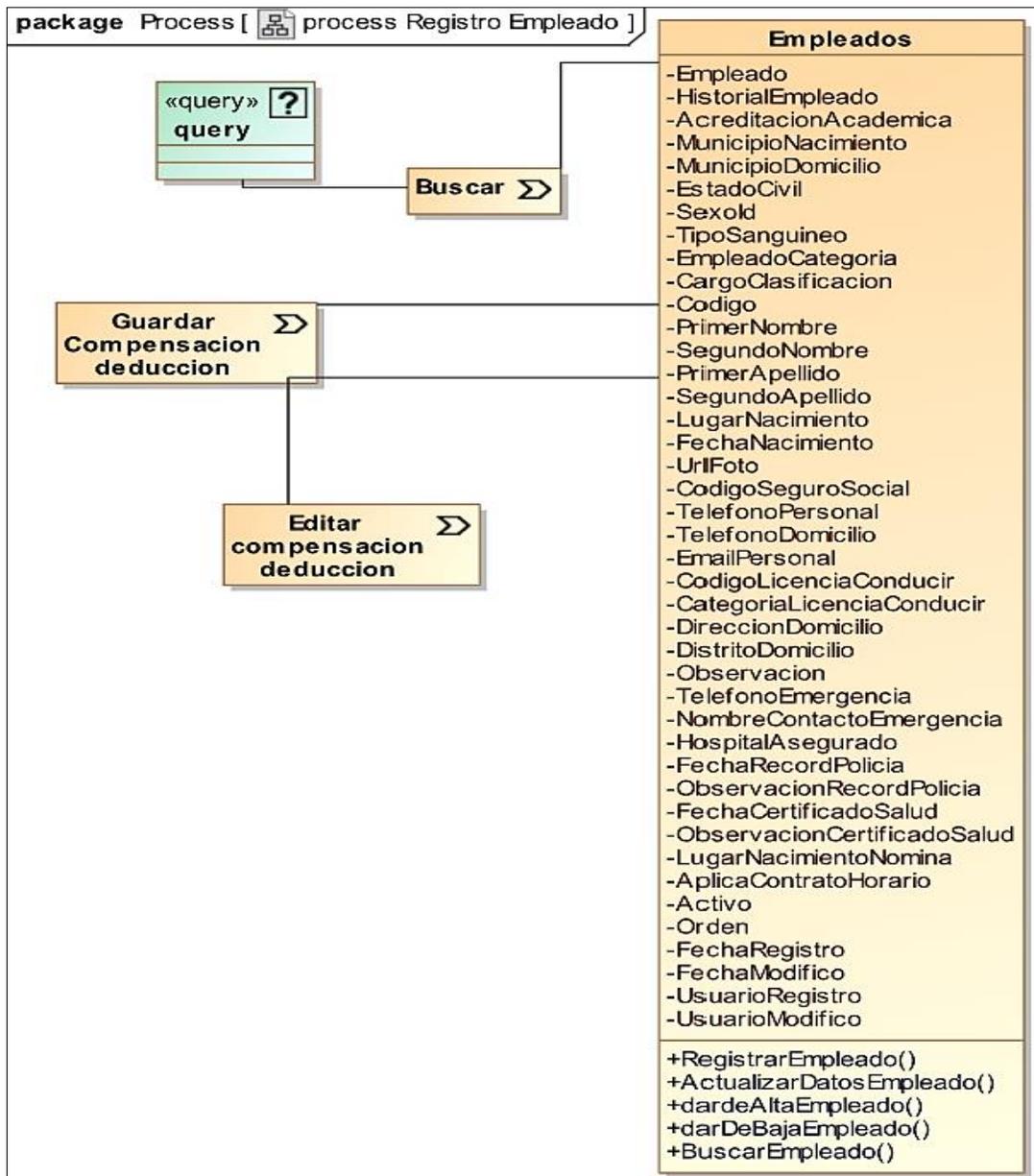
Fuente: Elaboración propia.

Diagrama de proceso - Generación de contrato.



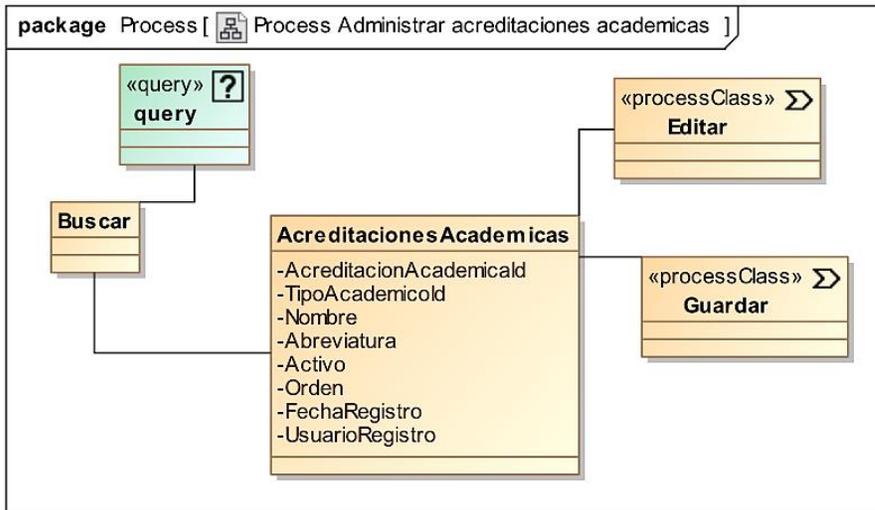
Fuente: Elaboración propia.

Diagrama de proceso - Registro empleado.



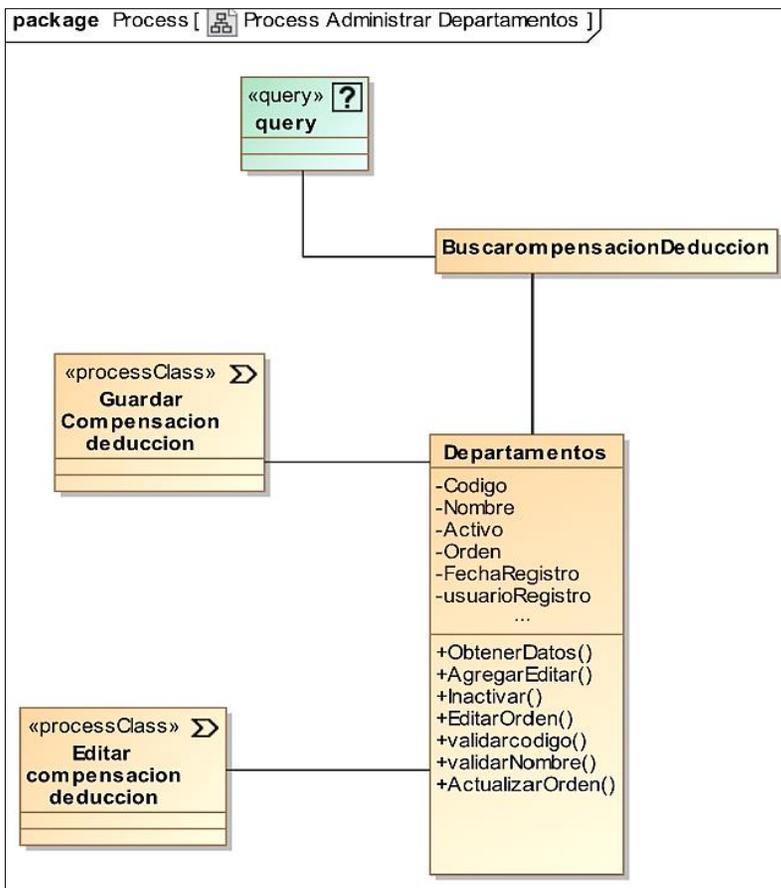
Fuente: Elaboración propia.

