

Universidad de las Ciencias Informáticas

Facultad 6



Título: Sistema de Registro y Control de Cuadro

Trabajo de Diploma para optar por el título de
Ingeniero en Ciencias Informáticas

Autor(es): Alberto Gómez Figueroa
Alexis Lorenzo Herrera

Tutor(es): Yunier Miguel Sánchez López

Grupo de Investigación:
MINFAR

Julio 2008



"La revolución hace al hombre. Pero el hombre día a día tiene que forjar su espíritu revolucionario ."

Ernesto Che Guevara.

DECLARACIÓN DE AUTORÍA

Declaramos que somos los únicos autores de este trabajo y autorizamos a la Facultad 6 de la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio.

Para que así conste firmamos la presente a los ____ días del mes de _____ del año _____.

Alberto Gómez Figueroa

Firma del autor

Alexis Lorenzo Herrera

Firma del autor

Yunier Miguel Sánchez López

Firma del Tutor

AGRADECIMIENTOS

Alexis

Ha a mi tutor Yunier Miguel Sanchez Lopez por estar siempre cuando lo necesitamos, a Yosney Hernández Hernández , Yanet Del Risco Batista, Rolando Ramírez Concepción, Yuliet Galan Ramirez , a Enier Maceo Cisnero gracias por ayudarme y siempre evacuar las dudas que no eran pocas, a todos los mencionados gracias, sin ustedes no hubiera podido salir adelante.

Alberto

A nuestro tutor por la confianza depositada en la realización de este trabajo.

A las amistades por su dedicación y por su enorme apoyo. Por estar siempre en los momentos precisos.

A mis familiares que me infundieron la ética y el rigor que me guían por el transitar de la vida.

A todos aquellos que no vacilaron en ofrecernos su mano amiga en los momentos más difíciles para lograr este objetivo. A todos muchas gracias.

DEDICATORIA

Alexis

A mis padres, Yolanda de los Milagros Herrera Feblez, Alexis Lorenzo Herrera, Santiago Dortas Perez gracias por apoyarme siempre sin su ayuda no estuviera estado estudiando hoy. A mi abuela Yolanda y tias Margarita y Damaris que siempre me han ayudado, a Yadira , a mis primos Marbin y Mario que sigan estudiando para que se puedan hacer profesionales a mis Hermanos Alexander y Alain que sigan adelante y no se detengan en sus planes.

Alberto

A mis dos madres que quiero mucho Elena y Juana por ser mis consejeras, guias y darme su apollo siempre que lo necesite. A ustedes que se lo debo todo en esta vida.

A mi papa Armando quien ahora estaria muy orgulloso de mí.

A mis abuelos Enrique y Edilia que fueron mi motor impulsor y un ejemplo a seguir.

A mi tio Julio a quien admiro mucho por ser un ejemplo de padre para mí.

A mi hermano Ramon por enseñarme como ser un hombre mejor.

A mi primo William a quien quiero como un hermano, que siga estudiando para que se supere.

A todos mis amigos que de una forma u otra siempre me ayudaron. En especial a Siul y Leosbel quienes estuvieron siempre ahi cuando mas los necesite y con quien comparti mis mejores años de la universidad.

A nuestro eterno Comandante en Jefe y a la Revolución, que han hecho posible que todos tengamos igualdad de derechos y posibilidades para superarnos profesionalmente.

RESUMEN

La política de cuadros es llevada a cabo en nuestro país desde principios de la revolución, su principal cantera está en el pueblo. Los cuadros que pasan a formar parte de ella, su fragua se halla en el trabajo, en la lucha. Todos caracterizados por su lealtad a la patria, al socialismo, al internacionalismo y a los valores éticos tales como cultivar la vergüenza, el honor, la dignidad, respeto, combatir la apatía, la indolencia, el pesimismo, el hipercriticismo, el derrotismo, ser sincero, no ocultar ni tergiversar jamás la verdad, luchar contra la mentira, el engaño, la demagogia y el fraude. Es por eso que es necesario tener un sistema el cual sirva a todas las organizaciones del país. Para tener el Registro y control de estos a todos los niveles.

El siguiente trabajo tiene como propósito desarrollar una aplicación Web, la cual debe elevar el nivel de informatización en el proceso de gestión del Sistema de Registro de Control de Cuadros de las entidades del país, permitiendo que estos procesos se realicen de forma óptima, rápida y segura.

El desarrollo de este sistema se sostiene de la aplicación de metodologías y el uso de herramientas actuales que aseguran que su resultado sea un producto de software confiable, con un alto grado de calidad.

ÍNDICE

AGRADECIMIENTOS	I
DEDICATORIA	I
RESUMEN	I
INTRODUCCIÓN	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA	3
1.1 INTRODUCCIÓN	3
1.2 SISTEMAS DE INFORMACIÓN	3
1.2.1 <i>Otras definiciones.</i>	<i>3</i>
1.2.2 <i>Funciones de los Sistemas de Información</i>	<i>4</i>
1.2.3 <i>Tipos de Sistemas de Información.</i>	<i>6</i>
1.2.4 <i>Sistemas automatizados existentes vinculados al campo de acción</i>	<i>7</i>
1.3 SOLUCIÓN A TRAVÉS DE UNA APLICACIÓN DE GESTIÓN WEB.	8
1.3.1 <i>Arquitectura.</i>	<i>9</i>
1.3.2 <i>Patrones Arquitectónicos</i>	<i>9</i>
1.4 TÉCNICA Y TECNOLOGÍAS DEL LADO DEL CLIENTE.....	13
1.4.1 <i>Script del lado del cliente.</i>	<i>13</i>
1.4.2 <i>AJAX: Una técnica de desarrollo Web.</i>	<i>14</i>
1.5 EXTENSIBLE MARKUP LANGUAGE (XML).	15
1.6 TECNOLOGÍAS DEL LADO DEL SERVIDOR.	15
1.6.1 <i>Servidor Apache</i>	<i>16</i>
1.6.2 <i>Script del lado del servidor.....</i>	<i>16</i>
1.7 GESTIÓN DE DATOS.	17
1.7.1 <i>Sistema Gestor de Base de Datos.</i>	<i>17</i>
1.8 METODOLOGÍAS DE DESARROLLO DE SOFTWARE.....	19
1.8.1 <i>Lenguaje de Modelado.</i>	<i>22</i>
1.9 HERRAMIENTAS CASE.....	22
1.10 HERRAMIENTAS DE APOYO.....	23
1.11 CONCLUSIONES.	24

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA	25
2.1 INTRODUCCIÓN.	25
2.2 PROPUESTA DEL SISTEMA.	25
2.3 MODELADO DEL NEGOCIO.	26
2.4 ACTOR Y TRABAJADORES DEL NEGOCIO.....	26
2.5 DIAGRAMA DE CASOS DE USO DEL NEGOCIO.	27
2.6 DESCRIPCIONES DE LOS CASOS DE USOS DEL NEGOCIO.	27
2.6.1 Descripción de Casos de Uso: Aprobar Cuadro.....	28
2.6.1.1 Diagrama de Actividades<Aprobar Cuadro>	29
2.6.1.2 Modelo de Objetos<Aprobar Cuadro>.....	29
2.6.2 Descripción de Casos de Uso: Realizar Evaluación.....	30
2.6.2.1 Diagrama de Actividades < Realizar Evaluación >	31
2.6.2.2 Modelo de Objetos <Realizar Evaluación >.....	31
2.6.3 Descripción de Casos de Uso: Verificar Persona.....	32
2.6.3.1 Diagrama de Actividades <Verificar Persona>	33
2.6.3.2 Modelo de Objetos <Verificar Persona>.....	33
2.7 REQUERIMIENTOS FUNCIONALES	34
2.8 REQUERIMIENTOS NO FUNCIONALES	36
2.9 REGLAS DEL NEGOCIO A CONSIDERAR	38
2.10 SISTEMA.....	38
2.10.1 Actores del Sistema	38
2.10.2 Listado de casos de uso	38
2.10.3 Diagrama de Casos de Uso del Sistema.....	43
2.10.4 Paquetes y sus relaciones.	43
2.10.4.1 Paquete de Actualización.....	43
2.10.4.2 Paquete de Evaluación.	44
2.10.4.3 Paquete de Gestor de Nomencladores.....	45
2.10.5 Descripción Extendida de los CU del Sistema	48
2.11 CONCLUSIONES	54
CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA.....	55

3.1 INTRODUCCIÓN	55
3.2 ANÁLISIS	55
3.2.1 <i>Diagrama de clases del análisis.....</i>	<i>55</i>
3.2.1.1 <i>Caso de Uso Buscar Persona.</i>	<i>55</i>
3.2.1.2 <i>Caso de Uso Actualizar Datos Personales.....</i>	<i>55</i>
3.2.1.3 <i>Caso de Uso Actualizar Datos Políticos.</i>	<i>56</i>
3.2.1.4 <i>Caso de Uso Actualizar Familiares.</i>	<i>56</i>
3.2.1.5 <i>Caso de Uso Actualizar Trayectoria.</i>	<i>56</i>
3.2.1.6 <i>Actualizar Vehículo_Armamento.</i>	<i>57</i>
3.2.1.7 <i>Caso de Uso Realizar Reporte.....</i>	<i>57</i>
3.2.1.8 <i>Caso de Uso Actualizar Salud.....</i>	<i>58</i>
3.2.1.9 <i>Caso de Uso Actualizar Idioma</i>	<i>58</i>
3.3 DIAGRAMA DE CLASES DEL DISEÑO.....	58
3.3.1 <i>Caso de Uso Actualizar Trayectoria.....</i>	<i>61</i>
3.3.2 <i>Caso de Uso Actualizar Familiares</i>	<i>61</i>
3.3.3 <i>Caso de Uso Actualizar Estudios.....</i>	<i>62</i>
3.3.4 <i>Caso de Uso Actualizar Idioma</i>	<i>62</i>
3.3.5 <i>Caso de Uso Actualizar Salud.....</i>	<i>63</i>
3.3.6 <i>Caso de Uso Buscar Cuadro.</i>	<i>63</i>
3.3.9 <i>Caso de uso Actualizar Datos Personales</i>	<i>65</i>
3.3.10 <i>Caso de Uso Actualizar Datos políticos</i>	<i>65</i>
3.3.11 <i>Caso de uso Gestionar Cuadro.....</i>	<i>66</i>
3.4 DIAGRAMAS DE INTERACCIÓN.....	66
3.4.1 <i>Diagrama de Secuencia Actualizar Trayectoria Política</i>	<i>67</i>
3.4.2 <i>Diagrama de Secuencia Actualizar Estudios.....</i>	<i>70</i>
3.4.3 <i>Diagrama de Secuencia Actualizar Familiares</i>	<i>73</i>
3.5 MECANISMO DE DISEÑO DE ACCESO A DATOS	76
3.5.1 <i>Descripción de las clases.....</i>	<i>77</i>
3.6 DISEÑO DE LA BASE DE DATOS.....	78
3.7 PRINCIPIOS DE DISEÑO.	97
3.8 CONCLUSIONES.	99

CONCLUSIONES	100
RECOMENDACIONES.....	101
BIBLIOGRAFÍA.....	102
GLOSARIO.....	103

INTRODUCCIÓN

En nuestro país se lleva a cabo la política de Cuadros la cual cuenta con un sin numero de hombres y mujeres dispuestos a defender los principios de la revolución. La revolución se encarga de prepararlos y forjarlos. La política de cuadro tiene como objetivo seleccionar, preparar a los cuadros que nos representan a todos los niveles. Es por eso que se necesita el uso de una aplicación la cual permita controlar estos procesos.

Estos procesos se realizan utilizando sistemas informáticos desarrollados con tecnologías que no nos permiten emplear a fondo los avances de la informática y las comunicaciones, carece de actualización en tiempo real, las aplicaciones no son centralizadas, no hay conectividad con otros nodos las recuperaciones que se obtienen del sistema se realizan de forma no personalizada lo que provoca que la gestión de la información correspondiente a estos procesos no sea la mejor, impidiendo disponer de la misma cuando se requiere. Se determinó la necesidad de desarrollar una aplicación donde se integre toda la información referente a los cuadros de las diferentes organizaciones y organismos del país. Por lo que se hará uso de las tecnologías de la información en función de este proceso. Esta tarea se concibió para dar respuesta a la necesidad de registrar, controlar, evaluar y recuperar información en el menor tiempo posible y con la mayor veracidad.

Todo esto nos impone una concepción cualitativamente nueva y superior en cuanto al empleo de una herramienta automatizada capaz de satisfacer todas las necesidades que se lleven a cabo para controlar los cuadros en cualquier entidad de nuestro país. Como meta superior en este trabajo se propone alcanzar, la realización del análisis y diseño del Sistema de Registro y Control de Cuadros de manera que satisfaga la expectativa que se espera, se propone la realización de una aplicación Web utilizando la tecnología **LAPP** (Linux, Apache, PostgreSQL, PHP).

De lo expresado anteriormente se puede deducir que la no existencia de un sistema informático con un enfoque general del problema, trae consigo todo una serie de problemas informativos que disminuye la eficiencia del Registro y Control de los Cuadros, por lo cual surge el siguiente **problema científico**: Gestión del proceso de registro y control de cuadros durante la aprobación y Evaluación del Cuadro.

Este problema se enmarca en el **objeto de estudio**: Los Procesos de Registro y Control de Cuadros.

Para resolver el problema planteado se propone como **objetivo general**: Diseñar un sistema informático que favorezca los procesos de Registro y Control de Cuadros.

De acuerdo con la propuesta anterior se proponen los siguientes **objetivos específicos**:

1. Investigar a fondo la etapa del proceso de Registro y Control de Cuadros de las entidades del país.
2. Modelar los principales procesos del negocio
3. Realizar el análisis del sistema.
4. Diseñar el sistema.

El objetivo que delimita el **campo de acción**: Informatización del proceso de Registro y Control de Cuadros.

Para guiar la investigación se plantea la siguiente **hipótesis**: Si se desarrolla un sistema informático consistente se favorecerá el proceso de Registro y Control de Cuadros llevado a cabo en nuestro país.

Esta investigación estará guiada por las siguientes **tareas**:

1. Realizar una investigación profunda sobre el proceso de Registro y Control de Cuadros llevado a cabo en nuestro país.
2. Investigar la existencia de sistema informáticos empleados en estos procesos.

El siguiente trabajo tiene como propósito llevar a cabo las tareas previamente descritas para el desarrollo del producto propuesto a construir, el mismo está estructurado por tres capítulos.

En el **Capítulo 1** se tratan aquellos temas que constituyen la fundamentación teórica de la investigación a realizar, o sea el estado del arte, las principales tecnologías y herramientas contendientes en el mundo para el desarrollo de este tipo de aplicaciones.

En el **Capítulo 2** se aborda todo lo referente a las características del sistema, se modelan los procesos de negocio y se da una descripción de la solución propuesta, definiéndose los requisitos que debe cumplir la misma.

El **Capítulo 3** está dedicado al análisis y diseño del sistema en cuestión a través de los diferentes artefactos propuestos por la metodología de desarrollo de software aplicada para la solución de nuestro problema.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

1.1 Introducción

En el presente capítulo se realiza una revisión de los conceptos a los cuales se hará referencia en todo el contexto del trabajo. Se brinda información acerca del proceso de Registro y Control de Cuadros y se hace referencia a las principales características de algunas soluciones de software relacionadas con el sistema. Además se abordan importantes conceptos que son necesarios conocer para la creación de una aplicación de gestión Web y para enfocar correctamente en qué consiste la misma y el uso de las tecnologías. Por último se ofrece una visión de la metodología y las herramientas que se utilizarán durante el trabajo.

1.2 Sistemas de Información

Un Sistema de Información es un conjunto de componentes interrelacionados que reúnen u obtienen, procesan, almacenan y distribuyen información para apoyar la toma de decisiones y el control en una organización. La información se obtiene luego de procesar los datos. Las actividades del sistema de información son: entrada, procesamiento y salida de datos. La retroalimentación sirve para mejorar o controlar el funcionamiento [LAU 2002].

1.2.1 Otras definiciones.

Sistema de Información es un conjunto formal de procesos que, operando sobre una colección de datos estructurada según las necesidades de la empresa, recopilan, elaboran y distribuyen la información (o parte de ella) necesaria para las operaciones de dicha empresa y para las actividades de dirección y control correspondientes (decisiones) para desempeñar su actividad de acuerdo a su estrategia de negocio [AUD 1991].

El objetivo principal de los Sistemas de Información (SI) es ayudar al desempeño de las actividades en todos los niveles de la organización, mediante el suministro de la información adecuada, con la calidad suficiente, a la persona apropiada, en el momento y lugar oportunos, y con el formato más útil para el receptor. Para realizar la recolección de los datos debemos hacerlo de forma estructurada, así como normalizar la información que entra y sale, por ello deben participar todos los órganos implicados de la empresa, para que pueda ser entendida la información que llegue al Sistema de Información o la que

parta de él, por todos los integrantes de la organización a los cuales va dirigida, y les pueda servir a realizar las operaciones cotidianas, tomar decisiones o como método de control. Además afecta a las funciones y estrategias del negocio, pues es un elemento de la empresa que ayuda a conseguir los objetivos, y que puede ser utilizado como ventaja competitiva frente a otros rivales. En las definiciones anteriores de Sistema de Información se manejan gran cantidad de elementos de los cuales se infiere la complejidad de las interrelaciones entre ellos. En muchos casos esta complejidad es acentuada por elevados volúmenes de información que deben ser manejados. Esta dificultad ha traído consigo la necesidad de aprovechar los adelantos informáticos y tecnológicos para automatizar el funcionamiento de los Sistemas de Información y de esta manera disminuir su complejidad para los usuarios de dichos sistemas. El sistema informativo está comprendido por una serie de elementos que se detallarán a continuación:

Documentos: Manuales, los impresos, u otra información descriptiva que explica el uso y/o la operación del sistema. Conjunto de datos originales que conforman el conjunto de datos.

Procedimiento: Conjunto de pasos que define el uso específico de cada elemento del sistema o el contexto.

Bases de Datos: Es la colección de datos relacionados entre si, la cual es accedida por el software y es el resultado de relacionar la información.

Personas: Individuos que son usuarios u operadores del sistema.

Hardware: Es el conjunto de dispositivos electrónicos que proporciona la capacidad de computación y los dispositivos electromecánicos encargados de extraer o suministrar la información en/de los soportes magnéticos.

Software: Es el conjunto de programas de computación encargados de suministrar la información deseada al cliente, con la coordinación y relación requerida.

1.2.2 Funciones de los Sistemas de Información

El Sistema de Información lleva a cabo una serie de funciones que se pueden agrupar en cuatro grupos:

- ✓ Funciones de captación y recolección de datos.
- ✓ Almacenamiento de la información.
- ✓ Tratamiento de la información.
- ✓ Distribución de la Información.

Captación y Recolección de Datos:

Recoge la información externa (o del entorno) e interna, enviando dicha información a través del sistema de captura a los órganos del Sistema de Información, encargados de reagruparla, para evitar duplicidades e información inútil o ruido, la captación de información depende del tipo de empresa o del destino que se espera de la información, pero la empresa puede crear un servicio de vigilancia del entorno, este proceso de captación y recolección de datos debe realizarse de forma continuada, en las áreas o partes del entorno y/o empresa sujetas a más cambios, sin dejar de vigilar las áreas más estables, aunque de forma puntual.

Almacenamiento:

Una vez filtrada la información relevante, ésta se almacenará, puede ser en un lugar único (archivo central, sistema informático), accesible a todos los usuarios, o bien en los distintos departamentos, pero igualmente accesible a cualquier usuario que la necesite, y el sistema utilizado dependerá de las variables, tamaño, dispersión geográfica o especificidad de la información, quien determine el sistema a adoptar, también se puede realizar un sistema mixto, donde se centralicen la información que afecte de forma global a la empresa, y las específicas en cada departamento. El acceso a la recuperación de la información por parte de los usuarios, puede realizarse a través de códigos o claves que eviten a los miembros que no necesitan la información puedan acceder a ella.

Tratamiento de la Información:

Es la función clave del Sistema de Información tiene por objeto transformar los datos de la información almacenada en información significativa, para ofrecérsela a quien la necesite, en la medida y formato que el usuario requiera. Generalmente en esta función, se utilizan medios informáticos por su capacidad de almacenar y velocidad en el tratamiento, así como la reducción de costes que representan, esto ha hecho que se generalice su uso.