Diagrama de proceso - Administrar acreditaciones académicas.



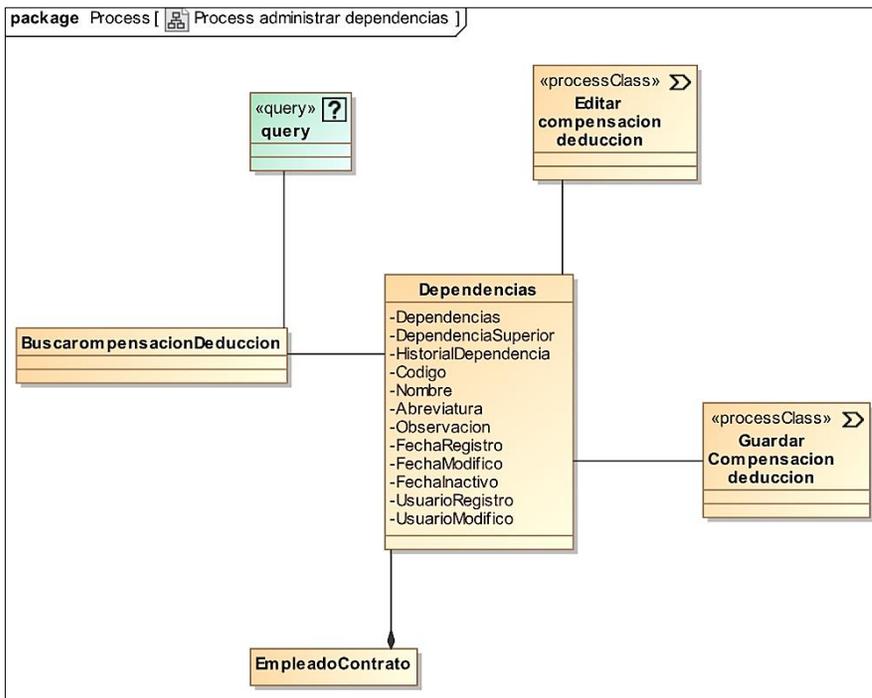
Fuente: Elaboración propia.

Diagrama de proceso - Administrar departamentos.



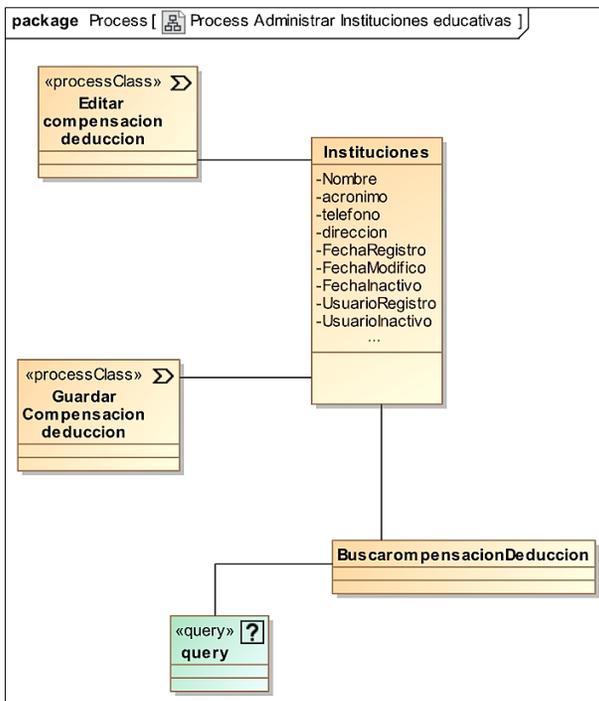
Fuente: Elaboración propia.

Diagrama de proceso - Administrar dependencias.



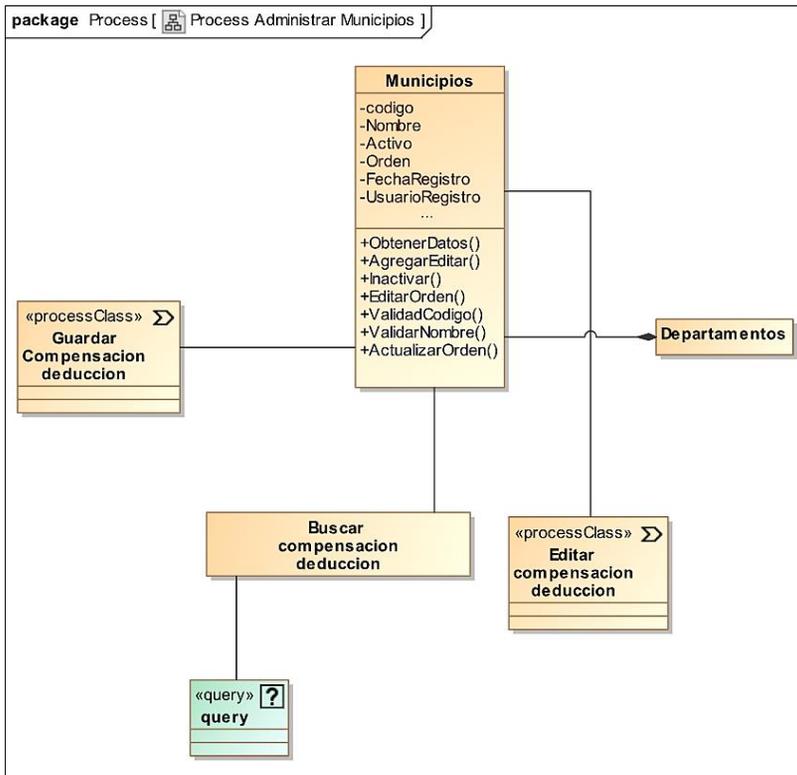
Fuente: Elaboración propia.

Diagrama de proceso - Administrar instituciones educativas.



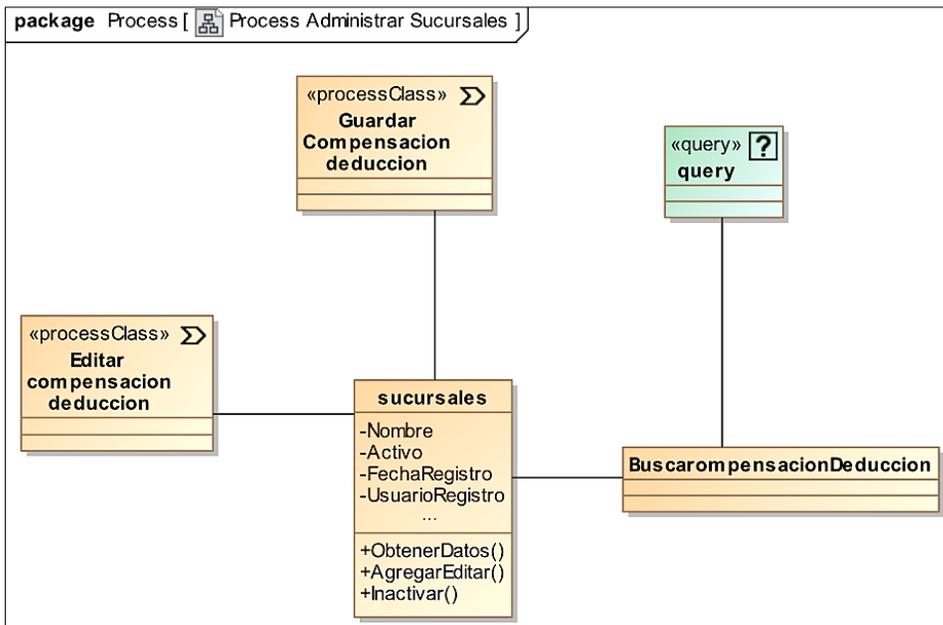
Fuente: Elaboración propia.

Diagrama de proceso - Administrar municipios.



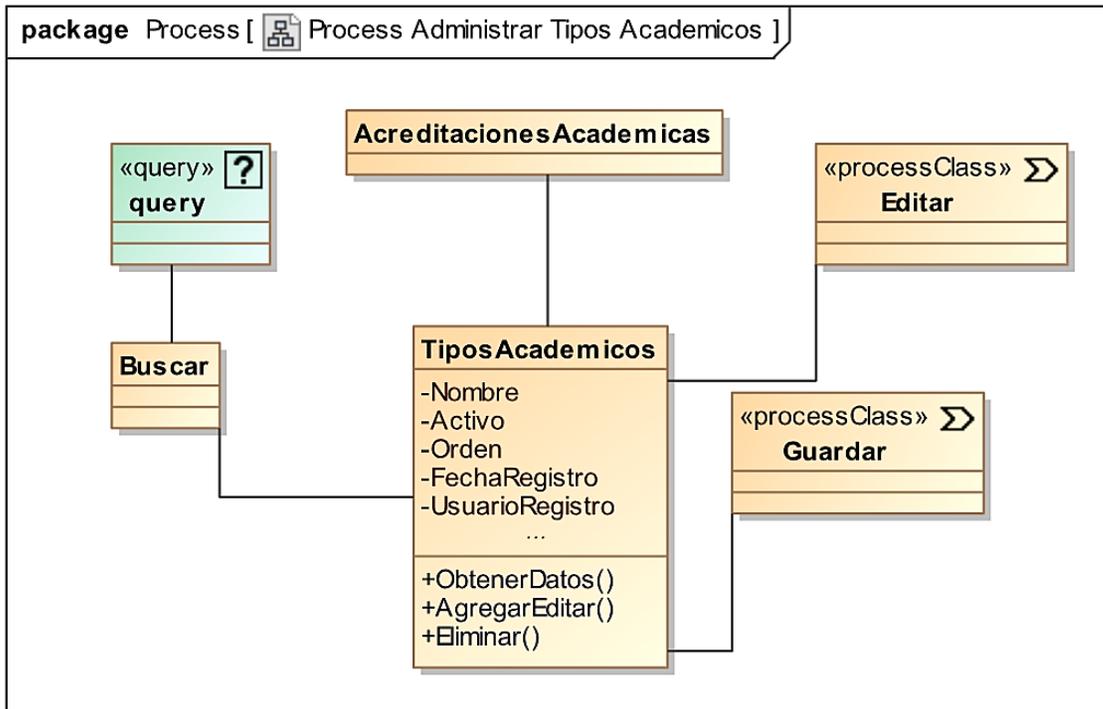
Fuente: Elaboración propia.

Diagrama de proceso - Administrar sucursales.



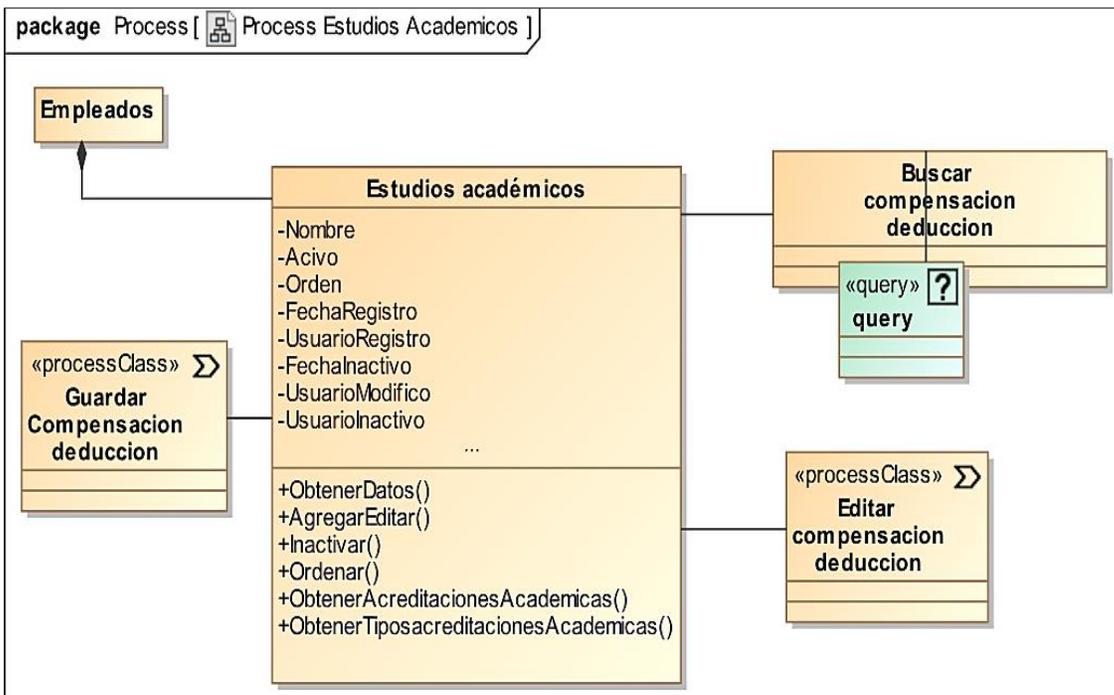
Fuente: Elaboración propia.

Diagrama de proceso - Administrar tipos académicos.



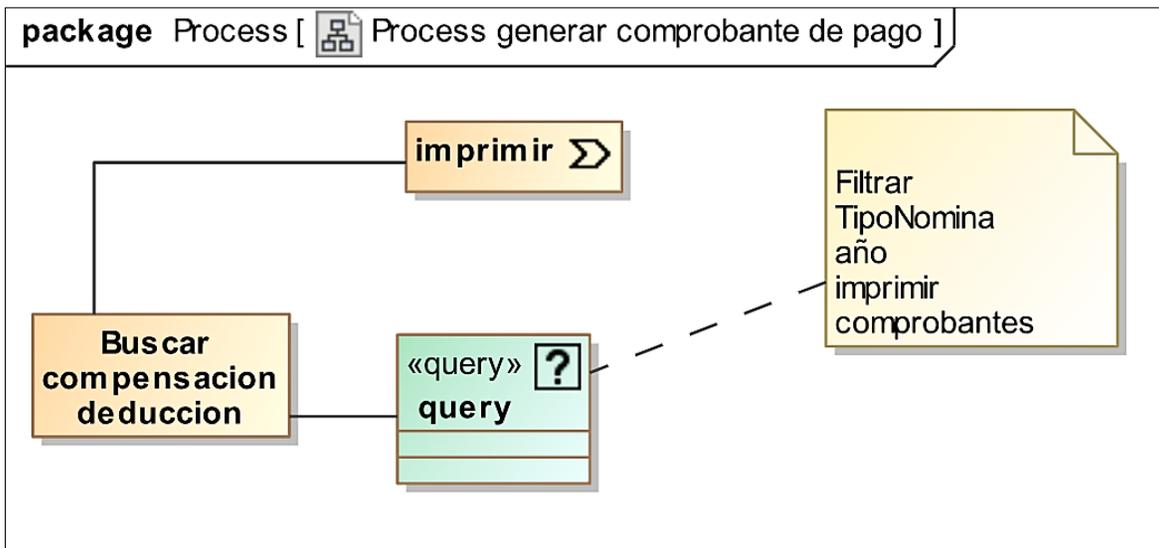
Fuente: Elaboración propia.