Distribución y Diseminación:

Es muy importante para la empresa que cada usuario posea la información requerida en el momento preciso y de una forma normalizada para su correcta interpretación, además existe la necesidad de que alguna información acerca de la empresa y su entorno sean conocidas por diferentes miembros de la

organización, para hacer frente con rapidez de forma conjunta a las situaciones que se presenten, y en las que se hace necesaria la resolución de problemas y adopción de decisiones de forma coordinada o cuando afecte a varias áreas de la empresa [MÉNDEZ 2002].

1.2.3 Tipos de Sistemas de Información.

Los sistemas de información se desarrollan con diferentes propósitos, los cuales dependen de las necesidades de la empresa. Por ejemplo, a) los sistemas de procesamiento de datos, b) los sistemas de información para la administración (MIS, Management Information Systems), y c) los sistemas de apoyo para la toma de decisiones (DSS, Decisions Support Systems), son diferentes tipos de sistemas de información computarizados. Estos se analizan y se diseñan mediante la aplicación de los conceptos y las técnicas del diseño y del análisis de sistemas. En cierto grado, esto también se aplica a los sistemas expertos. Se sintetizan a continuación cada uno de ellos.

Sistemas de Procesamiento de Datos.

Los sistemas de procesamiento de datos son aquellos sistemas de información computarizados que se desarrollan para procesar grandes volúmenes de información generada en las funciones administrativas, tales como la nómina (sueldos y jornales) o el control de inventarios (stock). Los sistemas de procesamiento de datos liberan del tedio y la rutina a las tareas que se realizan manualmente; sin embargo, el elemento humano sigue participando, al llevar a cabo la captura de la información requerida (la cual también puede estar automatizada: el código de barra, utilizado en supermercados, por ejemplo). Estos sistemas ejecutan periódicamente los programas de manera automática. Una vez preparados, escasamente se requiere el tomar decisiones. En términos generales, los sistemas de procesamiento de datos ejecutan las actividades de carácter rutinario de las empresas.

Sistemas informáticos para la administración.

Los sistemas de información para la administración (MIS) no sustituyen a los sistemas de procesamiento de datos, más bien se basan en los datos obtenidos por éstos. Los MIS son sistemas que se sustentan en la relación que surge entre las personas y las computadoras.

Los MIS requieren para su operación de: las personas, del software (programas de cómputo) y del hardware (computadoras, impresoras, etc.), lo cual es común para los sistemas de procesamiento también, obviamente. Estos tres componentes suelen abreviarse: MW, SW y HW (MW = “manware”).

Los MIS, sistemas de información para la administración, soportan un amplio espectro de tareas de las organizaciones, más aún que los sistemas de procesamiento de datos, incluyendo el análisis, decisiones y la toma de decisiones.

Los usuarios de los sistemas de información para la administración, utilizan una base de datos compartidos para tener acceso a la información. La base de datos almacena tanto datos como modelos, que ayudan al usuario en la interpretación y el uso de la información. Los MIS generan la información que eventualmente se utiliza en la toma de decisiones. Un sistema de información para la administración puede llegar a unificar ciertas funciones informáticas de la empresa, aunque el sistema como tal no exista dentro del negocio como una estructura individual.

Sistemas de Apoyo para la toma de Decisiones.

El sistema de apoyo para la toma de decisiones es un tercer tipo de sistema de información computarizada (DSS; Decisions Support Systems). Es similar a los sistemas de información tradicionales para la administración, en el sentido de que ambos dependen de una base de datos como fuentes de información; pero se distinguen del MIS, al hacer énfasis en el soporte en cada unas de las etapas de la toma de decisiones. Sin embargo la decisión en si depende de la persona responsable de la misma. Los Sistemas de Apoyo para la toma de decisiones se diseñan con una orientación hacia la persona o el grupo que los utilizará, y no como los sistemas de información tradicionales para la administración.

1.2.4 Sistemas automatizados existentes vinculados al campo de acción

En el país existen dos sistemas para el registro y control de los cuadros. Uno para el Registro y Control de cuadros del partido y el otro para el registro y control de cuadros del gobierno y para recuperar la información existen otros 5 sistemas, permitiendo solo hacer recuperaciones de manera estática. La información que maneja hasta el momento carece de actualización en tiempo real, y las aplicaciones no son centralizadas y no hay conectividad entre los terminales.

1.3 Solución a través de una aplicación de gestión Web.

El **concepto de aplicación Web** puede tener diferentes significados para cada desarrollador. Como consenso general se considera que una aplicación Web será un sistema Web donde la entrada del usuario (entrada de datos y navegación) afecta el estado del negocio. Esta definición intenta establecer que una aplicación Web es un sistema de software con estado de negocio, y por tanto puede elaborarse utilizando la metodología de trabajo que propone RUP, y utilizando el UML para los modelos.

Para UML una Aplicación Web es un software diseñado para automatizar procesos cuyo resultado se entrega a través de **Internet** o una **Intranet**.

En la actualidad y debido al auge de este tipo de aplicaciones, se reconocen diferentes tipos de categorías en dependencia del tipo de negocio que modelen:

- ✓ **Informacionales:** Periódicos, catálogos, manuales, libros electrónicos, etc.
- ✓ **Interactivas:** Formularios de registros, presentación de información personalizada, etc.
- ✓ **Transaccionales:** Tienda electrónica, bancos online, etc.
- ✓ **Flujo de Trabajo:** Planificación online, monitoreo, gerencia de inventario, etc.
- ✓ **Comunitarias:** Chat, mercados, subastas online, etc.
- ✓ **Portales:** Tiendas electrónicas, etc.

Una de las ventajas más significativas de las aplicaciones Web, consiste en su forma de instalación y distribución, ya que normalmente instalar una aplicación Web consiste en configurar los componentes del lado del servidor en la red y no es necesaria una instalación o configuración en el lado cliente.

Además el protocolo principal de comunicación en una aplicación Web es **HTTP** y **HTTPS** el cual funciona normalmente desconectado, es decir, el cliente hace una petición al servidor, este la procesa y le devuelve el resultado, terminando la comunicación entre estos.

Las **diferencias entre una aplicación Web y un sitio Web** (aunque este sea dinámico), radican en su utilización. Las aplicaciones Web implementan “lógica de negocios” y su uso cambia el estado del negocio. Es interés de los autores destacar estos términos pues define el enfoque del esfuerzo a la hora de modelarlas.

Entonces cabría preguntarse: **¿qué es la lógica de negocios?**

La lógica de Negocio son las reglas y los requerimientos que proporcionan el entorno para desarrollar las aplicaciones que pueden favorecer a la entidad a alcanzar sus metas.

Se entiende como reglas del negocio las restricciones que existen en el negocio dado; o sea acciones no válidas que la aplicación debe controlar para que el negocio no colapse, Ejemplo: No permitir crear facturas pertenecientes a clientes inexistentes, controlar que el saldo negativo nunca se produzca.

Los requerimientos del negocio, por su parte, son restricciones, usualmente impuestas por la propia entidad, que estas utilizan para operar en su ambiente de negocio.

Las aplicaciones Web ejecutan lógica de negocios por lo que a la hora de modelar es más importante enfocarse en esta lógica y en el estado del negocio, y no en los detalles de presentación.

1.3.1 Arquitectura.

El objetivo principal de la Arquitectura del Software es aportar elementos que ayuden a la toma de decisiones y al mismo tiempo, proporcionar conceptos y un lenguaje común que permitan la comunicación entre los equipos que participen en un proyecto. Para conseguirlo, la Arquitectura del Software construye abstracciones, materializándolas en forma de diagramas comentados.

La arquitectura en n capas está dada por la división del sistema en capas, separando los niveles de abstracción en sub-partes que se comunican entre sí e intercambian entradas y salidas para su funcionamiento, además, los cambios en una de las capas no afectan substancialmente el funcionamiento de las otras. Dentro de este estilo (arquitecturas en n capas), se va a utilizar la arquitectura en 3 capas definida por nuestro centro para el desarrollo del sistema, definiendo el mismo por capas lógicas: capa de presentación, lógica de negocio y capa de acceso a datos. Por último, se representará la BD como fuente de datos, no siendo esta una capa debido a que en el sistema no se tienen procedimientos almacenados, lo que implica que no hay procesamiento lógico.

- ✓ **Capa de presentación o capa de interfaz de usuario:** Es la forma de posibilitarle al usuario intercambiar con la aplicación. Las interfaces de usuario son implementadas utilizando formularios, controles u otro tipo de tecnología que permita procesar y dar formato a los datos de los usuarios, así como adquirir y validar los datos entrantes procedentes de los mismos.

- ✓ **Capa de lógica de negocio:** Es la parte más importante de una aplicación debido a que encapsula las responsabilidades y los procesos que en ella se comprenden. Su funcionamiento está basado en el uso de clases o componentes, y éstos, a su vez, por métodos que de una forma u otra establecen la relación entre los componentes de presentación y los de acceso a datos. Para acceder al subsistema de la capa de acceso a datos se presenta una interfaz denominada “Factoría Típica”, clase que implementa el patrón de diseño Factoría, que centraliza en una clase controladora la creación de objetos.
- ✓ **Capa de acceso a datos:** Capa final en el flujo e intercambio entre todas las capas. Permite conocer el formato de los datos que se intercambian a través de los componentes de acceso a datos y la capa de lógica de negocio. Usa la tecnología de acceso a datos PDO e implementa una interfaz de programación para la gestión de los datos.

1.3.2 Patrones Arquitectónicos

Los patrones arquitectónicos a diferencia de los estilos expresan un problema muy específico referente al diseño. Brindan soluciones generales a problemas comunes. La selección de un patrón arquitectónico es, por lo tanto, una decisión fundamental de diseño en el desarrollo de un sistema de software.

A continuación se describen aquellos patrones utilizados de acuerdo a las soluciones que brindan dentro del contexto del sistema.

Application Controller:

Este patrón se encarga de gestionar la interacción entre el usuario y la aplicación, dirigiendo el flujo de navegación y controlando el estado de la sesión.

Capas (Layers):

Consiste en estructurar aplicaciones que pueden ser descompuestas en grupos de sub-tareas, las cuales se clasifican de acuerdo a un nivel particular de abstracción. Esta descomposición en capas fomenta la reusabilidad, portabilidad, escalabilidad y ejecución de pruebas.

De esta manera se estructura la aplicación; en tres capas. Una capa de presentación la cual permite la interacción del usuario con el sistema a través de una interfaz. Una capa lógica de negocio en la cual residen las clases encargadas de controlar las peticiones provenientes de la capa de presentación, y una capa de acceso a datos la cual accede y/o modifica los mismos.

Active record y Active table:

Este patrón propone una tabla para cada clase entidad y clases intermedias que se encargan de llevar de objetos a tablas y de tablas a objetos, de hecho el framework hace invisible prácticamente para el usuario que esta trabajando con una base de datos.

Fachada:

El patrón de diseño Fachada se utiliza para proveer de una interfaz unificada sencilla que haga de intermediaria entre un cliente y una interfaz o grupo de interfaces más complejas. Fachada debe utilizarse sólo para crear clases sencillas, no clases que “sirvan para todo” o “lo hagan todo”.

Singleton:

El patrón Singleton aplica a situaciones en las cuales hay la necesidad de ser una sola instancia de una clase. El ejemplo más común de esto es una conexión de base de datos. Implementando este patrón permite a un programador hacer esta simple instancia fácilmente accesible a muchos otros objetos.

Factory:

El patrón Factory permita la instancia de objetos en tiempo de ejecución. Es llamado el patrón Factory puesto que es responsable de "manufacturar" un objeto.

Patrones GRASP

Los patrones GRASP (Patrones de Software para la Asignación General de Responsabilidades) basan su utilidad en definir normas o principios fundamentales en el diseño orientado a objetos y las responsabilidades de estos de acuerdo a su comportamiento. Facilitan la comprensión y sistematicidad de las soluciones implementadas.

Dentro de los patrones GRASP suelen destacarse al menos cinco (Experto, Creador, Alta cohesión, Bajo acoplamiento, Controlador). Siendo estos los más utilizados ya sea por el contexto en el que aparecen o las soluciones que brindan. De ahí que en la etapa de diseño se definió el uso de los principales patrones de asignación de responsabilidades con el objetivo de optimizar la implementación de la propuesta realizada.

1- Experto:

Consiste en asignarle una responsabilidad al experto en información, un caso de ello es cuando se buscan y se muestran personas o se realiza y muestra la ficha de una persona, el experto en la información es la clase controladora, que contiene a las personas con su información.

2- Creador:

Asignar la responsabilidad de crear una instancia a una clase que la contiene o la agrega, como un especialista puede crear reportes, etc.

3- Alta Cohesión:

Cuando se dice que una clase tiene alta cohesión, se quiere decir que el objeto tiene bien delimitadas sus responsabilidades. En la programación orientada a objetos, cada objeto tiene una responsabilidad que ha de cumplir dentro de un programa, en este caso todas las responsabilidades del especialista están bien definidas, es una clase altamente cohesiva.

La cohesión de un objeto significa cuán relacionadas están las acciones del objeto, de manera que sus responsabilidades sean pocas y limitadas a su función. Las responsabilidades de un objeto se traducen después en métodos que realizan acciones.

4- Bajo acoplamiento:

El bajo acoplamiento hace referencia a las relaciones que tienen los objetos entre sí dentro de un sistema. Teóricamente, cuando una serie de objetos tienen una relación con varios objetos, cuando estos últimos son cambiados, los objetos relacionados han de verse afectados necesariamente. El alto acoplamiento se da normalmente, cuando un objeto ha de saber demasiado detalles internos de otro para su

funcionamiento, es decir, se rompe el encapsulamiento de otro objeto. Por ello cuando menos acoplamiento y mayor cohesión, mejor diseñado estará el sistema.

5- Controlador:

Asignar un controlador para evitar que las clases del negocio y las vistas tengan una comunicación directa.

1.4 Técnica y tecnologías del lado del cliente.

Un cliente Web (navegador o browser) es una aplicación de software que permite al usuario recuperar y visualizar documentos de **Hipertexto**. La parte cliente de las aplicaciones Web está formada por el código **HTML** que forma la página Web, con opción a código ejecutable mediante los lenguajes de scripting de los navegadores (Java Script, VBScript, etc.) o mediante programas en Java (applets). La programación del lado del cliente tiene como principal ventaja que la ejecución de la aplicación se delega al cliente, con lo cual se evita recargar al servidor de trabajo. Entre los navegadores más populares están, el Netscape, Internet Explorer, Mozilla. Los lenguajes más comunes que se ejecutan en el cliente son: *VBScript* y *Java Script*.

1.4.1 Script del lado del cliente.

Una práctica muy difundida en el mundo de la programación web es emplear **Lenguajes Interpretados** del lado del cliente para añadir más funcionalidad a las aplicaciones, especialmente para crear una experiencia interactiva que no requiera recargar la página cada vez.

JavaScript

JavaScript es un lenguaje interpretado, con una sintaxis semejante a la de los lenguajes Java y C. Su diseño no le permite considerarse un lenguaje puramente orientado a objetos como es el caso de Java. Fue desarrollado por la empresa Netscape Communications. Es muy utilizado para controlar la apariencia y manipular los eventos dentro de la ventana del navegador Web así como para validar datos de entrada en las interfaces de las aplicaciones.

¿Por qué JavaScript?

Gracias a su compatibilidad con la mayoría de los navegadores modernos, es el lenguaje de programación del lado del cliente más utilizado. Con Javascript se pueden crear efectos en las páginas y definir interactividades con el usuario.

Todos los navegadores interpretan el código JavaScript integrado dentro de las páginas web. Para interactuar con una página web se provee al lenguaje JavaScript de una implementación del Document Object Model (DOM, ó Modelo de Objetos del Documento) que incorporan Konqueror, las versiones 6 de Internet Explorer y Netscape Navigator, Opera versión 7, y Mozilla desde su primera versión.

1.4.2 AJAX: Una técnica de desarrollo Web.

AJAX (JavaScript y XML Asíncronos) es una técnica de desarrollo web para crear aplicaciones interactivas. Éstas se ejecutan en el cliente, es decir, en el navegador del usuario, y mantiene comunicación asíncrona con el servidor en segundo plano. De esta forma es posible realizar cambios sobre la misma página sin necesidad de recargarla. Esto significa aumentar la interactividad, velocidad y usabilidad en la misma. Cargar y renderizar una página, luego mantenerse en esa página mientras scripts y rutinas van al servidor buscando, en background los datos son usados para actualizar la página, solo renderizando la página y mostrando u ocultando porciones de la misma.

La característica fundamental de AJAX es permitir actualizar parte de una página con información que se encuentra en el servidor sin tener que refrescar completamente la página. De modo similar podemos enviar información al servidor.

La complejidad se encuentra en que debemos dominar varias tecnologías:

- ✓ HTML o HTMLX
- ✓ CSS
- ✓ JavaScript
- ✓ DHTML Básicamente debemos dominar todos los objetos que proporciona el DOM.
- ✓ XML Para el envío y recepción de los datos entre el cliente y el servidor.
- ✓ PHP o algún otro lenguaje que se ejecute en el servidor (ASP.Net/JSP)

1.5 Extensible Markup Language (XML).

XML (Lenguaje de Marcas extensible) es Software libre, consiste en una serie de reglas, pautas para planificar formatos texto de manera que produzcan archivos que sean fácilmente generados y leídos (por un ordenador) que son inequívocos, y que evitan escollos comunes como la falta de extensibilidad, falta de soporte para la internacionalización o localismo, y la dependencia de una determinada plataforma.

Principales características

- Es una arquitectura más abierta y extensible. No se necesita versiones para que puedan funcionar en futuros navegadores. Los identificadores pueden crearse de manera simple y ser adaptados en el acto en internet/intranet por medio de un validador de documentos (parser).
- Mayor consistencia, homogeneidad y amplitud de los identificadores descriptivos del documento con XML
- Se podrá hacer el intercambio de documentos entre las aplicaciones tanto en el propio PC como en una red local o extensa.
- Datos compuestos de múltiples aplicaciones. La extensibilidad y flexibilidad de este lenguaje nos permitirá agrupar una variedad amplia de aplicaciones, desde páginas web hasta bases de datos.
- Gestión y manipulación de los datos desde el propio cliente web.
- Los clientes web pueden ser más autónomos para desarrollar tareas que actualmente se ejecutan en el servidor.

1.6 Tecnologías del lado del servidor.

La parte del servidor está formada por un programa o script que es ejecutado por el servidor web, y cuya salida se envía al navegador del cliente. Los principales beneficios de la aplicación de la tecnología de servidores de aplicación son la centralización y la disminución de la complejidad en el desarrollo de aplicaciones. Programar del lado del servidor tiene como gran ventaja que cualquier acción puede ejecutarse sin tener en cuenta el tipo de cliente, ya que la aplicación se ejecuta en el servidor que es un ambiente controlado. Una vez ejecutada la aplicación, el resultado que se envía al cliente.

1.6.1 Servidor Apache

Es un software libre servidor HTTP de código abierto para multiplataforma. Apache presenta entre otras características mensajes de error altamente configurables, bases de datos de autenticación y negociado de contenido, pero fue criticado por la falta de una interfaz gráfica que ayude en su configuración. Apache tiene amplia aceptación en la red, en el 2005 fue el servidor HTTP más usado, siendo el servidor http del 70% de los sitios web en el mundo.

Entre sus características destacan:

- Multiplataforma, lo que lo hace prácticamente universal. Es un servidor de Web conforme al protocolo HTTP/IP
- Modular: Puede ser adaptado a diferentes entornos y necesidades, con los diferentes módulos de apoyo que proporciona, y con la API de programación de módulos, para el desarrollo de módulos específicos. Incentiva la realimentación de los usuarios, obteniendo nuevas ideas, informes de fallos y parches para la solución de los mismos.
- Apache es una tecnología gratuita de código fuente abierta; el hecho de ser gratuita es importante pero no tanto como que se trate de código fuente abierto. Esto le da una transparencia a este software de manera que si queremos ver que es lo que estamos instalando como servidor, lo podemos saber, sin ningún secreto, sin ninguna puerta trasera.
- Extensible: Gracias a ser modular se han desarrollado diversas extensiones entre las que destaca PHP, un lenguaje de programación del lado del servidor.

1.6.2 Script del lado del servidor.

Los lenguajes de scripts que se ejecutan en el servidor Web son interpretados por alguna aplicación alojada en dicho servidor y que se envían al cliente en un formato comprensible para él, son independientes del cliente por lo que son mucho más flexibles en relación al cambio de un navegador a otro o con respecto a las versiones del mismo. El cliente solo verá el código HTML terminado.