Diagrama de proceso - Estudios académicos.



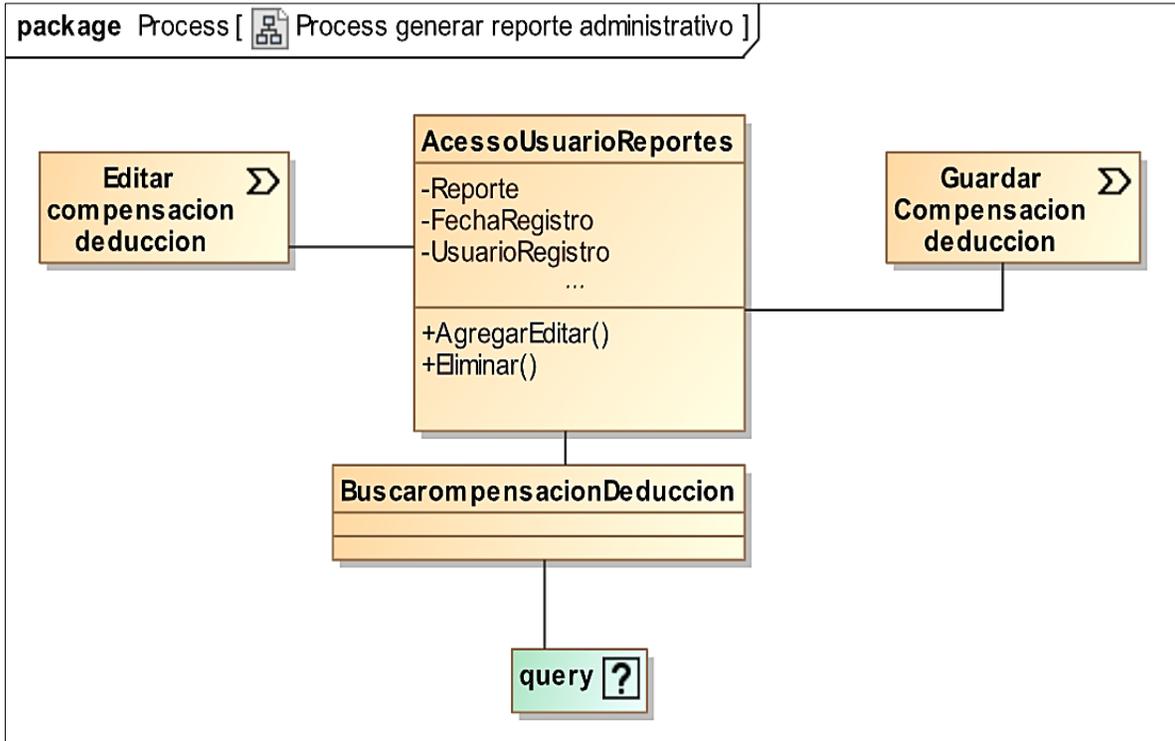
Fuente: Elaboración propia.

Diagrama de proceso - Generar comprobante de pago.



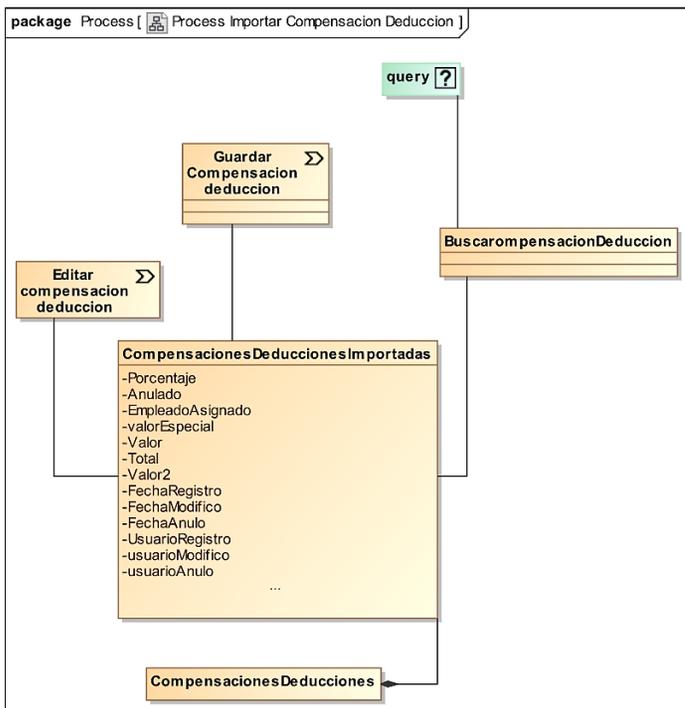
Fuente: Elaboración propia.

Diagrama de proceso - Generar reporte administrativo.



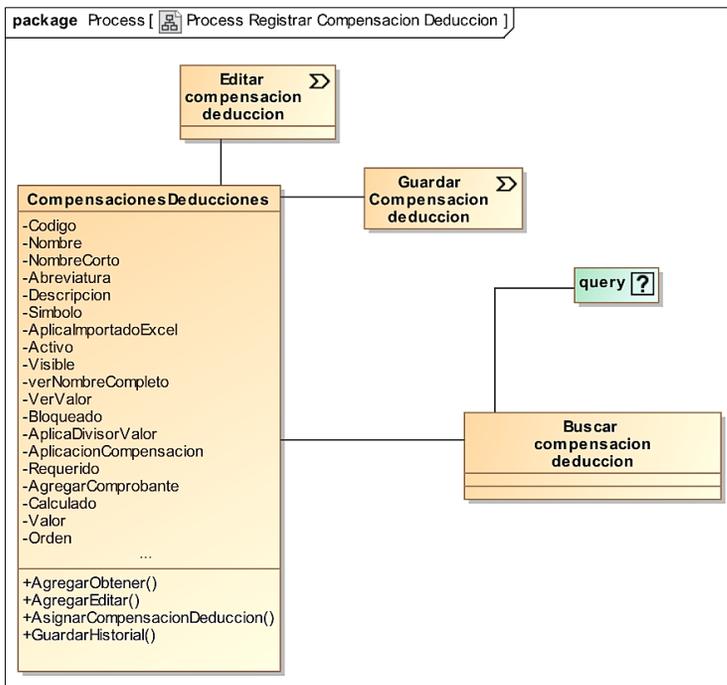
Fuente: Elaboración propia.

Diagrama de proceso - Importar compensación deducción.