PHP ¿Qué es PHP?

Es un lenguaje de fácil uso y se asemeja a los lenguajes más comunes de programación estructurada. Se trata de un lenguaje interpretado usado para la creación de aplicaciones para servidores, o creación de

contenido dinámico para sitios Web. Su interpretación y ejecución se da en el servidor, en el cual se encuentra almacenado el script, y el cliente sólo recibe el resultado de la ejecución. Últimamente también para la creación de otro tipo de programas incluyendo aplicaciones con interfaz gráfica usando las bibliotecas que posee. PHP es un lenguaje encapsulado dentro de los documentos HTML, de forma que se pueden introducir instrucciones PHP dentro de las páginas, debido a esto, el diseñador gráfico del Web puede trabajar de forma independiente al programador.

¿Por qué PHP?

Se pueden hacer grandes cosas con pocas líneas de código. El código PHP es mucho más legible que el de otros lenguajes. Viene acompañado por una excelente biblioteca de funciones que permite realizar cualquier labor: acceso a base de datos, encriptación, envío de correo, otros.

Al poderse encapsular dentro de código HTML se puede recoger el trabajo del diseñador gráfico e incrustar el código PHP posteriormente. Es multiplataforma, funciona en todas las plataformas que soporten Apache. Es software libre. Se puede obtener en la Web y su código está disponible bajo la licencia GPL.

1.7 Gestión de Datos.

Un Sistema Gestor de Bases de Datos (SGBD) o Sistema Gestor de Base de Datos Relacionales (RDBMS) es un conjunto de programas que permite a los usuarios crear y mantener una base de datos (BD), por lo tanto, el SGBD es un software de propósito general que facilita el proceso de definir, construir y manipular la BD para diversas aplicaciones así como la administración necesaria para mantenerlas operativas, mantener su integridad, consistencia, confidencialidad y seguridad. El propósito general de los sistemas de gestión de base de datos es el de manejar de manera clara, sencilla y ordenada un conjunto de información.

1.7.1 Sistema Gestor de Base de Datos.

PostgreSQL: Es un servidor de base de datos objeto relacional libre, liberado bajo la licencia BSD. Como muchos otros proyectos de código abierto, el desarrollo de PostgreSQL no es manejado por una sola compañía sino que es dirigido por una comunidad de desarrolladores y organizaciones comerciales las

cuales trabajan en su desarrollo, dicha comunidad es denominada el **PGDG** (PostgreSQL Global Development Group).

Algunas de sus principales características son:

.

1. Alta concurrencia

Mediante un sistema denominado MVCC (Acceso concurrente multiversión, por sus siglas en inglés) PostgreSQL permite que mientras un proceso escribe en una tabla, otros accedan a la misma tabla sin necesidad de bloqueos. Cada usuario obtiene una visión consistente de lo último a lo que se le hizo *commit*. Esta estrategia es superior al uso de bloqueos por tabla o por filas común en otras bases, eliminando la necesidad del uso de bloqueos explícitos.

2. Amplia variedad de tipos nativos

PostgreSQL provee nativamente soporte para:

- Números de precisión arbitraria.
- Texto de largo ilimitado.
- Figuras geométricas (con una variedad de funciones asociadas)
- Direcciones IP (IPv4 e IPv6).
- Arreglos.

3. Claves ajenas también denominadas Llaves ajenas o Llaves Foráneas.

4. Disparadores.

Un disparador se define en una acción específica basada en algo ocurrente dentro de la base de datos. En PostgreSQL esto significa la ejecución de un procedimiento almacenado basado en una determinada acción sobre una tabla específica. Ahora todos los disparadores se definen por seis características:

El nombre del disparador; El momento en que el disparador debe arrancar; El evento del disparador deberá activarse sobre; La tabla donde el disparador se activara; La frecuencia de la ejecución; La función que podría ser llamada. Entonces combinando estas seis características, PostgreSQL le permitirá crear una amplia funcionalidad a través de su sistema de activación de disparadores.

5.Vistas.

- ✓ Integridad transaccional.
- ✓ Herencia de tablas.
- ✓ Tipos de datos y operaciones geométricas.

1.8 Metodologías de desarrollo de software.

Un proceso de software detallado y completo suele denominarse “Metodología”. Sin embargo existen numerosas propuestas metodológicas que inciden en distintas dimensiones del proceso de desarrollo. Vale destacar que todo desarrollo de software es riesgoso y difícil de controlar, por lo que es necesario aplicar una metodología que permita obtener como resultado un producto de calidad. Una definición acertada, en correspondencia con el objetivo de la investigación es la abordada por Madison donde define metodología como “Conjunto de filosofías, fases, procedimientos, reglas, técnicas, herramientas, documentación y aspectos de formación para los desarrolladores de Sistemas de Información”. Las metodologías guían el proceso de desarrollo y la clave del éxito de un proyecto de software es la elección correcta de la metodología, que puede conducir al programador a desarrollar un buen sistema de software, logrando el crecimiento de la calidad y la productividad del producto. La elección de la metodología adecuada es más importante que utilizar las mejores y más potentes herramientas.

Metodología para el desarrollo unificado, Rational Unified Process (RUP)

El Proceso Unificado de Desarrollo (**RUP**) es un proceso de ingeniería de software que mejora la productividad del equipo de trabajo y entrega las mejores prácticas del software a todos los miembros del mismo. RUP, es una metodología basada en un pequeño grupo de principios claves: el equipo de un proyecto de software debe planificar el desarrollo; debe conocer hacia donde se dirige; debe documentar el proyecto de una manera perdurable y extensible. RUP es un proceso de desarrollo de software donde durante el paso por las diferentes etapas de desarrollo se van a “transformar los requerimientos de los usuarios en un sistema software”. La metodología RUP, llamada así por sus siglas en inglés Rational Unified Process, divide en 4 fases el desarrollo del software.

1. **Fase de Conceptualización:** El Objetivo en esta etapa es determinar la visión del Proyecto.
2. **Fase de Elaboración:** En esta etapa el objetivo es determinar la arquitectura óptima.
3. **Fase de Construcción:** En esta etapa el objetivo es llegar a obtener la capacidad Operacional inicial.
4. **Fase de Transición:** El objetivo es llegar a obtener el reléase del proyecto.

Cada una de estas etapas es desarrollada mediante el ciclo de iteraciones. Los objetivos de una iteración se establecen en función de la evaluación de las iteraciones precedentes. Ver Figura 1

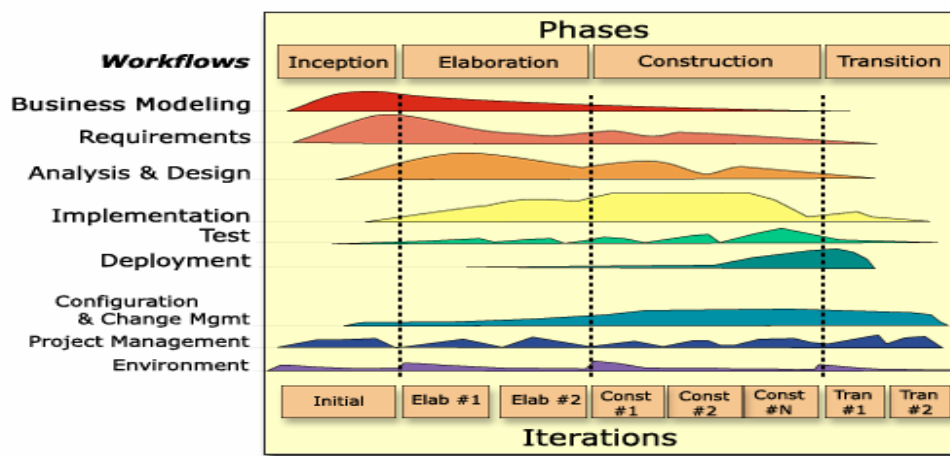


Figura 1. Fases e Iteraciones de la Metodología RUP.

Los elementos del RUP son:

1. Actividades, son los procesos que se llegan a determinar en cada iteración.
2. Trabajadores, vienen a ser las personas o entes involucrados en cada proceso.
3. Artefactos, un artefacto puede ser un documento, un modelo, o un elemento de modelo.

Características del Proceso Unificado

RUP es un proceso de desarrollo de software de forma tal que se asignan tareas y responsabilidades cuyos objetivos son asegurar la producción de software de calidad dentro de plazos y presupuestos predecibles. El ciclo de vida de RUP se caracteriza por ser: Dirigido por los casos de uso, Centrado en la arquitectura, Iterativo e incremental.

Dirigido por casos de uso:

Los casos de uso reflejan lo que los usuarios futuros necesitan y desean, lo cual se capta cuando se modela el negocio y se representa a través de los requerimientos. Un caso de uso es un fragmento de funcionalidad del sistema que proporciona al usuario un resultado importante. Representan requerimientos funcionales. Los casos de uso guían la arquitectura del sistema y la arquitectura del sistema influye en la selección de los casos de uso.

Centrado en la arquitectura:

La arquitectura muestra la visión común del sistema completo en la que el equipo de proyecto y los usuarios deben estar de acuerdo, por lo que describe los elementos del modelo que son más importantes para su construcción, los cimientos del sistema que son necesarios como base para comprenderlo, desarrollarlo y producirlo económicamente. RUP se desarrolla mediante iteraciones, comenzando por los CU relevantes desde el punto de vista de la arquitectura. La arquitectura se refleja en los casos de uso pues cada producto tiene tanto una función como una forma, ninguna es suficiente por si sola.

Iterativo e Incremental:

RUP propone dividir el trabajo en partes más pequeñas o mini proyectos, donde cada mini proyecto es una iteración que resulta en un incremento. Las iteraciones hacen referencia a pasos en los flujos de trabajo, y los incrementos, al crecimiento del producto. Cada iteración se realiza de forma planificada es por eso que se dice que son mini proyectos. Una iteración es una secuencia de actividades con un plan establecido y criterios de evaluación las cuales están estrechamente relacionadas, cuyo resultado es una versión del software.

Beneficios de la iteración:

1. Reduce el coste del riesgo al coste de un solo incremento.
2. Menos riesgo de no sacar el producto al mercado en fecha.
3. Acelera el ritmo de desarrollo.
4. Las necesidades del usuario y correspondientes requisitos no pueden definirse completamente al principio. Se requieren iteraciones sucesivas.

1.8.1 Lenguaje de Modelado.

El Lenguaje Unificado de Modelado (**UML**) prescribe un conjunto de notaciones y diagramas estándar para modelar sistemas orientados a objetos, y describe la semántica esencial de lo que estos diagramas y símbolos significan. UML se puede usar para modelar distintos tipos de sistemas: sistemas de software, sistemas de hardware, y organizaciones del mundo real. UML ofrece nueve diagramas en los cuales modelar sistemas:

1. Diagramas de Casos de Uso para modelar los procesos.
2. Diagramas de Secuencia para modelar el paso de mensajes entre objetos.
3. Diagramas de Colaboración para modelar interacciones entre objetos.
4. Diagramas de Estado para modelar el comportamiento de los objetos en el sistema.
5. Diagramas de Actividad para modelar el comportamiento de los Casos de Uso, objetos u Operaciones.
6. Diagramas de Clases para modelar la estructura estática de las clases en el sistema.
7. Diagramas de Objetos para modelar la estructura estática de los objetos en el sistema.
8. Diagramas de Componentes para modelar componentes.
9. Diagramas de Implementación para modelar la distribución del sistema.

1.9 Herramientas CASE.

Las Herramientas **CASE** son diversas aplicaciones informáticas destinadas a aumentar la Productividad en el desarrollo de software reduciendo el coste de las mismas en términos de tiempo y de dinero. Estas herramientas pueden ayudar en todos los aspectos del ciclo de vida de desarrollo del software en tareas como el proceso de realizar un diseño del proyecto, calculo de costes, implementación de parte del código automáticamente con el diseño dado, compilación automática, documentación o detección de errores entre otros.

Visual Paradigm

El Visual Paradigm para UML es una herramienta CASE visual que ayuda a construir aplicaciones rápidamente, mejor y económicamente. Utiliza “UML”: como lenguaje de modelado, siguiendo el estándar UML 2.1. Esta herramienta tiene unas características gráficas muy cómodas que facilitan la realización de los diagramas de modelado que sigue el estándar de UML, que son: Diagramas de clases, Casos de Uso,

Comunicación, Secuencia, Estado, Actividad, Componentes, etc. La herramienta Visual-Paradigm incorpora una funcionalidad de análisis textual, facilitando la interacción directa con un enunciado escrito en lenguaje natural, es decir, permite relacionar elementos presentes en un enunciado con los diagramas UML correspondientes, aunque no contempla el tratamiento directo en el texto de la relación entre los componentes del enunciado.

Beneficios:

1. Navegación intuitiva entre el modelo visual y el código.
2. Poderosa herramienta de generación de **PDF/HTML** a partir de diagramas UML.
3. Sincronización entre el código fuente y el modelo en tiempo real o bajo demanda.
4. Entorno visual de modelado superior.
5. Soporte para toda la notación UML.
6. Sofisticados y automáticos diagramas de capas.
7. Análisis de textos.
8. Diseño centrado en casos de uso y enfocado al negocio que generan un software de mayor Calidad.
9. Uso de un lenguaje estándar común a todo el equipo de desarrollo que facilita la Comunicación.
10. Capacidades de ingeniería directa (versión profesional) e inversa.
11. Modelo y código que permanece sincronizado en todo el ciclo de desarrollo.
12. Disponibilidad de integrarse en los principales **IDEs**.
13. Disponibilidad en múltiples plataformas.

1.10 Herramientas de Apoyo.

Navegador

El navegador sobre el cual correrá la aplicación es el Mozilla Firefox este permite abrir por defecto las nuevas páginas web en pestañas, cada una de esas pestañas tiene su propio botón de cerrado. Firefox incorpora bloqueo de ventanas emergentes, marcadores dinámicos, soporte para estándares abiertos, y un mecanismo para añadir funcionalidades mediante extensiones. Posee un corrector ortográfico para evitar que se cometan errores de ortografía en las entradas de información que se hagan. Tiene una sugerencia de búsqueda que se va desplazando a medida que se va introduciendo el texto que se desea buscar. Mantiene a salvo a la aplicación de programas espías e impostores usando el poder de una

comunidad de desarrollo que le da soporte. Estas y muchas otras posibilidades brindan este navegador del cual se hará uso para el sistema.

1.11 Conclusiones.

En este capítulo se detallaron las condiciones y problemas que rodean el objeto de estudio; y a través de los conceptos y definiciones planteadas, se determinaron las condiciones específicas que envuelven al mismo. En base a esto se obtuvieron los objetivos generales y específicos para este trabajo, se planteó una solución al problema, se analizaron las características de diferentes herramientas, así como algunas metodologías para el desarrollo de la aplicación Web. Después de este análisis y la fundamentación realizada la técnica de desarrollo web que se utilizará es AJAX y como lenguaje interpretado javascript para el control de las diferentes funciones a realizar en el cliente, como gestor de base de datos se estableció el PostgreSQL y como lenguaje en el servidor usaremos PHP. Se empleará la metodología RUP por tener un mejor monitoreo de la calidad en cada etapa del proceso, como lenguaje de modelado UML y la herramienta que se usará como entorno de creación de diagramas para UML es el Visual Paradigm.

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

2.1 Introducción.

En el campo del software también resulta útil la creación de modelos que organicen y presenten los detalles importantes de problemas reales que se vinculan con el sistema informático a construir.

Como se ha expresado anteriormente el objetivo del Proceso Unificado, es guiar a los desarrolladores de cualquier sistema software en la implementación y distribución eficiente de sistemas que se ajusten a las necesidades de los clientes.

En el presente capítulo, se dará una propuesta de solución de lo que funcionalmente debe realizar el sistema para erradicar los problemas existentes anteriormente mencionados. Se transitará a lo largo de la fase de inicio por cada uno de los flujos de trabajo desde el negocio hasta el sistema. Todo esto se desarrollará por medio de un conjunto de artefactos y modelos resultantes de la aplicación de la metodología de desarrollo de software. Estos modelos deben cumplir una serie de propiedades, entre ellas la de ser coherentes y relacionados. Uno de los modelos útiles previo al desarrollo de un software es el modelo del negocio.

2.2 Propuesta del sistema.

Para desarrollar el Sistema de Registro y Control de Cuadros, el sistema debe ser capaz de permitir registrar la información personal referente a todos los cuadros en nuestro país, sus trayectorias, todos los estudios realizados, las relaciones con familiares, las tenencia de vehículo o armamento, sus actualizaciones políticas, sus padecimientos, además debe permitir mantener actualizado la existencia de reservas por cada cargo, así como brindar una herramienta que permita realizar cualquier tipo de evaluación concerniente a un cuadro. El sistema debe ser capaz de recuperar toda la información referente a un cuadro en formato fijo o configurable, además de permitir la compartimentación de la información por el usuario según el privilegio que se le haya sido asignado.

2.3 Modelado del Negocio.

El modelado del negocio es una técnica para comprender y definir los procesos del negocio, roles y responsabilidades de la organización en los modelos de casos de uso del negocio y de objetos.

Los propósitos que se persiguen al realizarse el modelado del negocio, son:

1. Comprender la estructura y la dinámica de la organización en la cual se va a implantar el sistema.
2. Comprender los problemas actuales de la organización e identificar las mejoras potenciales.
3. Asegurar que los usuarios finales y desarrolladores tengan un entendimiento común de la organización.
4. Derivar los requerimientos del sistema que va a soportar la organización.

Por tanto, la finalidad del modelado del negocio es describir cada proceso del negocio, especificando sus datos, actividades (o tareas), roles (o agentes) y reglas de negocio.

Este proceso está soportado por dos tipos de modelos de UML: modelos de casos de uso y modelo de objetos.

Un modelo de casos de uso del negocio describe los procesos de una empresa en términos de casos de uso y actores del negocio.

Un modelo de objetos del negocio es un modelo que describe cómo colaboran los trabajadores y las entidades del negocio dentro del flujo (realización). Para lograr un mejor entendimiento del modelado del negocio, se presenta una descripción de cuales serán los actores y trabajadores del mismo.

2.4 Actor y Trabajadores del Negocio.

Un actor puede representar a una persona física, otro sistema, un dispositivo, siendo siempre un tercero fuera del sistema que colabora con él. La definición de actores, sirve para definir el contexto externo del sistema, esto es, delimitar los elementos que se encuentran fuera y dentro del mismo, y un actor juega un rol para cada caso de uso en el que colabora.

Tabla 2.1. Descripción de los actores del negocio.

Actor	Descripción
Comisión de Cuadro	Son los encargados de proponer y aprobar los cuadros y su reserva para cada una de las instancias del país.

El trabajador, por otro lado, representa una posición que se puede asignar a una persona o equipo en una organización de desarrollo de software, y para cada trabajador se especifican las responsabilidades y habilidades requeridas, ya que este define el comportamiento y las responsabilidades de un individuo.

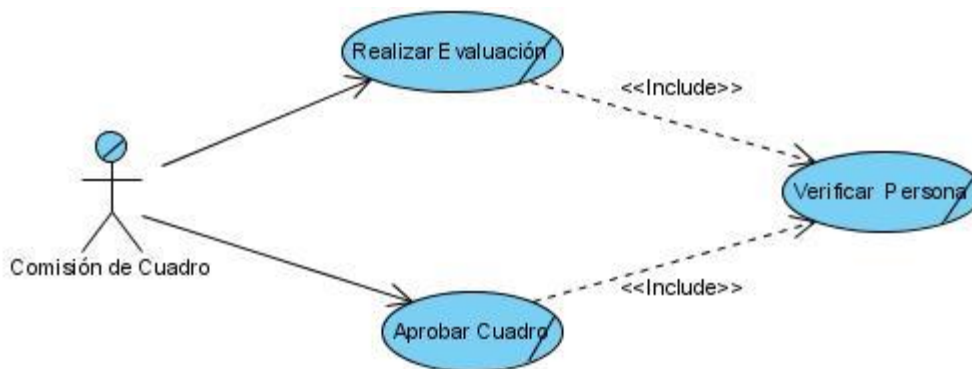
Tabla 2.2 Descripción de los trabajadores del negocio.

Trabajador	Descripción
Especialista	Es el que se encarga de buscar y procesar la información que es solicitada por la comisión de cuadro

2.5 Diagrama de Casos de Uso del Negocio.

Para lograr los propósitos del negocio y obtener una visión general de los diferentes procesos del mismo se construye el diagrama de casos de uso del negocio, donde cada proceso se representa como un caso de uso y solo aparecerán los actores del negocio correspondientes a los roles externos al sistema. Este diagrama permite mostrar los límites y el entorno de la organización.

2.6 Descripciones de los casos de usos del negocio.

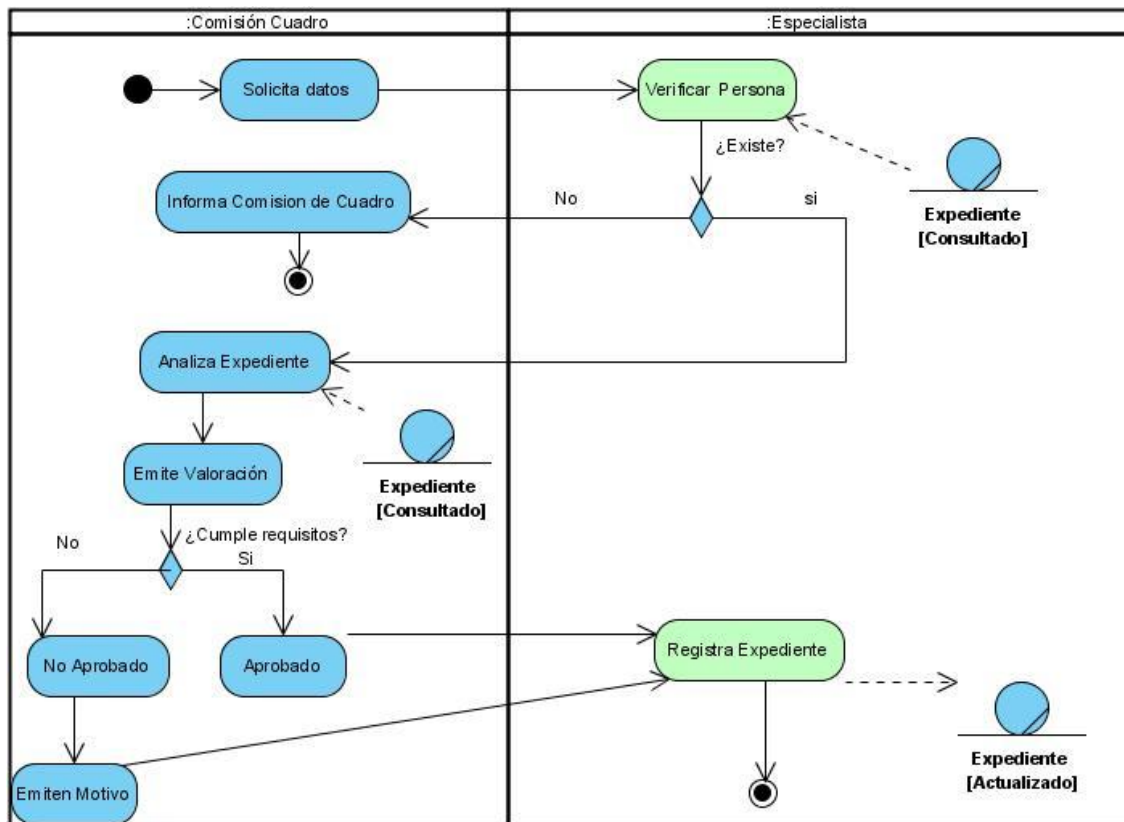


2.6.1 Descripción de Casos de Uso: Aprobar Cuadro

Tabla 2.3 Descripción de Casos de Uso Aprobar Cuadro.

Caso de Uso:	Aprobar Cuadro		
Actores:	Comisión Cuadro (inicia)		
Trabajadores:	Especialista		
Resumen:	El CU inicia cuando la Comisión Cuadro necesita aprobar un cuadro, para ello debe buscar a la persona a la que se le va a hacer el proceso		
Precondiciones:	Se debe elegir al cuadro que se le va a aprobar		
Flujo Normal de Eventos			
Acción del Actor		Respuesta del Negocio	
1.La comisión Cuadro solicita los datos del cuadro que se va a aprobar		2. El especialista Verifica a la persona que se le va a realizar el proceso.	
3. La comisión de cuadro analiza el Expediente de la persona encontrada.			
4. La comisión de cuadro verifica que cumple con los requisitos y es aprobado		5. El especialista registra en el Expediente de la persona que ha pasado a formar parte de los cuadros del país.	
Flujos Alternos			
		2.1 el Especialista de cuadro no encuentra a la persona se le informa a la comisión de cuadro y se termina el CU.	
4.1 La comisión de cuadro no aprueba a la persona se Emiten los motivos.		5. El especialista registra los motivos por los cuales no fue aprobado en el expediente de la persona.	
Pos condiciones			
Mejoras	La aprobación de cuadros se hará de manera informatizada		
Prioridad	Alta		

2.6.1.1 Diagrama de Actividades<Aprobar Cuadro>



2.6.1.2 Modelo de Objetos<Aprobar Cuadro>

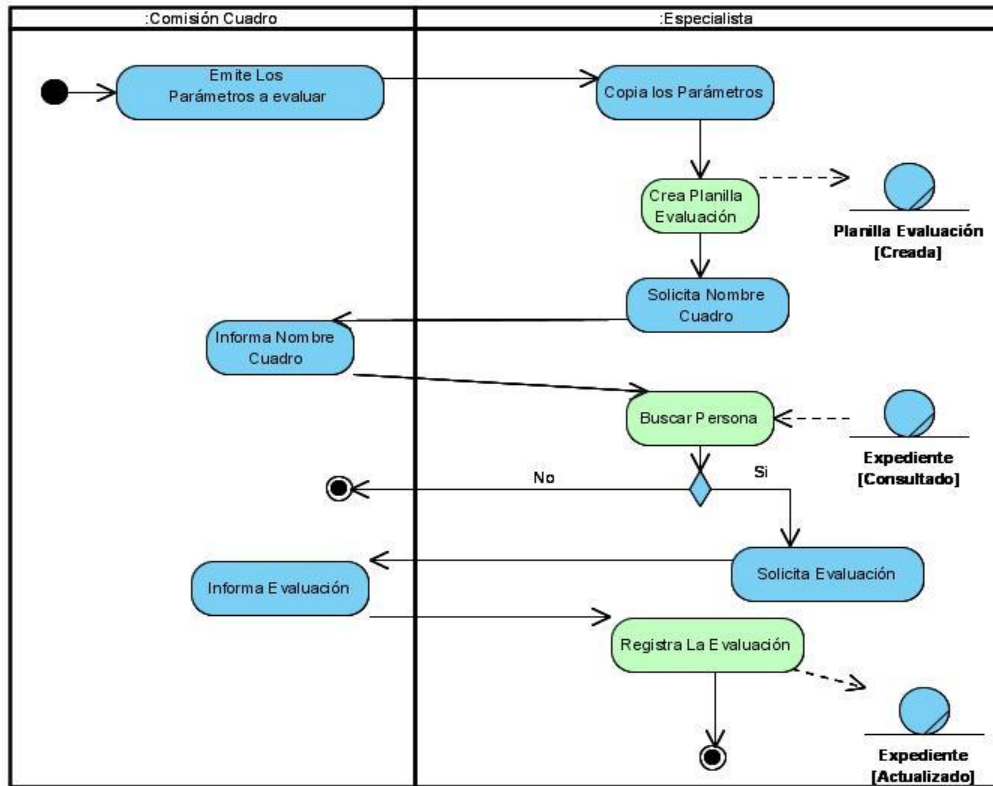


2.6.2 Descripción de Casos de Uso: Realizar Evaluación

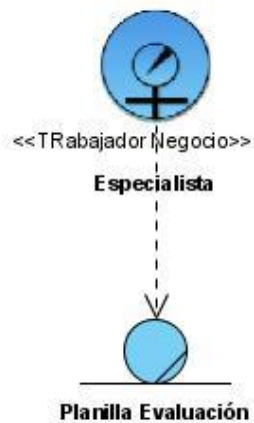
Tabla 2.4 Descripción de Casos de Uso Realizar Evaluación

Caso de Uso:	Realizar Evaluación
Actores:	Comisión de Cuadro
Trabajadores:	Especialista
Resumen:	El CU inicia cuando la Comisión de Cuadros necesita a realizar una evaluación del cuadro, para ello emite los parámetros por lo que será evaluado, y realiza la evaluación del cuadro que se propone.
Precondiciones:	Se debe seleccionar un Cuadro
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Negocio
1. Emite Parámetros a evaluar	2. Copia Los Parámetros
	3. Crea Planilla Evaluación
	4. Solicita Nombre Cuadro
5. Informa Nombre Cuadro	6. Buscar Persona
	7. Solicita Evaluación
8. Informa Evaluación	9. Registra La Evaluación
Flujos Alternos	
	6.1 El Especialista de cuadro no encuentra a la persona se le informa a la comisión de cuadro y se termina el CU.
Poscondiciones	
Mejoras	La Evaluación de Cuadro se hará de manera informatizada
Prioridad	Alta

2.6.2.1 Diagrama de Actividades < Realizar Evaluación >



2.6.2.2 Modelo de Objetos <Realizar Evaluación >

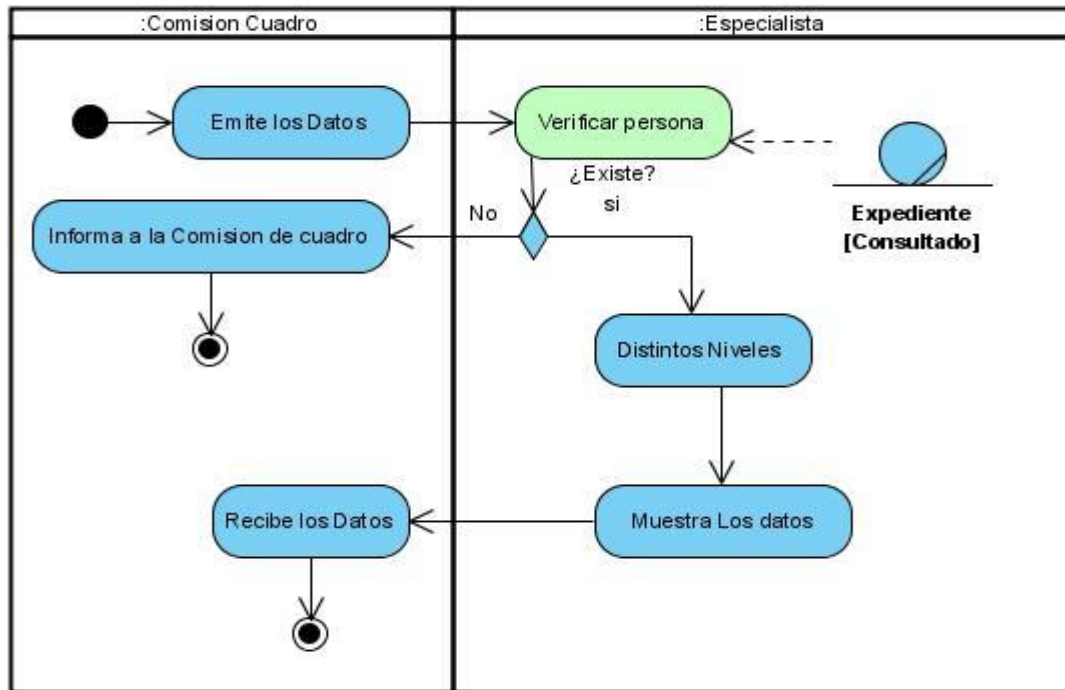


2.6.3 Descripción de Casos de Uso: Verificar Persona

Tabla 2.5 Descripción de Casos de Uso Verificar Persona

Caso de Uso:	verificar Persona	
Actores:	Comisión Cuadro	
Trabajadores:	Especialista	
Resumen:	El CU inicia cuando La Comisión de Cuadro necesita Realizar Alguna Actividad Relacionada con un cuadro para ello el especialista es el encargado de suministrarle los datos de esa persona	
Precondiciones:	Se debe dar el nombre de la persona que se esta buscando	
Flujo Normal de Eventos		
Acción del Actor		Respuesta del Negocio
1. Emite los Datos		2. Verifica a la persona a los distintos niveles.
		3. Muestra Los datos
Flujos Alternos		
		2.1 El Especialista de cuadro no encuentra a la persona se le informa a la comisión de cuadro y se termina el CU.
Poscondiciones		
Mejoras	La Búsqueda de Persona se hará de manera informatizada	
Prioridad	Alta	

2.6.3.1 Diagrama de Actividades <Verificar Persona>



2.6.3.2 Modelo de Objetos <Verificar Persona>



2.7 Requerimientos Funcionales

Los requerimientos funcionales son capacidades o condiciones que el sistema debe cumplir. En los diagramas de actividades se puso de color verde las actividades que se van a automatizar, estas son el punto de partida para desarrollar lo que el sistema debe de hacer.

- R1 Gestionar Cuadro.
 - R1.1 Agregar Persona.
 - R1.2 Eliminar Cuadro.
 - R1.3 Eliminar Cuadro Físicamente.
- R2. Actualizar Datos Personales.
- R3 Actualizar Datos Políticos.
 - R3.1 Meritos y Desméritos.
 - R3.1.1 Condecoraciones
 - R3.1.2 Sanciones.
 - R3.2 Militancia.
- R4 Buscar Cuadro.
- R5 Actualizar Trayectoria.
 - R5.1 Política.
 - R5.2 Estancia en el Exterior.
 - R5.3 Laboral.
 - R5.4 Militar.
 - R5.5 Residencial.
 - R5.6 Reserva Cargo.
- R6 Actualizar Vehículo_Armamento.
 - R6.1 Vehículo.
 - R6.2 Armamento.
- R7 Actualizar Familiares.
 - R7.1 Allegados.
 - R7.2 Otros Con los que convive.
 - R7.3 Residentes en el Exterior.

R7.4 Vínculos con Extranjeros.

R7.5 Que han Viajado al Exterior.

R7.6 Sancionados Jurídicamente.

R7.7 Con Ingresos Monetarios Extra.

-R8 Actualizar Idioma.

-R9 Actualizar Salud.

R9.1 Estado de Salud.

R9.2 Padecimientos.

-R10 Realizar Reporte.

-R11 Actualizar Estudios.

R11.1 Enseñanza Media.

R11.2 Enseñanza Políticas.

R11.3 Militares.

R11.4 Especiales.

-R12 Realizar Evaluación.

-R13 Gestionar Planilla Evaluativa

R13.1 Insertar Planilla Evaluativa

R13.2 Modificar Planilla Evaluativa

R13.3 Eliminar Planilla Evaluativa

-R14 Gestionar Nomencladores

R14.1 Insertar Nomenclador.

R14.2 Modificar Nomenclador.

R14.3 Eliminar Nomenclador.

-R15 Gestionar Nomenclador de Estructura

R15.1 Insertar Nomenclador de Estructura

R15.2 Modificar Nomenclador de Estructura

R15.3 Eliminar Nomenclador de Estructura

2.8 Requerimientos No Funcionales

Los requerimientos no funcionales son propiedades o cualidades que el producto debe tener. Estas propiedades como las características que hacen al producto atractivo, usable, rápido o confiable en otras palabras los requerimientos no funcionales son fundamentales en el éxito del producto.

Software

Para el cliente:

Navegador Mozilla Firefox.

Sistema operativo Linux.

Para el servidor:

Servidor apache con módulo PHP5 disponible.

Sistema operativo Linux en cualquiera de sus distribuciones.

Un servidor de base de datos PostgreSQL.

Hardware

Para el cliente:

Requerimientos mínimos: Procesador Pentium II a 133Mhz con 128 Mb de memoria RAM. Tarjeta de red.

Para el servidor:

Requerimientos mínimos: Procesador Pentium III a 1GHz de velocidad de procesamiento y 1Gb de memoria RAM. Al menos 20Gb de espacio libre en disco duro.

Apariencia o interfaz externa

Estará diseñado para resolución de 1024x768.

Emplear colores e imágenes que estén relacionadas con el negocio donde se implantará el sistema.

El sistema debe contar con una interfaz fácil, amigable, sencilla, permitiendo que los usuarios finales del mismo sean capaces de interactuar con este, aún teniendo conocimientos básicos.

Usabilidad

El sistema podrá ser utilizado por personas que tengan un conocimiento mínimo en el manejo de las computadoras. El software tendrá siempre visible la opción de Ayuda, lo que posibilitará una mejor explotación por parte de los usuarios de sus funcionalidades. En los controles tipo lista que posean más de 15 posibles selecciones se empleara una búsqueda por teclado.

Rendimiento

Los tiempos de respuesta y velocidad de procesamiento de la información serán rápidos, no mayores de 5 segundos para las actualizaciones y 20 para las recuperaciones.

El diseño y la implementación

Utilizar los estándares establecidos de codificación, diseño, entre otros. Emplear como servidores web y de bases de datos Apache y PostgreSQL respectivamente. Utilizar como lenguaje del lado del servidor al PHP y del lado del cliente el JavaScript, HTML, AJAX y XML.

Soporte

Se necesita un servidor apache para el trabajo con PHP que posea un gestor de bases de datos que soporte grandes volúmenes de datos. Debe elaborarse un paquete de instalación.

Portabilidad

El sistema será multiplataforma (Linux o Windows).

Seguridad

El usuario debe autenticarse antes de entrar al sistema. Garantizar el acceso controlado a la información.

Legales

El sistema debe ajustarse y regirse por las leyes estipuladas para dar cumplimiento a los procesos que se automatizarán.

I. Decreto Ley 196. Capítulo II/Artículo 10:

Para la correcta aplicación de la Política respecto al trabajo con los cuadros se establecen, de modo particular e interrelacionadas las principales acciones que integran el sistema y que son las siguientes:

- ✓ Evaluación
- ✓ Información

Políticos-culturales

El sistema que se automatizara solo podrá ser utilizado en nuestro país y por las entidades, organismos e instituciones autorizadas por Nuestro País. El producto no debe contener palabras en idiomas extranjeros.

Ayuda y documentación en línea

Se propone que el sistema cuente con una ayuda general en la página principal, que guiará al usuario de cómo trabajar en el sistema, también estará disponible en cada una de las interfaces, de esta forma los usuarios tendrán conocimiento de las funcionales del mismo y hacer un mejor uso de las funcionalidades del sistema.

2.9 Reglas del negocio a considerar

- La información del cuadro almacenada debe ser real.
- Debe existir el cuadro que se va a insertar en el sistema.
- La información debe ser accedida por personas autorizadas.

2.10 Sistema

Los requerimientos funcionales unidos por funcionalidades conforman los casos de usos del sistema, y los actores del negocio interactúan con estos.

2.10.1 Actores del Sistema

Tabla 2.6 Descripción de los actores del sistema.