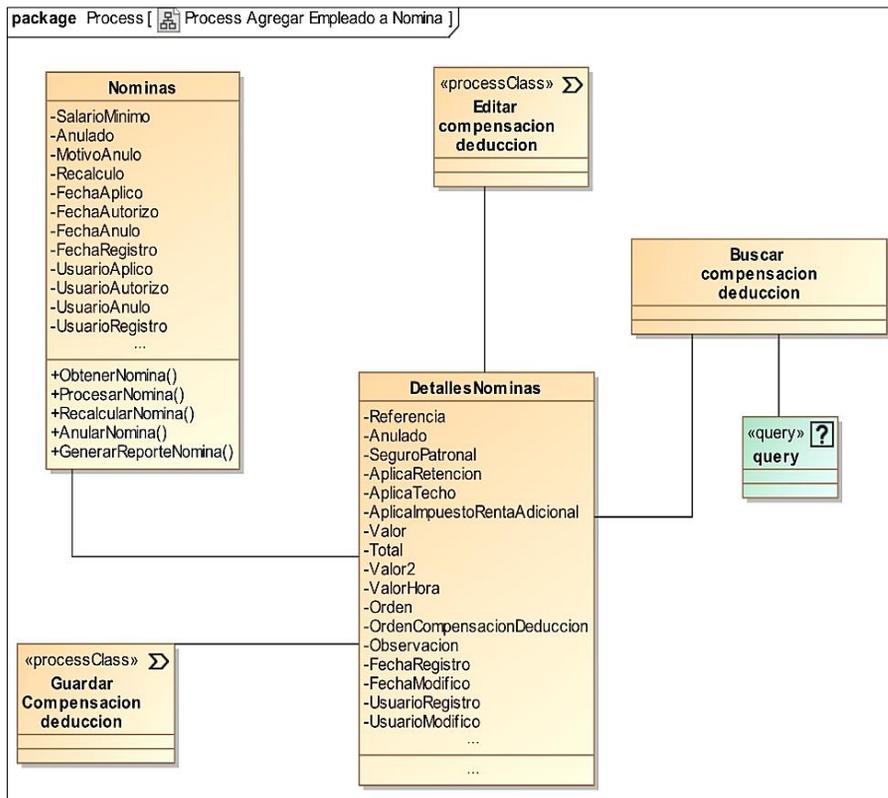
Fuente: Elaboración propia.

Diagrama de proceso - Registrar compensación deducción.



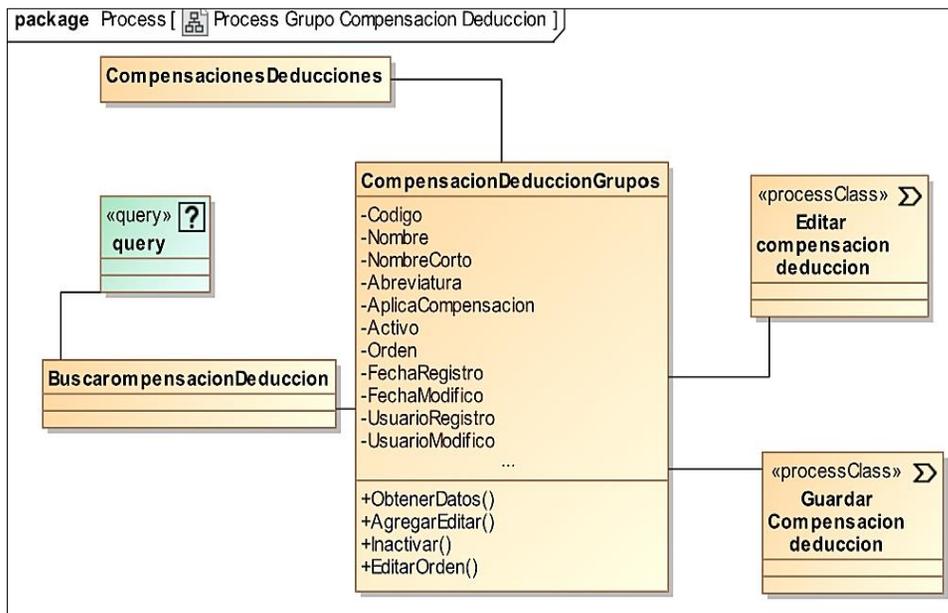
Fuente: Elaboración propia.

Diagrama de proceso - Agregar empleado a nómina.



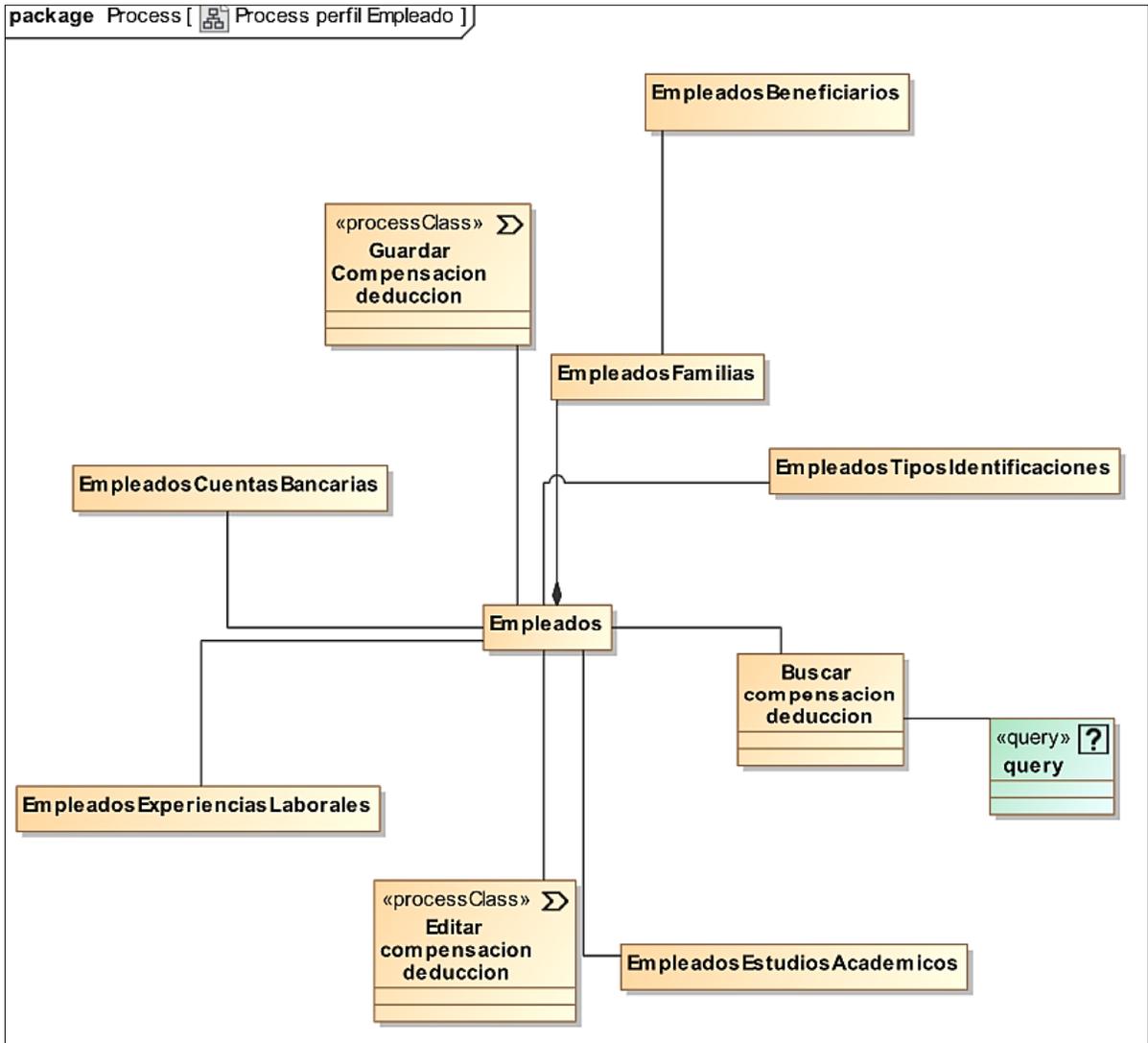
Fuente: Elaboración propia.

Diagrama de proceso - Grupo compensación deducción.



Fuente: Elaboración propia.

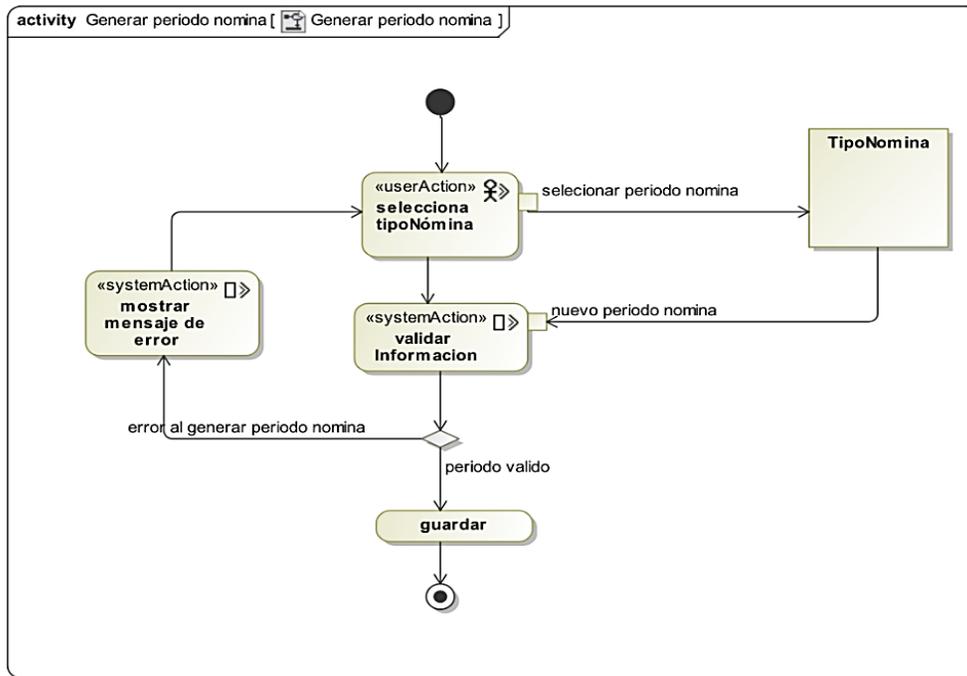
Diagrama de proceso - Perfil empleado.



Fuente: Elaboración propia.

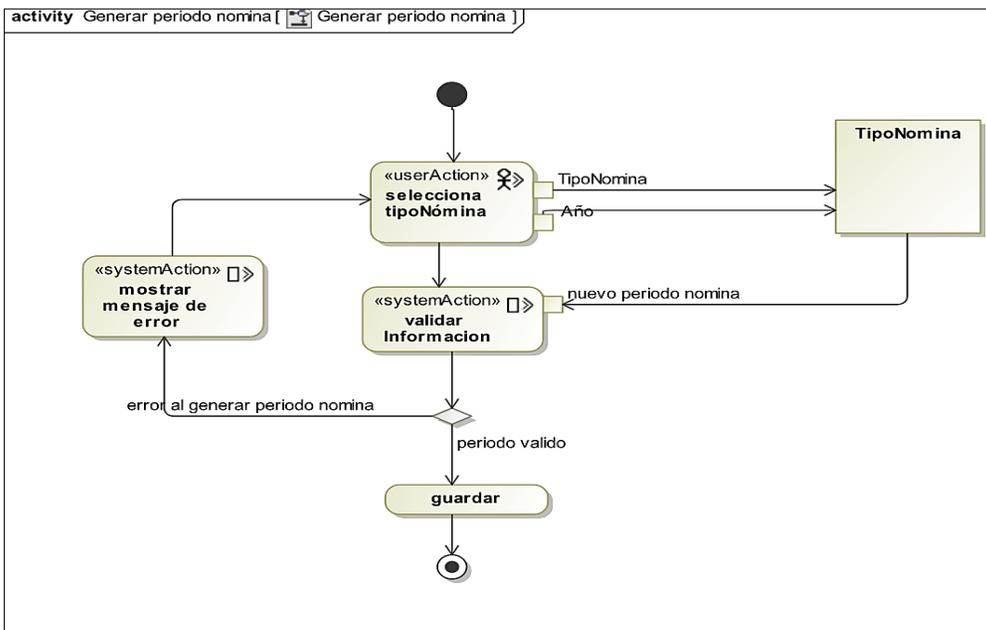
Anexo 6 : Diagramas de actividad.

Diagrama de actividad - Generar periodo nómina.



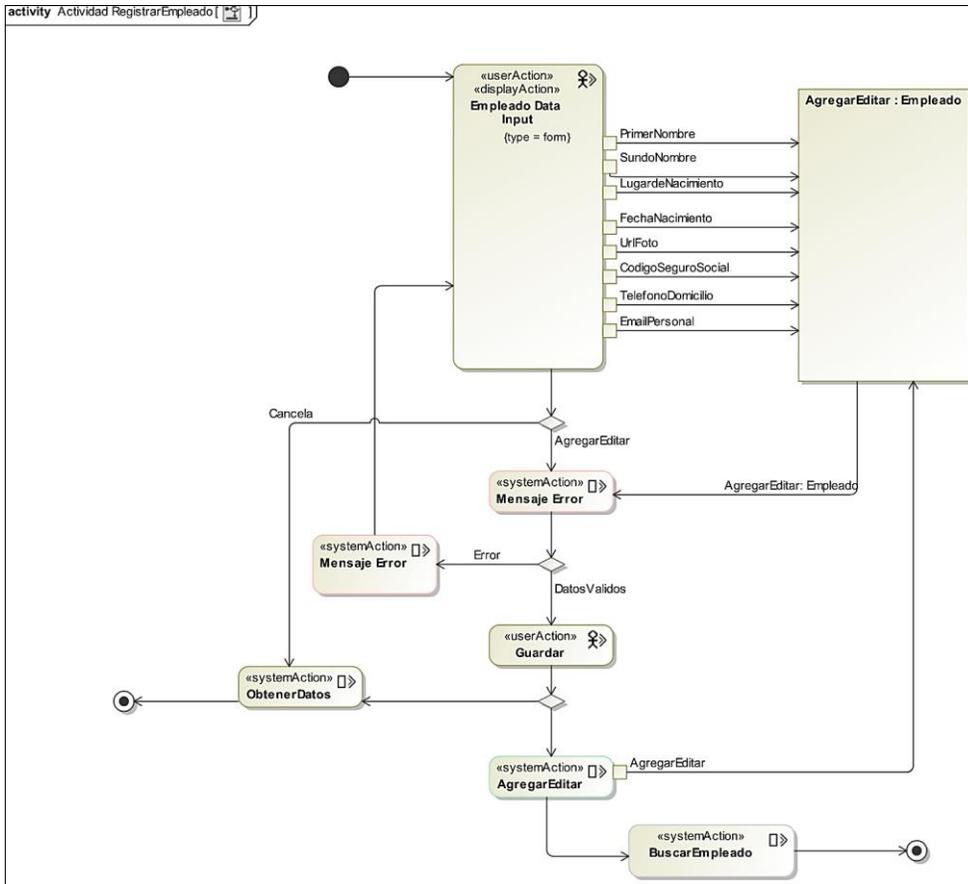
Fuente: Elaboración propia.

Diagrama de actividad - Generar Período de Nómina.



Fuente: Elaboración propia.