Actor	Descripción
Especialista	Es el actor encargado de realizar la mayoría de las operaciones, es el responsable de Actualizar los Datos Personales y políticos de los cuadros, Registrar las Evaluaciones

2.10.2 Listado de casos de uso

Tabla 2.9 CU del sistema Gestionar Cuadro

CU-1	Gestionar Cuadro
Actor	Especialista
Descripción	El CU inicia cuando el Especialista procede a realizar una operación (Agregar cuadro, Eliminar Cuadro, Eliminar cuadro físicamente), el sistema muestra la Interfaz correspondiente a la acción seleccionada, el actor actualiza los datos. El CU finaliza cuando queda actualizado la acción a tomar
referencia	R 1. R 1.2 , R 1.3

Tabla 2.10 CU del sistema Actualizar Datos Personales

CU-2	Actualizar Datos Personales
Actor	Especialista
Descripción	El CU se inicia cuando el Especialista procede actualizar los datos personales del Cuadro, el sistema muestra la interfaz correspondiente, el actor introduce los datos solicitados y actualiza aquellos específicos. El CU finaliza cuando queda actualizado actualizar los datos personales.
Referencia	R 2

Tabla 2.11 CU del sistema Actualizar Datos Políticos

CU-3	Datos Políticos
Actor	Especialista
Descripción	El CU inicia cuando el especialista decide Datos Políticos (Meritos y Desméritos, Militancia) para esto el sistema muestra la interfaz correspondiente a la opción seleccionada. El CU termina cuando el especialista introduce los datos y es actualizada la opción solicitada.
referencia	R 3 R 3.1, R 3.2

Tabla 2.12 CU del sistema Buscar Persona

CU-4	Buscar Persona
Actor	Especialista
Descripción	El CU se inicia cuando el Especialista procede a buscar a un Cuadro, el sistema muestra la interfaz correspondiente, el actor introduce los datos necesarios para la búsqueda. El CU finaliza cuando se encuentra el Cuadro.
Referencia	R 4

Tabla 2.13 CU del sistema Actualizar Trayectoria

CU-5	Actualizar Trayectoria
Actor	Especialista
Descripción	El CU inicia cuando el especialista decide Actualizar Trayectoria del cuadro (Política, Estancia en el Exterior, Laboral, Militar, Residencia, Asignar a la Reserva) para ello el sistema muestra la interfaz correspondiente a la opción seleccionada. El CU termina cuando el especialista introduce los datos y es actualizada la opción solicitada.
referencia	R 5.1, R 5.2 , R 5.3 , R 5.4 , R 5.5, R 5.6

Tabla 2.14 CU del sistema Actualizar Vehículo_Armamento

CU-6	Actualizar Vehículo_Armamento
Actor	Especialista
Descripción	El CU se inicia cuando el Especialista procede actualizar el vehículo y armamento del Cuadro, el sistema muestra la interfaz correspondiente, el actor introduce los datos solicitados y actualiza aquellos específicos. El CU finaliza cuando queda actualizado el vehículo y armamento.
Referencia	R 6, R 6.1,R 6.2

Tabla 2.15 CU del sistema Actualizar Familiares

CU-7	Actualizar Familiares
Actor	Especialista
Descripción	El CU inicia cuando el especialista decide actualizar datos del familiar (Allegados, Otros Con los que convive, Residentes en el Exterior, Vínculos con Extranjeros, Que han Viajado al Exterior Sancionados Jurídicamente, Con Ingresos Monetarios Extra) para ello el sistema muestra la interfaz correspondiente a la opción seleccionada. El CU termina cuando el especialista introduce los datos y es actualizada la opción solicitada.
Referencia	R 7, R 7.1, R 7.2, R 7.3 R 7.4, R 7.5

Tabla 2.16 CU del sistema Actualizar Idioma

CU-8	Actualizar Idioma
Actor	Especialista
Descripción	El CU se inicia cuando el Especialista procede actualizar los idiomas del Cuadro, el sistema muestra la interfaz correspondiente, el actor introduce los datos solicitados y actualiza aquellos específicos. El CU finaliza cuando queda actualizado actualizar los idiomas.
Referencia	R 8

Tabla 2.17 CU del sistema Actualizar Salud

CU-9	Actualizar Salud
Actor	Especialista
Descripción	El CU se inicia cuando el Especialista procede actualizar salud (Estado de salud, padecimientos). El sistema muestra la interfaz correspondiente, el actor introduce los datos solicitados y actualiza aquellos específicos. El CU finaliza cuando queda actualizado actualizar los padecimientos.
Referencia	R 9

Tabla 2.18 CU del sistema Realizar Reporte

CU-10	Realizar Reporte
Actor	Especialista
Descripción	El CU inicia cuando el Especialista procede a realizar un reporte. El sistema muestra la interfaz correspondiente, elige los datos por lo que se hará el reporte. El CU termina cuando esta conformado el reporte
referencia	R 10

Tabla 2.19 CU del sistema Actualizar Estudios

CU-11	Actualizar Estudios
Actor	Especialista
Descripción	El CU inicia cuando el especialista decide actualizar Estudios (Enseñanza Media, Enseñanza Políticas, Militares, Especiales) para ello el sistema muestra la interfaz correspondiente a la opción seleccionada. El CU termina cuando el especialista introduce los datos y es actualizada la opción solicitada.
referencia	R 11, R 11.1 ,R 11.2 ,R 11.3, R 11.4

Tabla 2.20 CU del sistema Realizar Evaluación

CU-12	Realizar Evaluación
Actor	Especialista
Descripción	El CU se inicia cuando el Especialista procede a realizar la evaluación del Cuadro, el sistema muestra la interfaz correspondiente, el actor introduce los datos. El CU finaliza cuando se termina la evaluación.
Referencia	R 12

Tabla 2.21 CU del sistema Gestionar Planilla Evaluativa

CU-13	Gestionar Planilla Evaluativa
Actor	Especialista
Descripción	El CU se inicia cuando el Especialista procede a Gestionar Planilla (Insertar, Eliminar, Modificar), el sistema muestra la interfaz correspondiente, el actor introduce los datos. El CU finaliza cuando es actualizada la información.
Referencia	R 13

Tabla 2.22 CU del sistema Gestionar Nomencladores

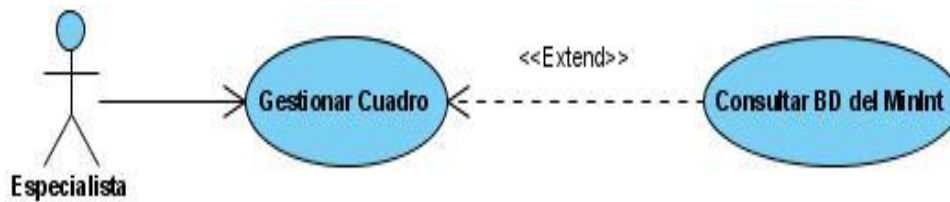
CU-14	Gestionar Nomencladores
Actor	Especialista
Descripción	El CU se inicia cuando el Especialista procede a Gestionar Nomencladores (Insertar, Eliminar, Modificar), el sistema muestra la interfaz correspondiente, el actor introduce los datos. El CU finaliza cuando es actualizada la información.
Referencia	R 14

Tabla 2.23 CU del sistema Gestionar Nomenclador de Estructura

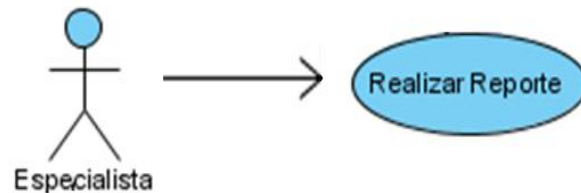
CU-15	Gestionar Nomenclador de Estructura
Actor	Especialista
Descripción	El CU se inicia cuando el Especialista procede a Gestionar Nomenclador de Estructura (Insertar, Eliminar, Modificar), el sistema muestra la interfaz correspondiente, el actor introduce los datos. El CU finaliza cuando es actualizada la información.
Referencia	R 15

2.10.3 Diagrama de Casos de Uso del Sistema

3.3.1 Diagrama de CU del Sistema Gestionar Cuadro.



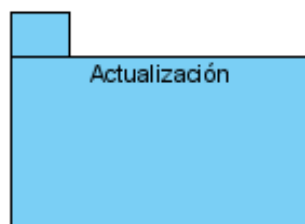
3.3.2 Diagrama de CU del Sistema Realizar Reporte.



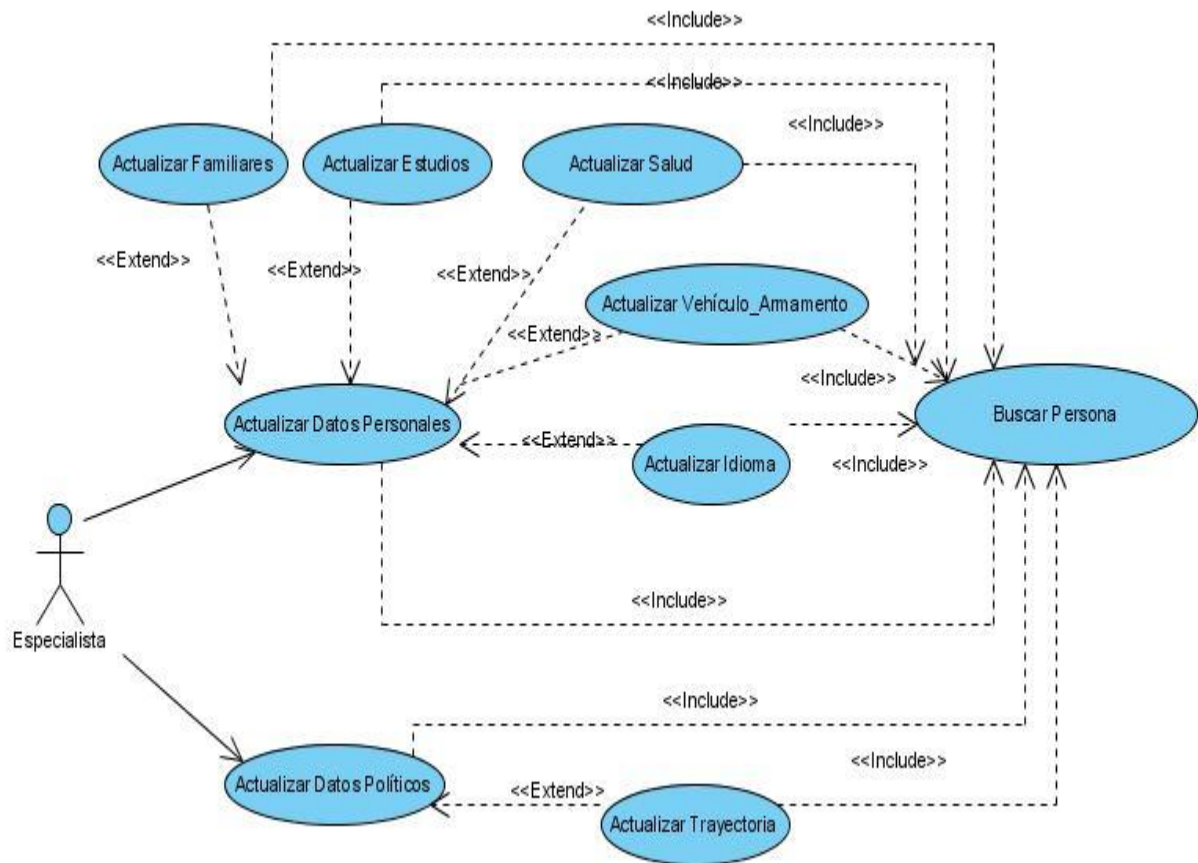
2.10.4 Paquetes y sus relaciones.

Para la modelación de los casos de uso del sistema se decidió dividirlos en distintos paquetes de acuerdo a sus funcionalidades.

2.10.4.1 Paquete de Actualización.

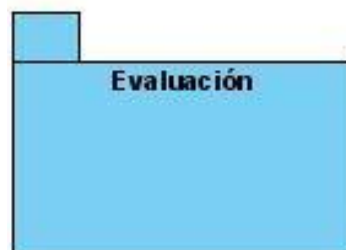


Los CU representados a continuación reflejan el Proceso de Actualización

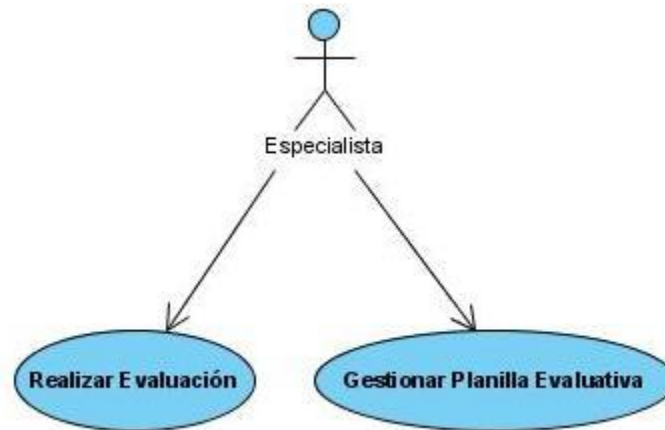


3.3.3 Diagrama de CU del Sistema del paquete Actualizar.

2.10.4.2 Paquete de Evaluación.

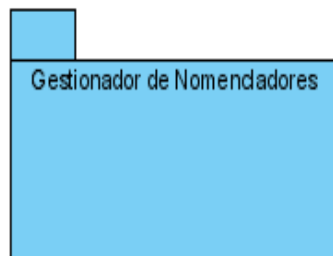


Los CU representados a continuación reflejan el Proceso de Evaluación.

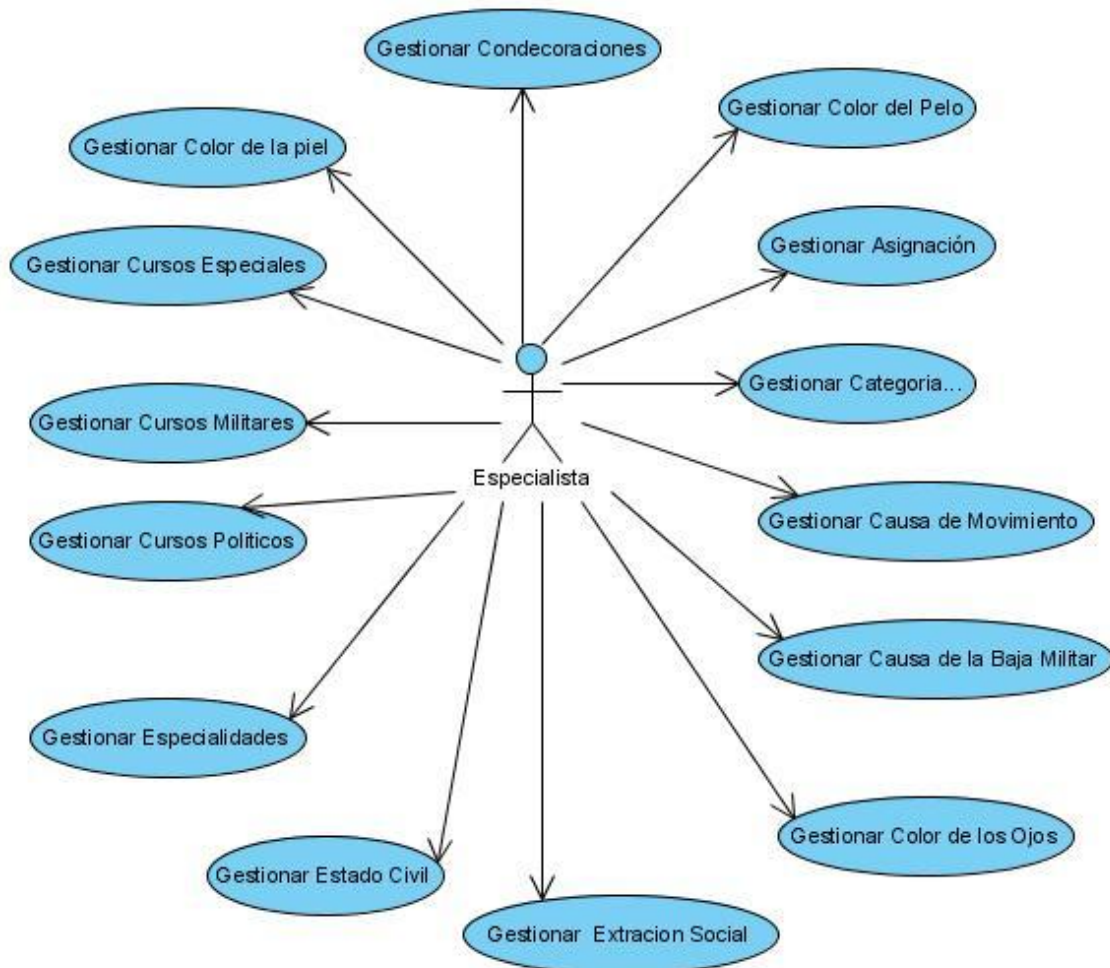


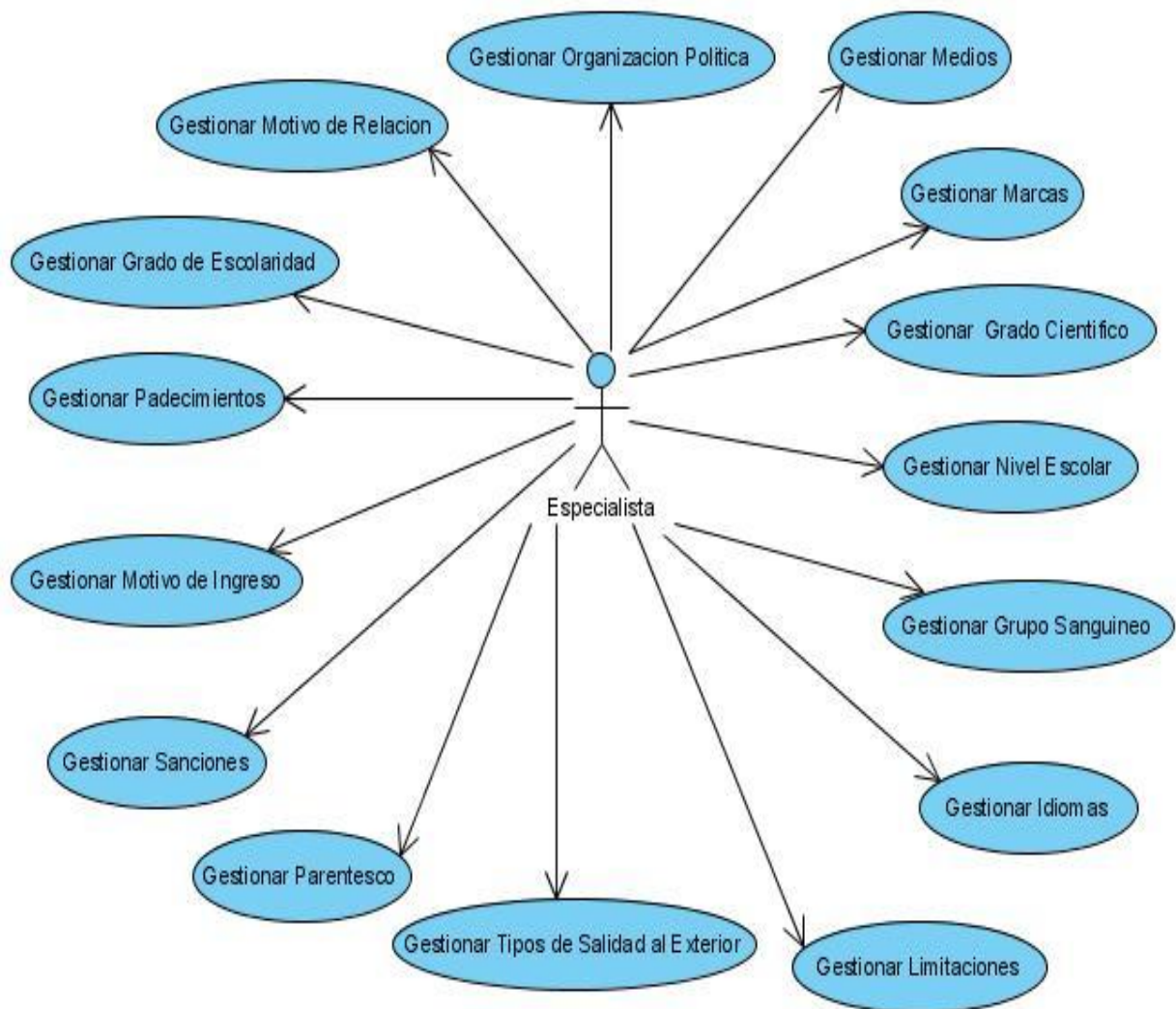
3.3.4 Diagrama de CU del Sistema del paquete Evaluación.

2.10.4.3 Paquete de Gestionador de Nomencladores.



Los CU representados a continuación reflejan el Proceso de Gestionador de Nomencladores.





3.3.5 Diagrama de CU del Sistema del paquete Gestionador de Nomencladores.

2.10.5 Descripción Extendida de los CU del Sistema

Tabla 2.4 Descripción del caso de uso Gestionar Cuadro.

Caso de Uso	Gestionar Cuadro
Actores	Especialista(inicia)
Propósito	El Especialista autorizado Agrega Persona, Elimina Cuadro o Elimina cuadro físicamente.
Resumen:	El CU inicia cuando el Especialista procede a realizar una operación (Agregar cuadro, Eliminar Cuadro, Eliminar cuadro físicamente), el sistema muestra la Interfaz correspondiente a la acción seleccionada, el actor actualiza los datos, el CU finaliza cuando queda actualizada la acción a tomar.
Responsabilidades	R 1. R1.1 R1.2 R1.3
CU asociados	CU asociado Buscar Persona relación incluye
Precondiciones	El actor debe estar autenticado. Debe Mostrarse la Interfaz de acuerdo a la opción seleccionada
Requisitos especiales	El CU Buscar Persona dispone de 20 segundos para buscar a la persona como máximo
Postcondiciones	Los Datos son insertados correctamente
Prioridad	Critica

Tabla 2.4 Descripción del caso de uso Buscar Persona.

Caso de uso:	Buscar Persona
Actores:	Especialista
Propósito:	Buscar una persona.
Resumen:	El caso de uso se inicia cuando el actor selecciona "Buscar Persona" producto a ejecutar cualquier caso de uso sin haber seleccionado a la persona, el sistema muestra la interfaz correspondiente, el actor introduce los criterios de búsqueda. El caso de uso finaliza cuando se haya encontrado a la persona.
Responsabilidades:	R 4
CU asociados:	
Precondiciones :	El actor debe estar autenticado.
Postcondiciones	Se encuentra la persona.
Prioridad	Critica

Tabla 2.5 Descripción del caso de uso Actualizar Datos Políticos.

Caso de Uso	Actualizar Datos Políticos
Actores	Especialista
Propósito	El Especialista autorizado Se encarga de Agregar Meritos y Desméritos y Militancia del Cuadro seleccionado
Resumen	El CU inicia cuando el Especialista procede a Actualizar Datos Políticos (Meritos y Desméritos, Militancia), el sistema muestra la Interfaz correspondiente a la acción seleccionada, el actor actualiza los datos, el CU finaliza cuando queda actualizada la acción a tomar.
Responsabilidades	R3, R3.1,R3.2
CU asociados	CU asociado Buscar Persona relación incluye.
Precondiciones	El actor debe estar autenticado. Debe Mostrarse la Interfaz de acuerdo a la opción seleccionada
Poscondiciones	Los datos fueron insertados correctamente
Prioridad	Critica

Tabla 2.6 Descripción del caso de uso Actualizar Trayectoria.

Caso de Uso	Actualizar Trayectoria.
Actores	Especialista (inicia)
Propósito	El Especialista autorizado Actualiza la Trayectoria los datos del cuadro (Política, Estancia en el Exterior, Laboral, Militar, Residencial, Reserva Cargo).
Resumen	El CU inicia cuando el Especialista procede a realizar una operación (Política, Estancia en el Exterior, Laboral, Militar, Residencial, Reserva Cargo).
Responsabilidades	R5, R 5., R 5.2, R 5.3, R 5.4, R 5.5, R 5.6.
CU asociados	Buscar Persona
Precondiciones	Debe Mostrarse la Interfaz de acuerdo a la opción seleccionada
Postcondiciones	Los datos son insertados correctamente
Prioridad	Critica

Tabla 2.7 Descripción del caso de uso Actualizar Familiares.

Caso de Uso	Actualizar Familiares.
Actores	Especialista(inicia)
Propósito	El Especialista autorizado Actualiza los datos del cuadro de acuerdo a sus familiares (Allegados, Otros con los que convive, Residentes en el Exterior, Vínculos con Extranjeros, Que han Viajado al Exterior, Sancionados Jurídicamente, Con Ingresos Monetarios Extra).
Resumen	El CU inicia cuando el Especialista procede a realizar una operación (Allegados, Otros Con los que convive, Residentes en el Exterior, Vínculos con Extranjeros, Que han Viajado al Exterior, Sancionados Jurídicamente, Con Ingresos Monetarios Extra) el sistema muestra la Interfaz correspondiente a la acción seleccionada, el actor actualiza los datos, el CU finaliza cuando queda actualizada la acción a tomar.
Responsabilidades	R7, R7.1, R7.2, R7.3, R7.4, R7.5, R7.6,R7.7
CU asociados	Buscar Persona Relación de Incluye
Precondiciones	El actor debe estar autenticado.
Postcondiciones	Los datos son insertados correctamente
Prioridad	Crítica

Tabla 2.8 Descripción del caso de uso Actualizar Datos Personales.

Caso de uso:	Actualizar Datos Personales
Actores:	Especialista(inicia)
Propósito:	Actualizar los datos personales de un cuadro.
Resumen:	El caso de uso se inicia cuando el actor selecciona Datos Personales en el menú principal, el sistema muestra la interfaz correspondiente, el actor introduce los datos del cuadro. El caso de uso finaliza cuando se ha actualizado los datos personales del cuadro.
Responsabilidades:	R 2
CU asociados:	
Precondiciones :	El actor debe estar autenticado.
Postcondiciones	Se actualiza los datos personales.
Prioridad	Secundaria

Tabla 2.9 Descripción del caso de uso Actualizar Vehículo_Armamento.

Caso de Uso	Actualizar Vehículo_Armamento
Actores	Especialista(inicia)
Propósito	El Especialista autorizado Actualiza los datos del cuadro de acuerdo a sus Vehículos y armamentos.
Resumen	El CU inicia cuando el Especialista procede a realizar una operación (Vehículos, armamentos), el sistema muestra la Interfaz correspondiente a la acción seleccionada, el actor actualiza los datos, el CU finaliza cuando queda actualizada la acción a tomar.
Responsabilidades	R6,R6.1,R6.2
CU asociados	Buscar Persona Relación de Incluye
Precondiciones	El actor debe estar autenticado. Debe Mostrarse la Interfaz de acuerdo a la opción seleccionada
Postcondiciones	Los datos son insertados correctamente
Prioridad	Secundaria

Tabla 2.10 Descripción del caso de uso Actualizar Idioma.

Caso de uso:	Actualizar Idioma
Actores:	Especialista(inicia)
Propósito:	Actualizar la información sobre los idiomas que domina el cuadro.
Resumen:	El caso de uso se inicia cuando el actor selecciona "Idioma" en el menú principal, el sistema muestra la interfaz correspondiente, el actor actualiza los idiomas que domina el cuadro. El caso de uso finaliza cuando se ha actualizado la información sobre los idiomas que domina el cuadro.
Responsabilidades:	R 8
CU asociados:	
Precondiciones :	El actor debe estar autenticado Especialista.
Postcondiciones	Se actualiza los idiomas.
Prioridad	Secundaria

Tabla 2.10 Descripción del caso de uso Actualizar Salud.

Caso de Uso	Actualizar Salud.
Actores	Especialista(inicia)
Propósito	El Especialista autorizado Actualiza el estado de salud del cuadro de acuerdo a sus Vehículos y Armamentos.
Resumen	El CU inicia cuando el Especialista procede a realizar una operación (Estado de Salud, Padecimientos), el sistema muestra la Interfaz correspondiente a la acción seleccionada, el actor actualiza los datos, el CU finaliza cuando queda actualizada la acción a tomar.
Responsabilidades	R9,R9.1,R9.2
CU asociados	Buscar Persona Relación de Incluye
Precondiciones	El actor debe estar autenticado. Debe Mostrarse la Interfaz de acuerdo a la opción seleccionada
Postcondiciones	Los datos son insertados correctamente
Prioridad	Secundaria

Tabla 2.11 Descripción del caso de uso Actualizar Gestionar Planilla Evaluativa.

Caso de Uso	Gestionar Planilla Evaluativa
Actores	Especialista(inicia)
Propósito	El Especialista autorizado confecciona la planilla con la que decide evaluar al cuadro.
Resumen	El CU inicia cuando el Especialista procede a confeccionar la planilla evaluativa, el sistema muestra la Interfaz correspondiente a la acción seleccionada, el actor actualiza los datos, el CU finaliza cuando queda actualizada la acción a tomar.
Responsabilidades	R13, R13.1, R13.2, R13.3
Precondiciones	El actor debe estar autenticado. Debe Mostrarse la Interfaz de acuerdo a la opción seleccionada
Postcondiciones	Los datos son insertados correctamente
Prioridad	Critica

Tabla 2.12 Descripción del caso de uso Realizar Evaluación.

Caso de Uso	Realizar Evaluación
Actores	Especialista(inicia)
Propósito	El Especialista autorizado Realiza una Evaluación al cuadro
Resumen	El CU inicia cuando el Especialista procede a realizar la evaluación del cuadro, el sistema muestra la Interfaz correspondiente a la acción seleccionada, el actor actualiza los datos, el CU finaliza cuando queda actualizada la acción a tomar.
Responsabilidades	R12
CU asociados	Buscar Persona Relación de Include
Precondiciones	El actor debe estar autenticado. Debe Mostrarse la Interfaz de acuerdo a la opción seleccionada
Postcondiciones	Los datos son insertados correctamente
Prioridad	Critica

Tabla 2.13 Descripción del caso de uso Gestionar Nomencladores.

Caso de Uso	Gestionar Nomencladores
Actores	Especialista(inicia)
Propósito	El Especialista autorizado Gestiona algún nomenclador
Resumen	El CU inicia cuando el Especialista procede a realizar una operación (insertar, modificar, eliminar algún nomenclador), el sistema muestra la Interfaz correspondiente a la acción seleccionada, el actor actualiza los datos, el CU finaliza cuando queda actualizada la acción a tomar.
Responsabilidades	R14, R14.1, R14.2, R14.3
Precondiciones	El actor debe estar autenticado
Postcondiciones	Los datos son insertados correctamente
Prioridad	Critica

Tabla 2.14 Descripción del caso de uso Realizar Reporte.

Caso de Uso	Realizar Reporte
Actores	Especialista(inicia)
Propósito	El Especialista autorizado Busca los datos de la persona de acuerdo a criterios de selección
Resumen	El CU inicia cuando el Especialista procede a realizar una recuperación, el sistema muestra la Interfaz correspondiente, el actor selecciona los parámetros que desea en el reporte, el CU finaliza cuando queda actualizada la acción a tomar.
Responsabilidades	R10
CU asociados	Buscar Persona relación de (include)
Precondiciones	El actor debe estar autenticado. Debe Mostrarse la Interfaz de acuerdo a la opción seleccionada
Postcondiciones	Los datos son imprimidos correctamente
Prioridad	Critica

2.11 Conclusiones

Con el flujo de trabajo modelamiento del negocio y requerimientos, se puede comprender mejor como se desarrollará el sistema. Siendo los requerimientos la base para el desarrollo de los modelos de análisis y diseño. En este capítulo se mostraron los artefactos fundamentales relacionados con estos flujos de trabajos, los cuales consisten en: reglas del negocio, descripción de los actores y de los trabajadores del negocio, diagrama de casos de uso del negocio, descripción de los casos de uso del negocio, diagrama de actividades, el modelo de objeto, modelo de casos de uso del sistema, descripción de los actores del sistema y prototipo de interfaz usuario. Siendo los principales artefactos del negocio y requerimientos: El modelo de caso de uso del negocio y el modelo de caso de uso sistema respectivamente.

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

3.1 Introducción

En el presente capítulo se modelaran varios artefactos que tienen lugar durante el flujo de trabajo análisis y diseño, teniendo como objetivo traducir los requisitos definidos anteriormente, a una especificación que describe cómo implementar el sistema.

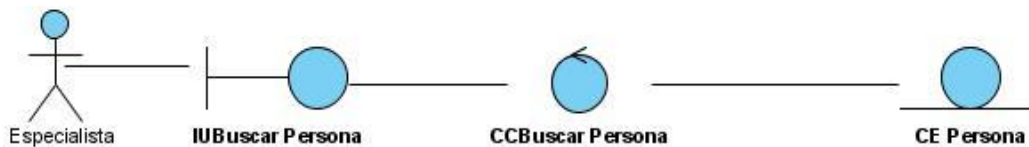
3.2 Análisis

En el análisis se realiza un bosquejo del diseño del sistema, incluyendo su arquitectura. Puede no estar mantenido durante todo el ciclo de vida del software.

3.2.1 Diagrama de clases del análisis

El diagrama de clases del análisis posee tres estereotipos conceptuales sobre las clases: control, entidad y interfaz. A continuación se presentarán los diagramas de clases del análisis que se desarrollaron.

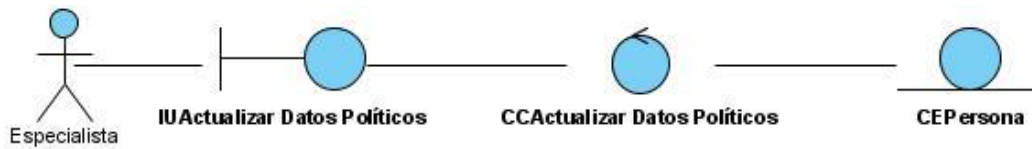
3.2.1.1 Caso de Uso Buscar Persona.



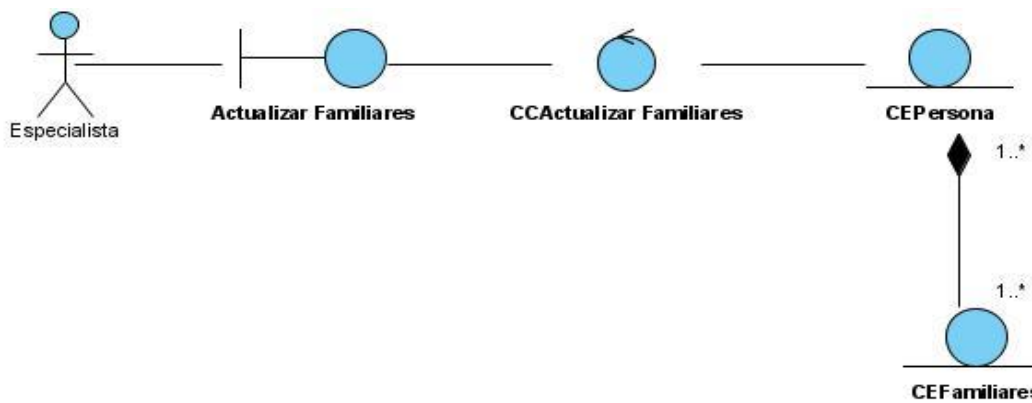
3.2.1.2 Caso de Uso Actualizar Datos Personales.



3.2.1.3 Caso de Uso Actualizar Datos Políticos.



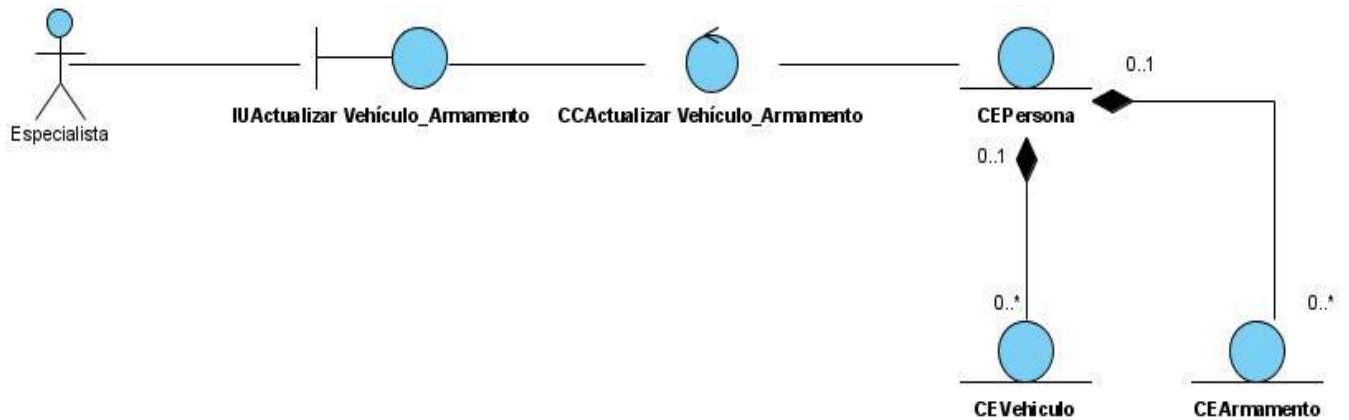
3.2.1.4 Caso de Uso Actualizar Familiares.



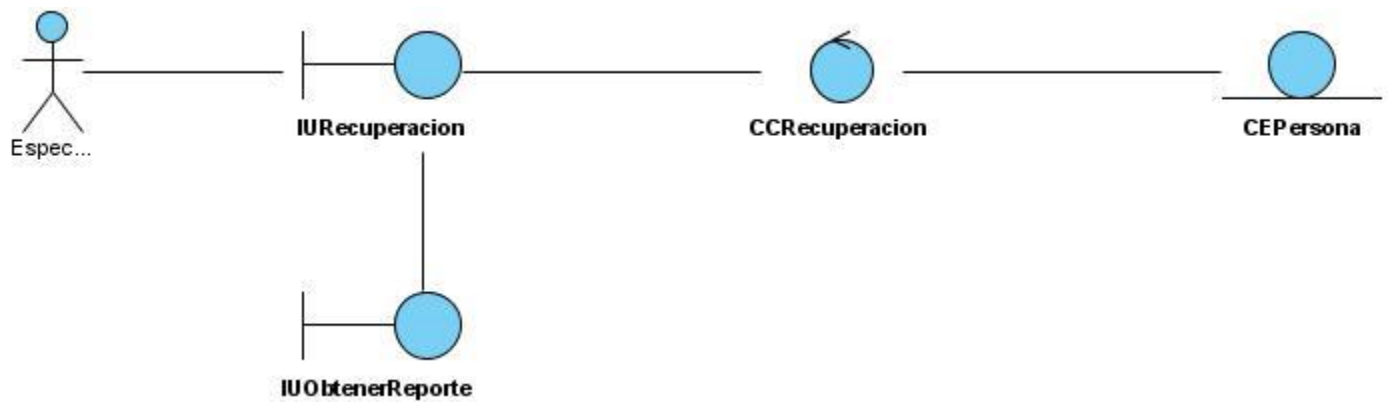
3.2.1.5 Caso de Uso Actualizar Trayectoria.



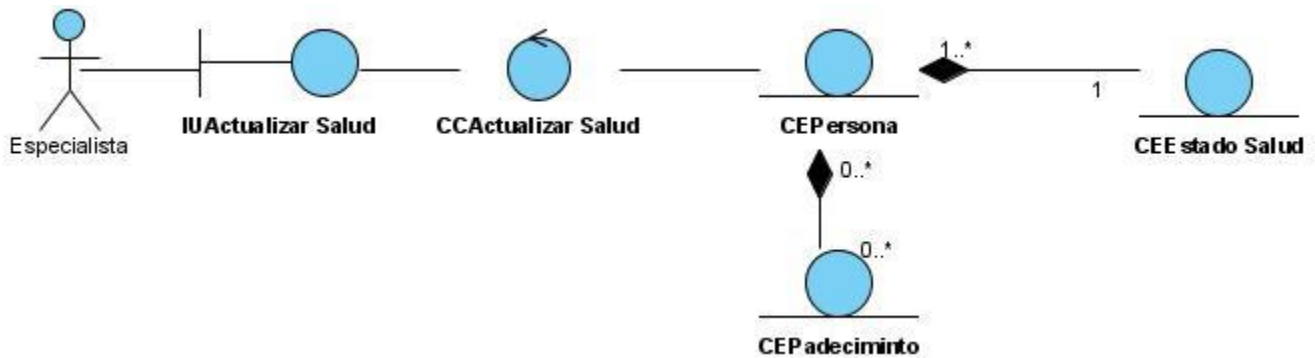
3.2.1.6 Actualizar Vehículo_Armamento.



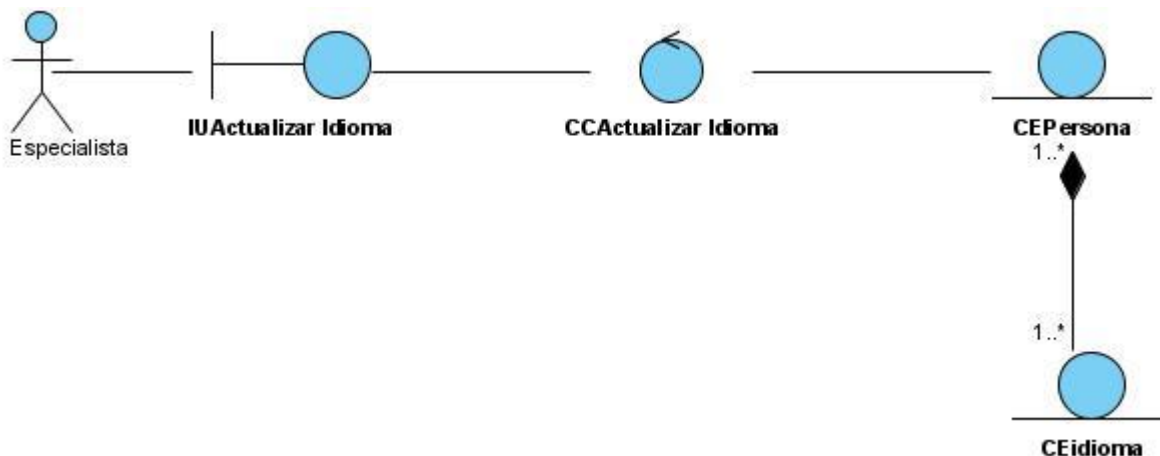
3.2.1.7 Caso de Uso Realizar Reporte.