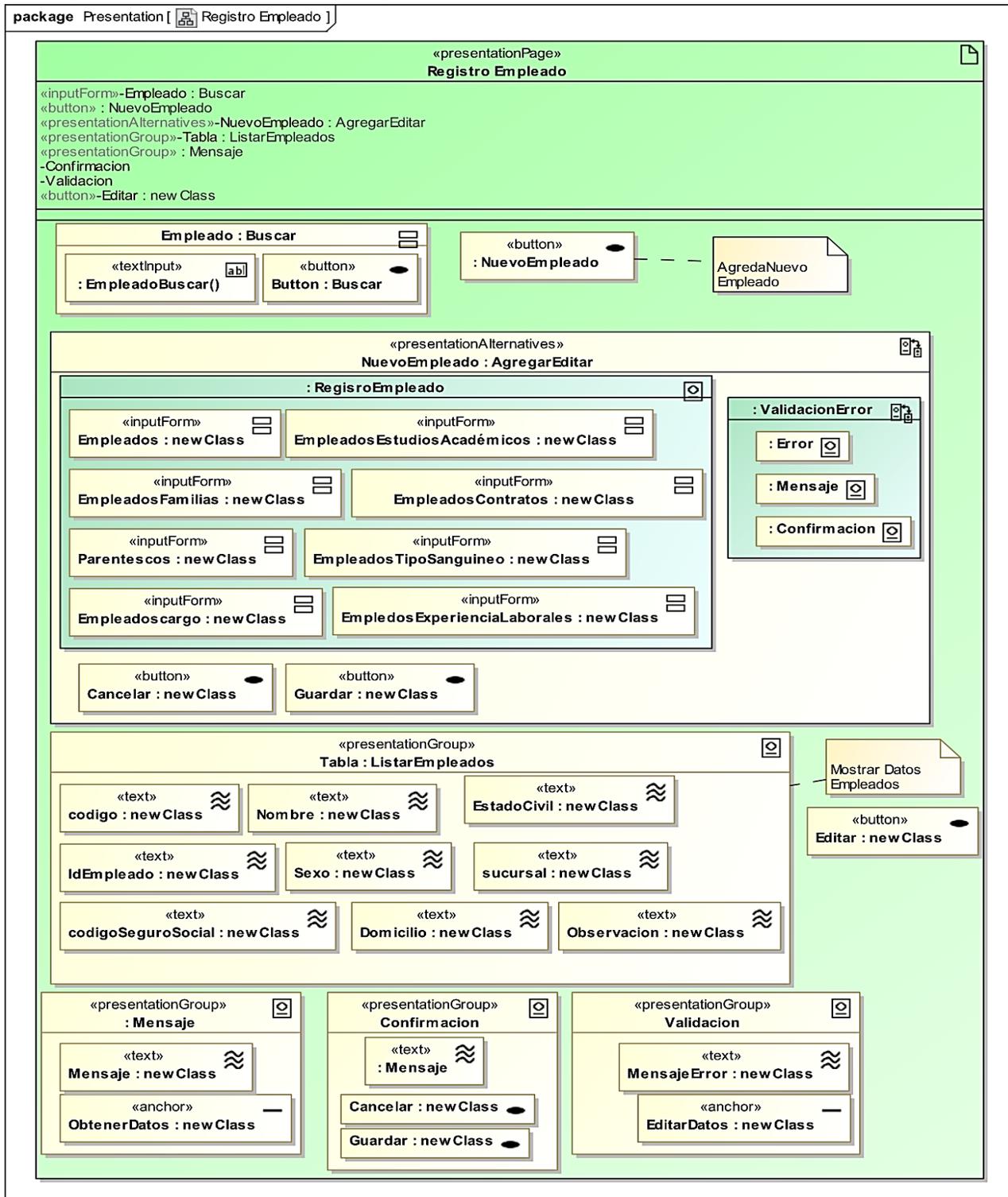
Diagrama de actividad – Registrar Empleado.



Fuente: Elaboración propia.

Anexo 7 : Diagrama de presentación.

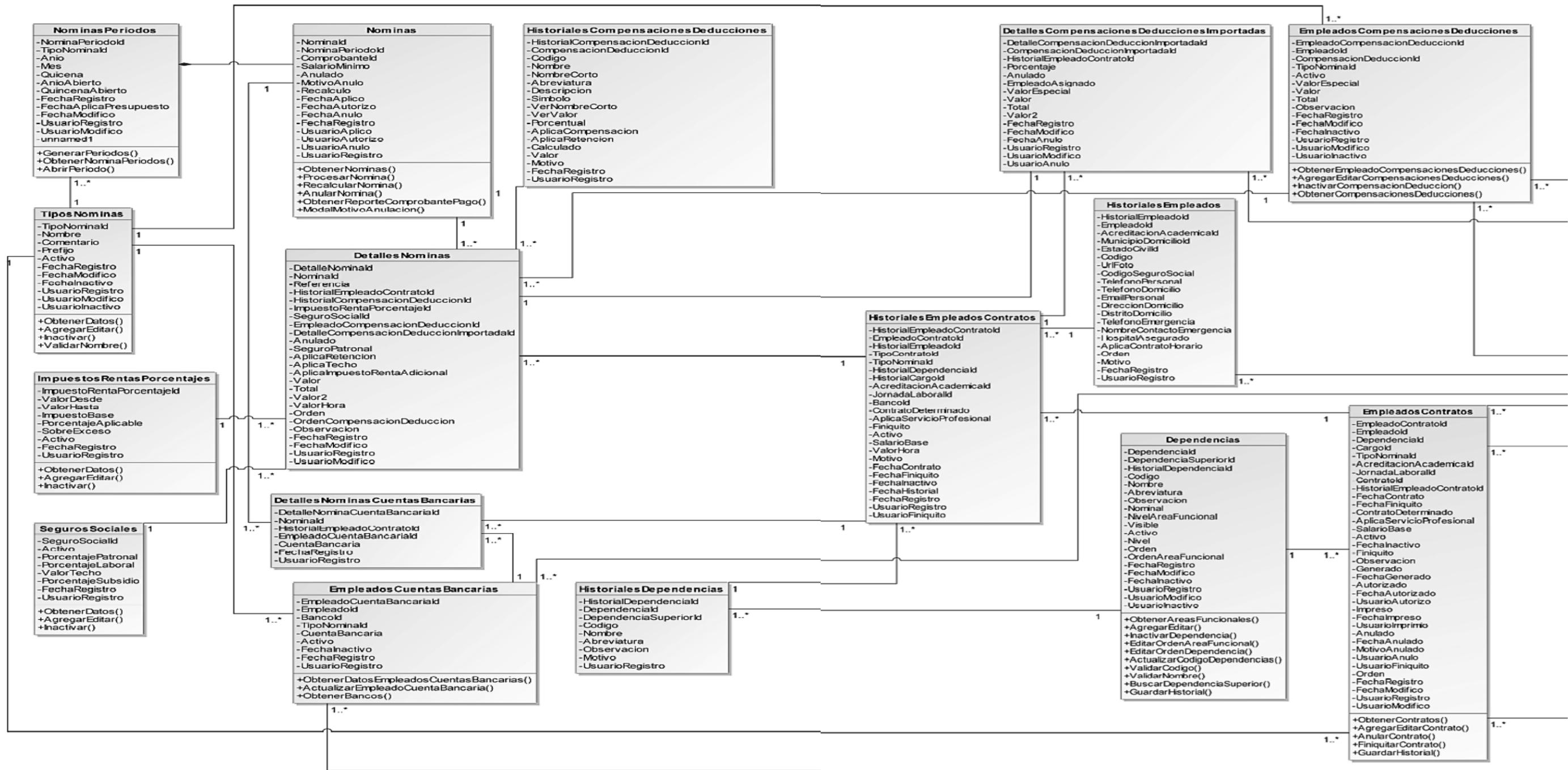
Diagrama de presentación - Registro Empleado.



Fuente: Elaboración propia.