3.2.1.8 Caso de Uso Actualizar Salud

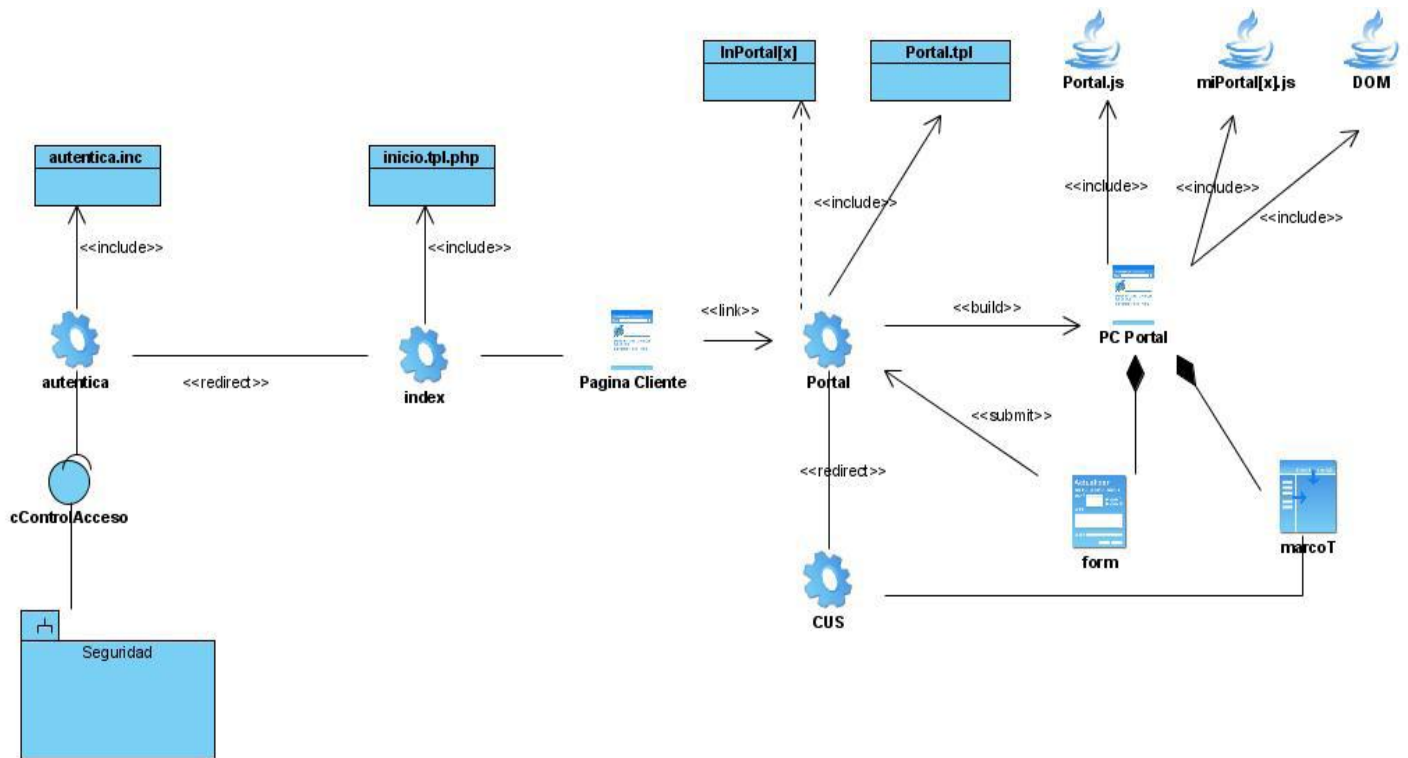


3.2.1.9 Caso de Uso Actualizar Idioma



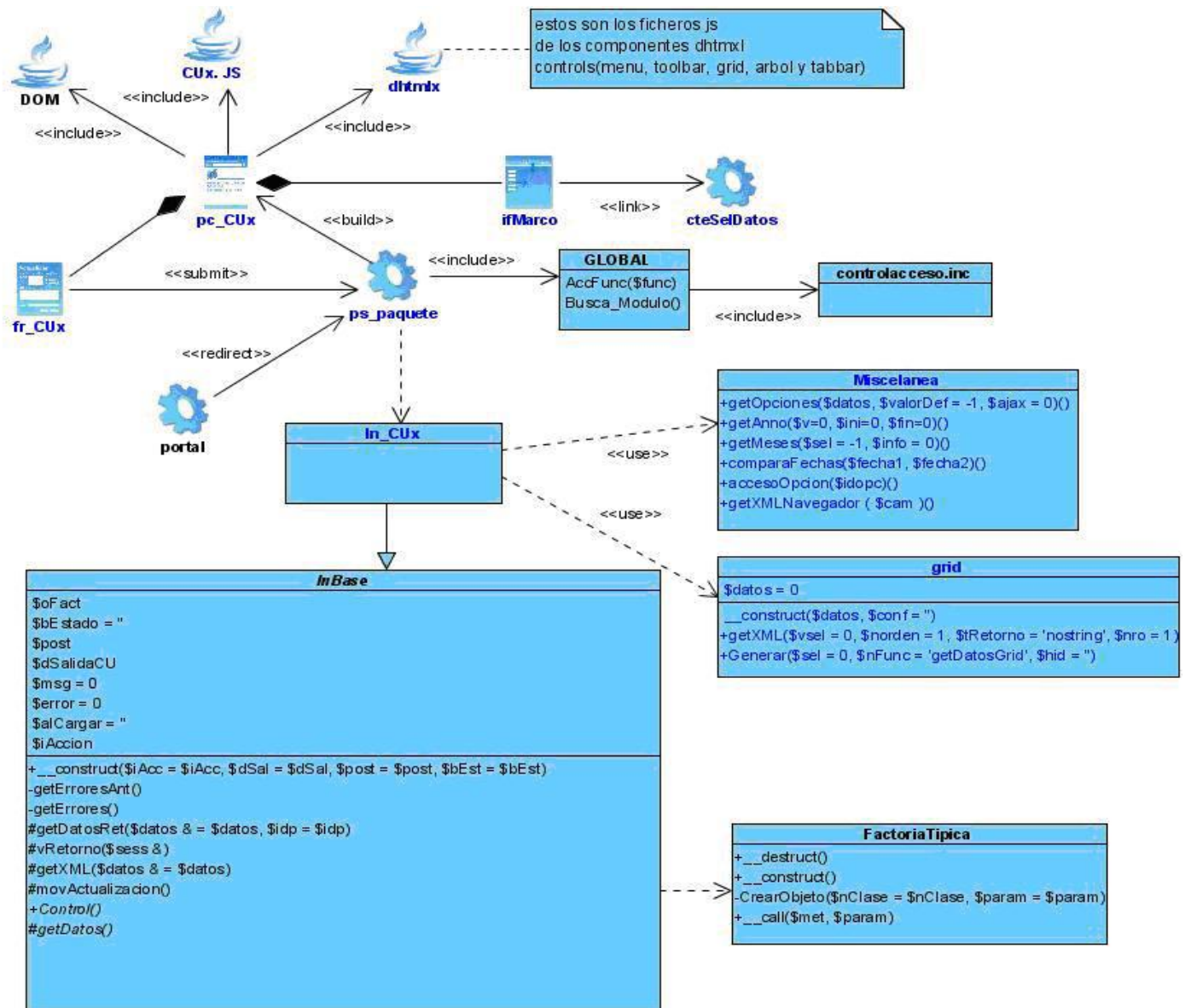
3.3 Diagrama de clases del diseño.

Un diagrama de clases muestra un conjunto de clases, interfaces, colaboraciones, así como sus relaciones de dependencia, generalización y asociación. Los diagramas de clases se utilizan para modelar la vista de diseño estática de un sistema. A continuación se presentará el diagrama de clases de diseño web del CU de la seguridad del sistema que se gestiona en un sistema aparte que brinda este servicio. Antes de mostrar el portal del sistema se verifica la seguridad del mismo y si el usuario está autenticado entonces es que se muestra el portal del sistema.

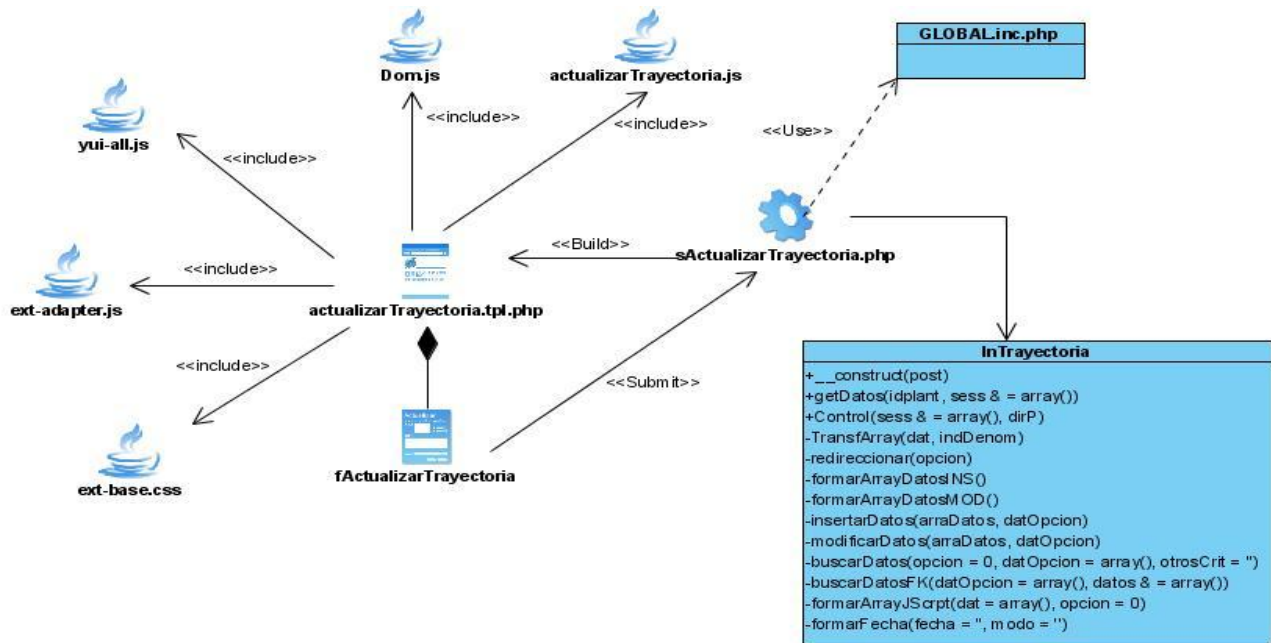


Se desarrolló el diagrama de clases del diseño web genérico, este presenta una página servidora (paquete), una página js (CUx.js), un formulario (fCUx) y una página de lógica de negocio (In_CUx) se representan en todos los diagramas de clases de todos los CU pero cambia el nombre de ellos. Por otra parte la página del Dom se incluye siempre para todos los CU. La página servidora del portal se usa en todos los CU ya que estos se pueden iniciar desde el portal principal del sitio.

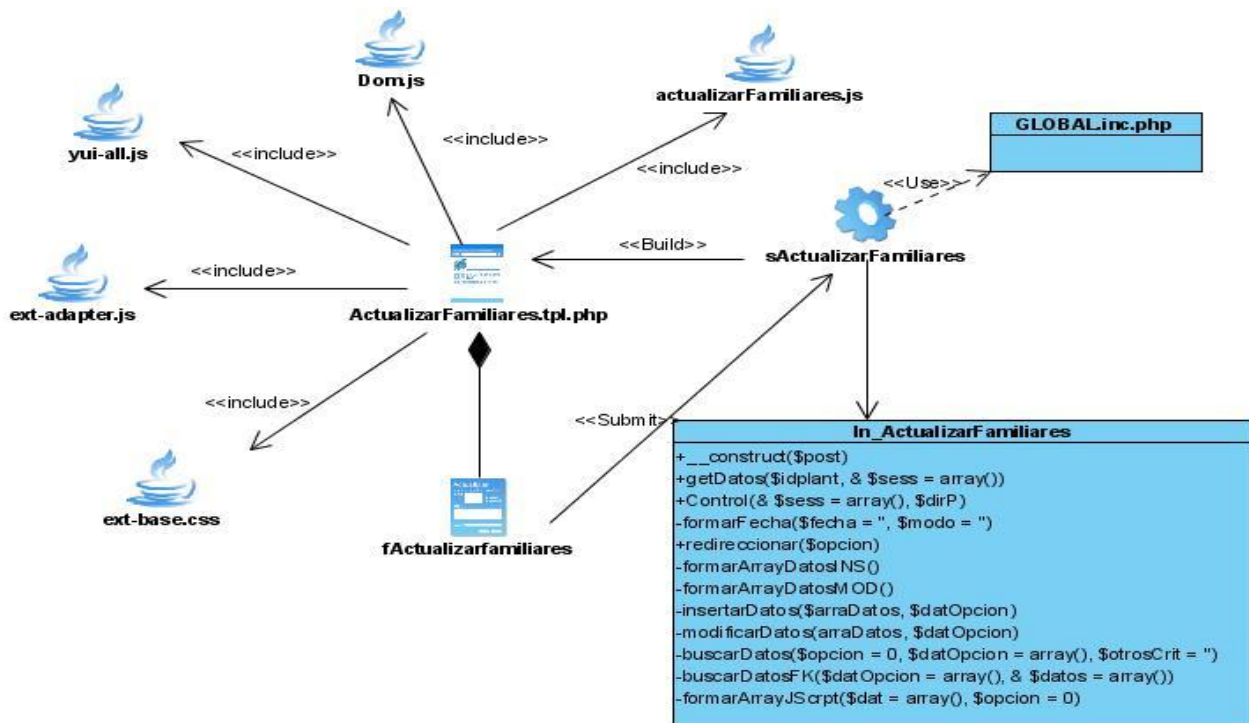
El CU incluye Grid cuando se usa un grid en la interfaz, y entonces se pone además el archivo JS dhtmlx. El frame lfmarco junto con el componente de Selección de datos (cteSelDatos) se usan solo en los CU que lo requieran trabajo con la fecha. Miscelánea se utiliza cuando se necesita incorporar contenido dinámico las listas. Factoría típica se utiliza siempre que se vaya a realizar alguna modificación sobre un nomenclador.



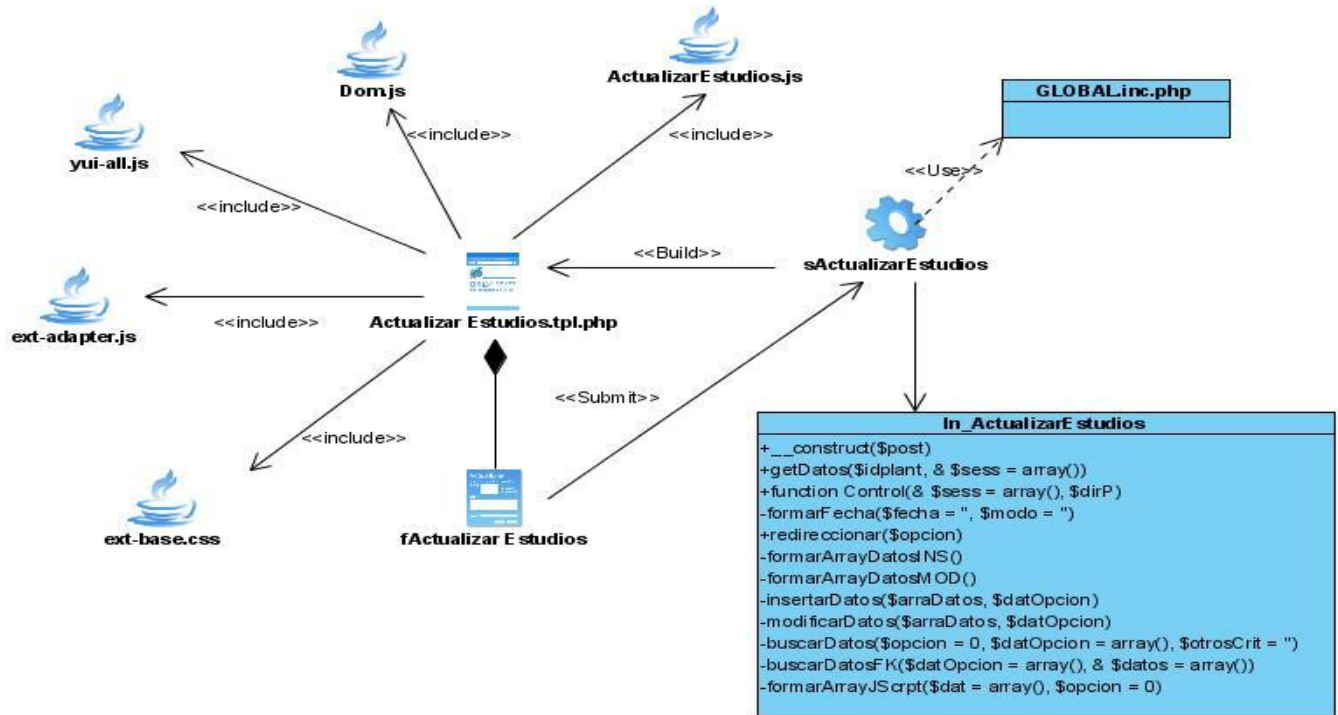
3.3.1 Caso de Uso Actualizar Trayectoria