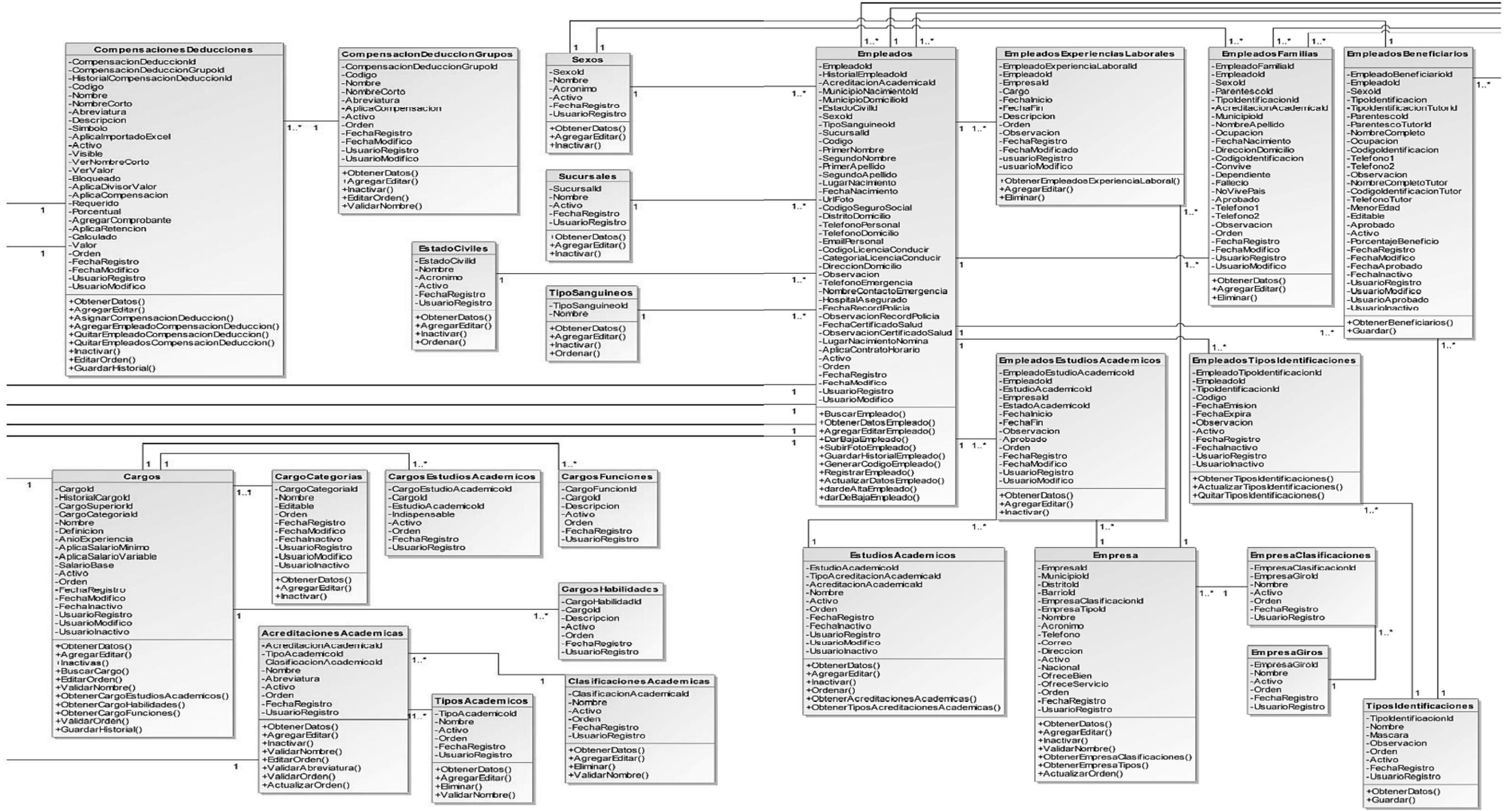
Anexo 8 : Modelo relacional.

(Parte 1).



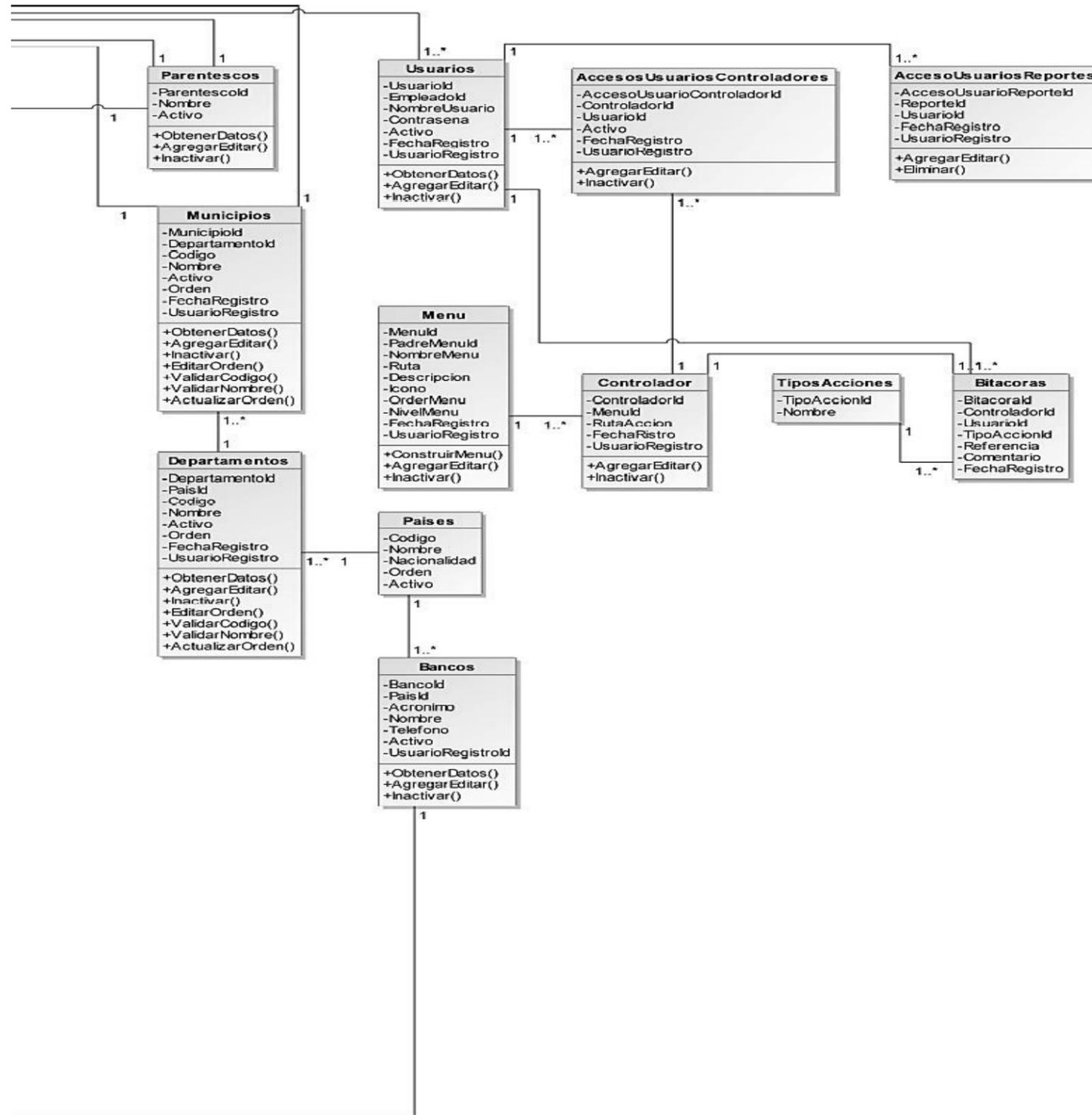
Fuente: Elaboración propia.

Anexo 8: Modelo relacional. (Parte 2).



Fuente: Elaboración propia.

Anexo 8: Modelo relacional. (Parte 3).



Fuente: Elaboración propia.