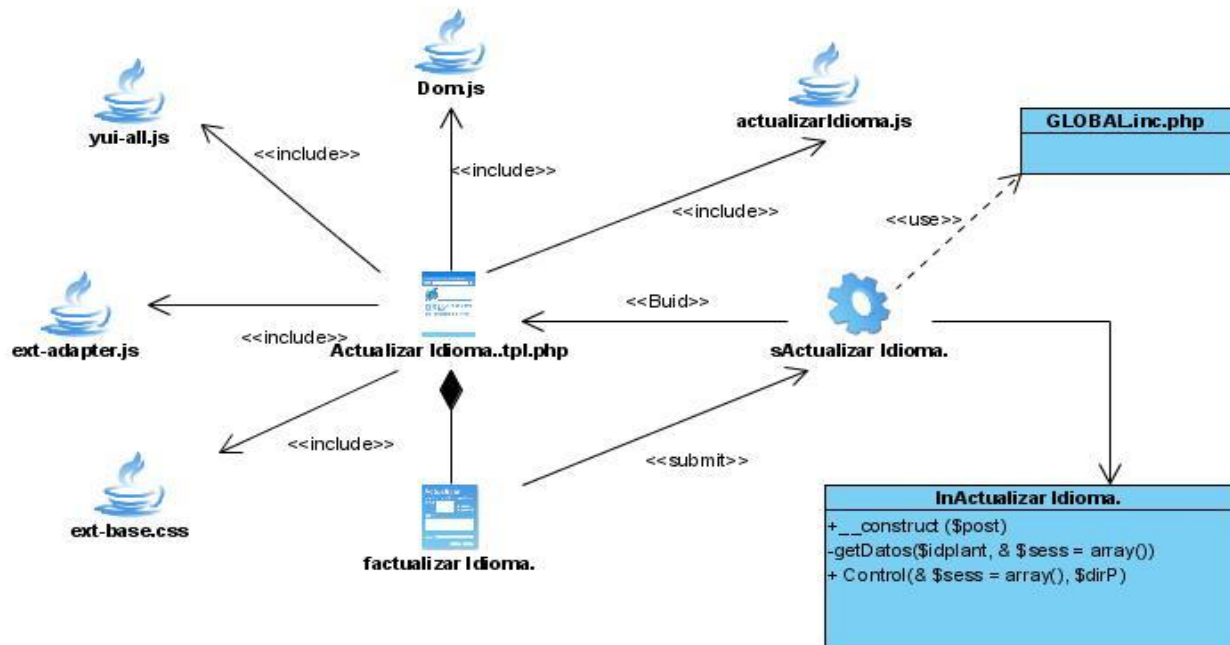
3.3.2 Caso de Uso Actualizar Familiares



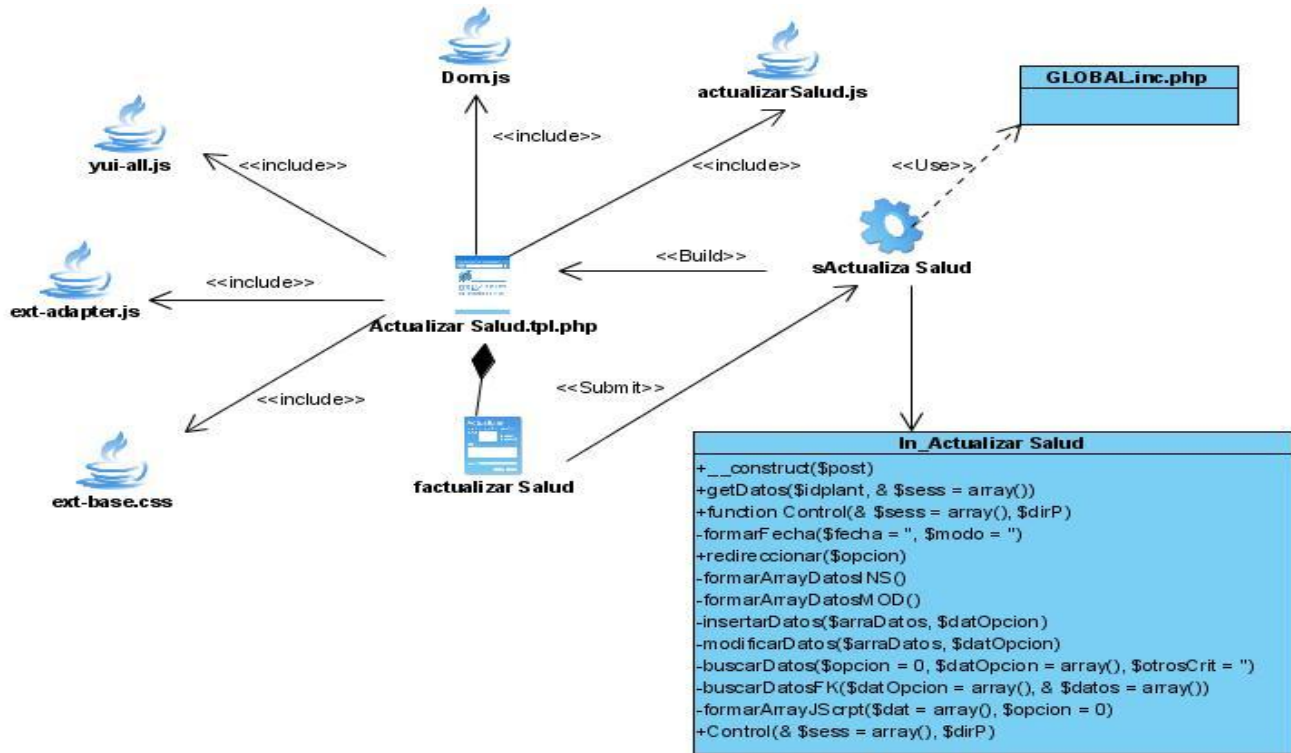
3.3.3 Caso de Uso Actualizar Estudios



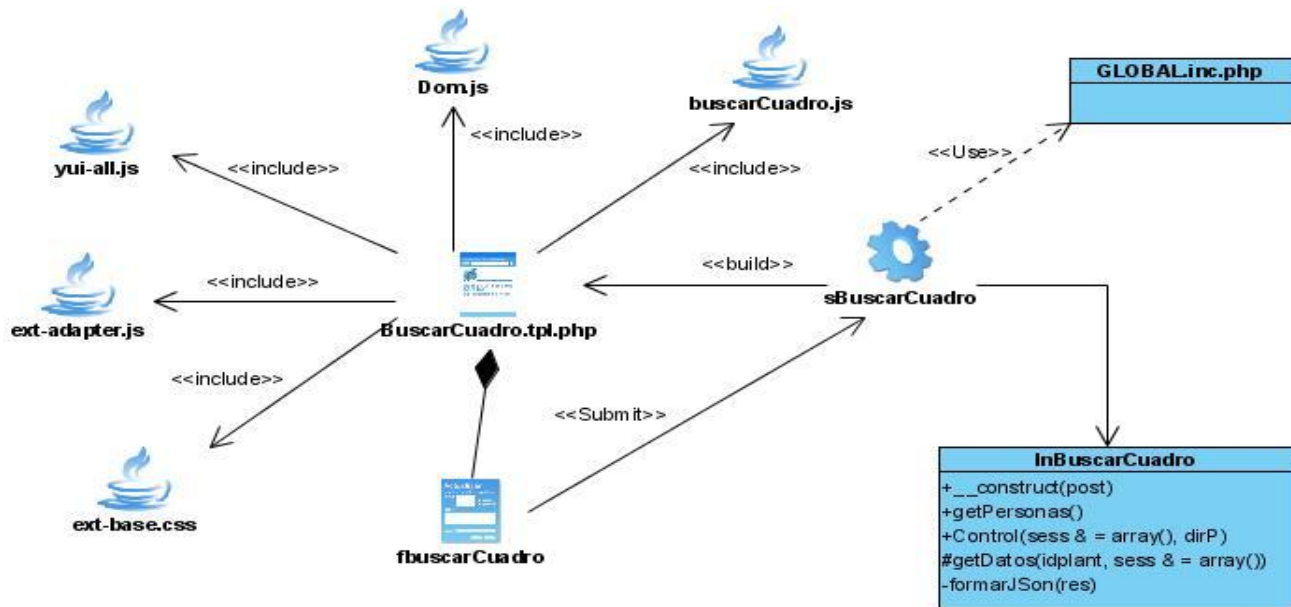
3.3.4 Caso de Uso Actualizar Idioma



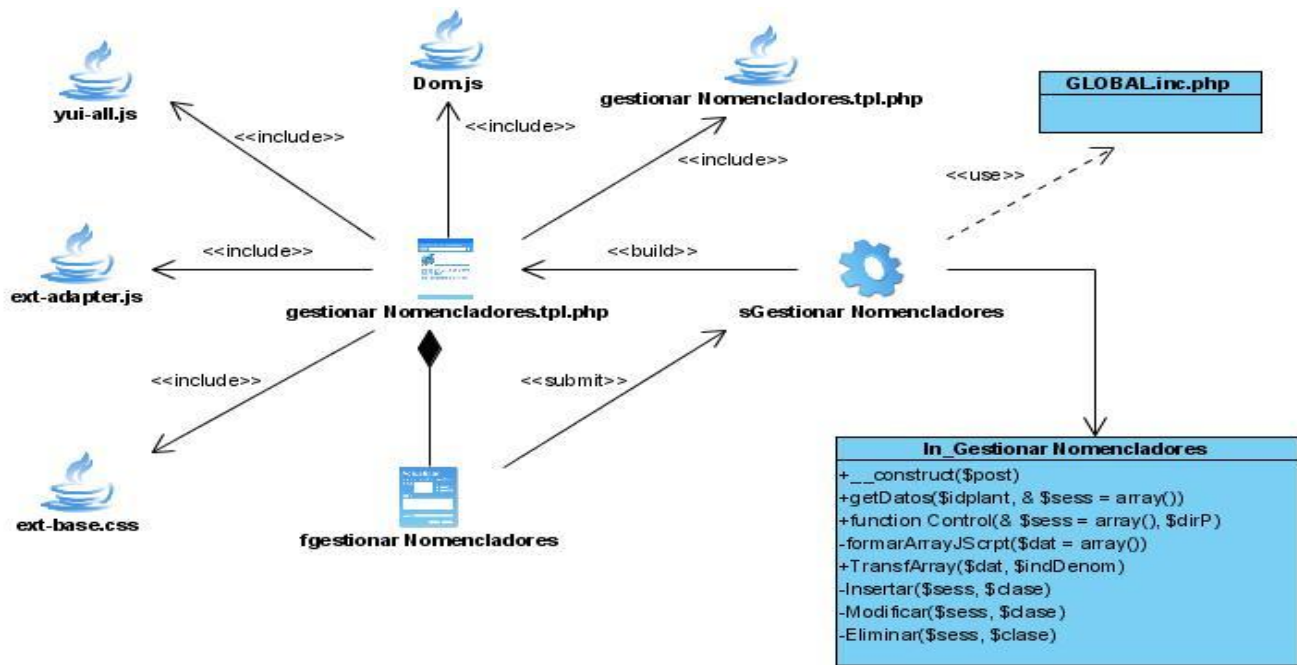
3.3.5 Caso de Uso Actualizar Salud



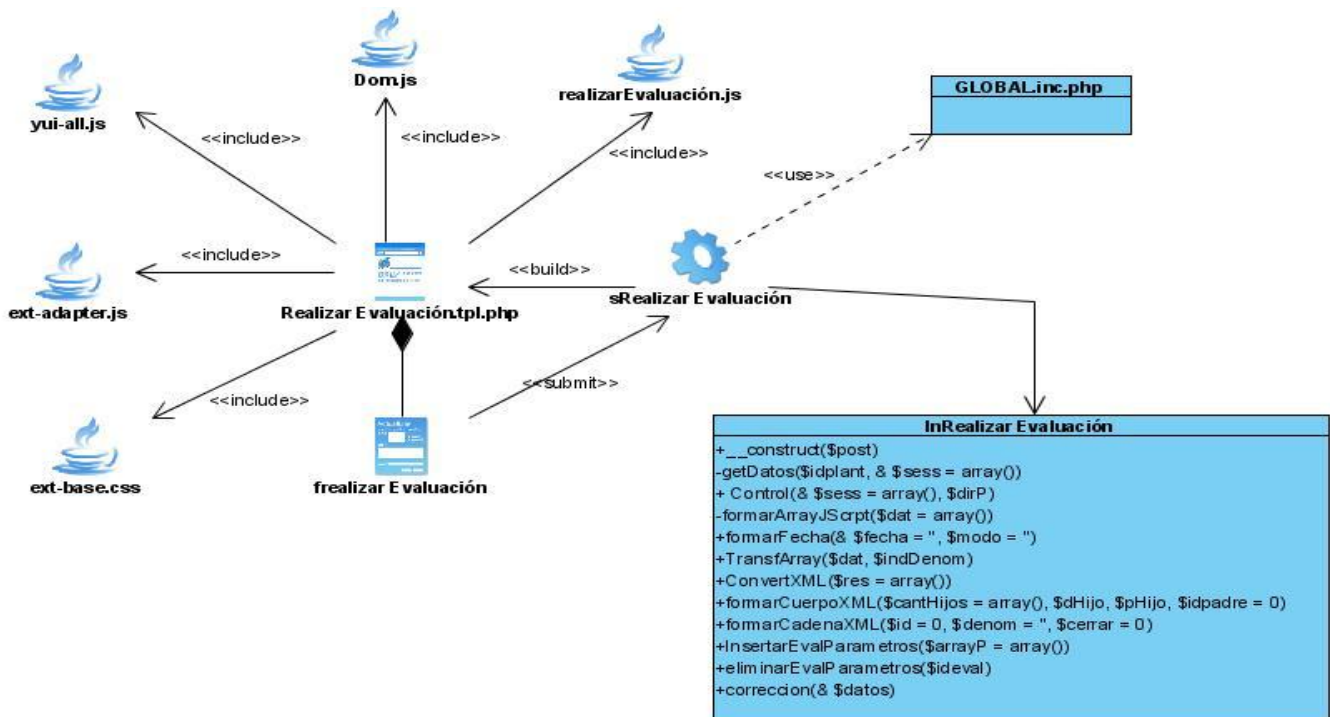
3.3.6 Caso de Uso Buscar Cuadro.



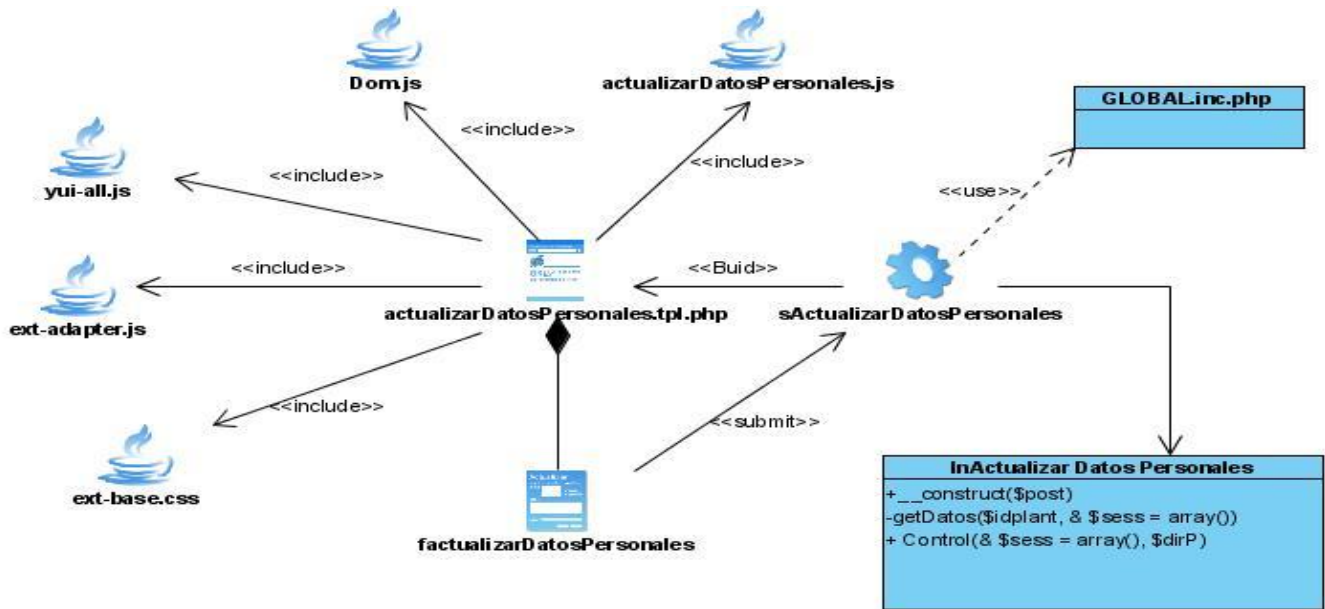
3.3.7 Caso de Uso Gestionar Nomencladores



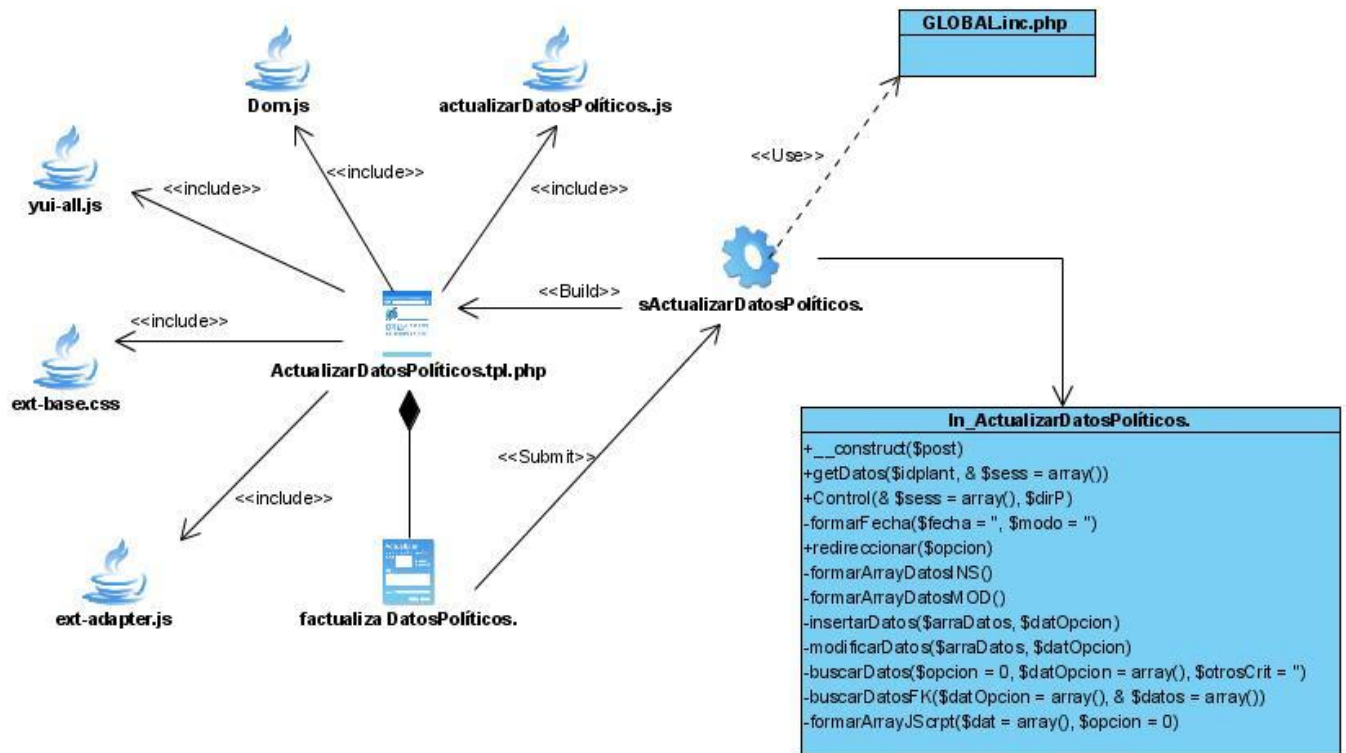
3.3.8 Caso de Uso Realizar Evaluación



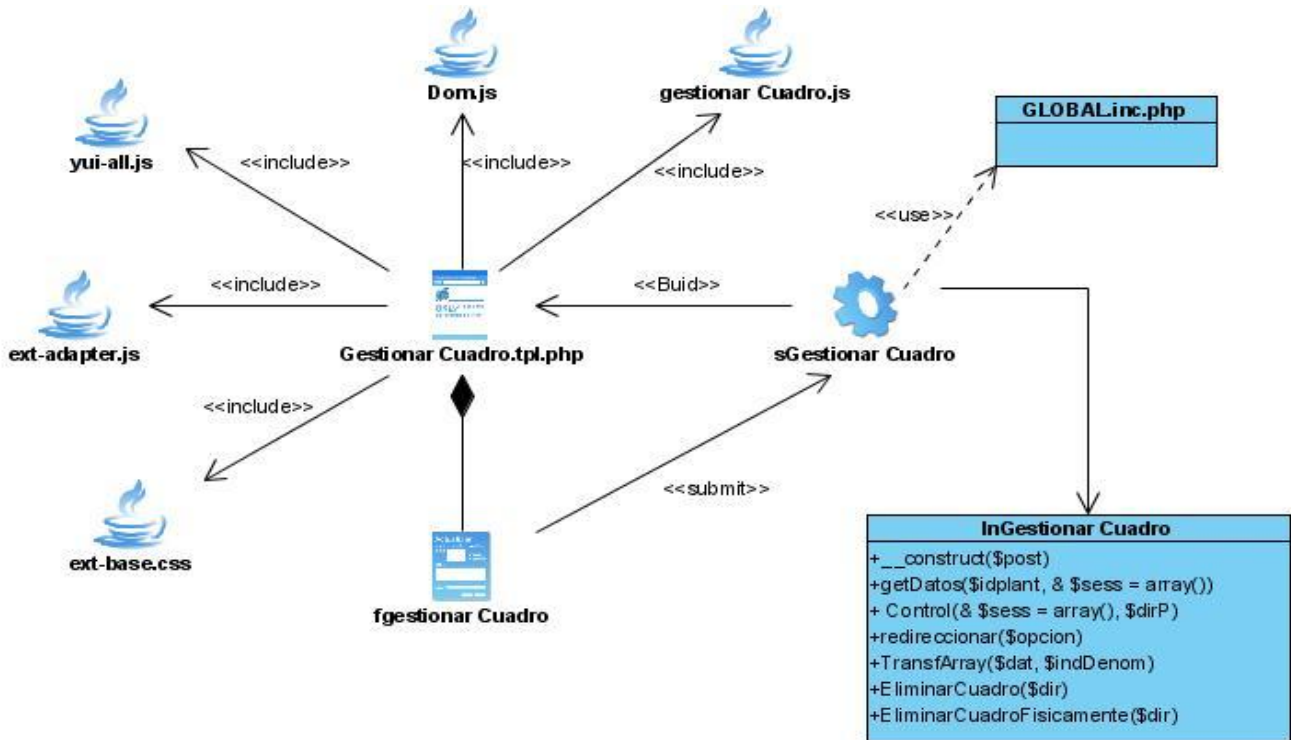
3.3.9 Caso de uso Actualizar Datos Personales



3.3.10 Caso de Uso Actualizar Datos políticos



3.3.11 Caso de uso Gestionar Cuadro



3.4 Diagramas de Interacción

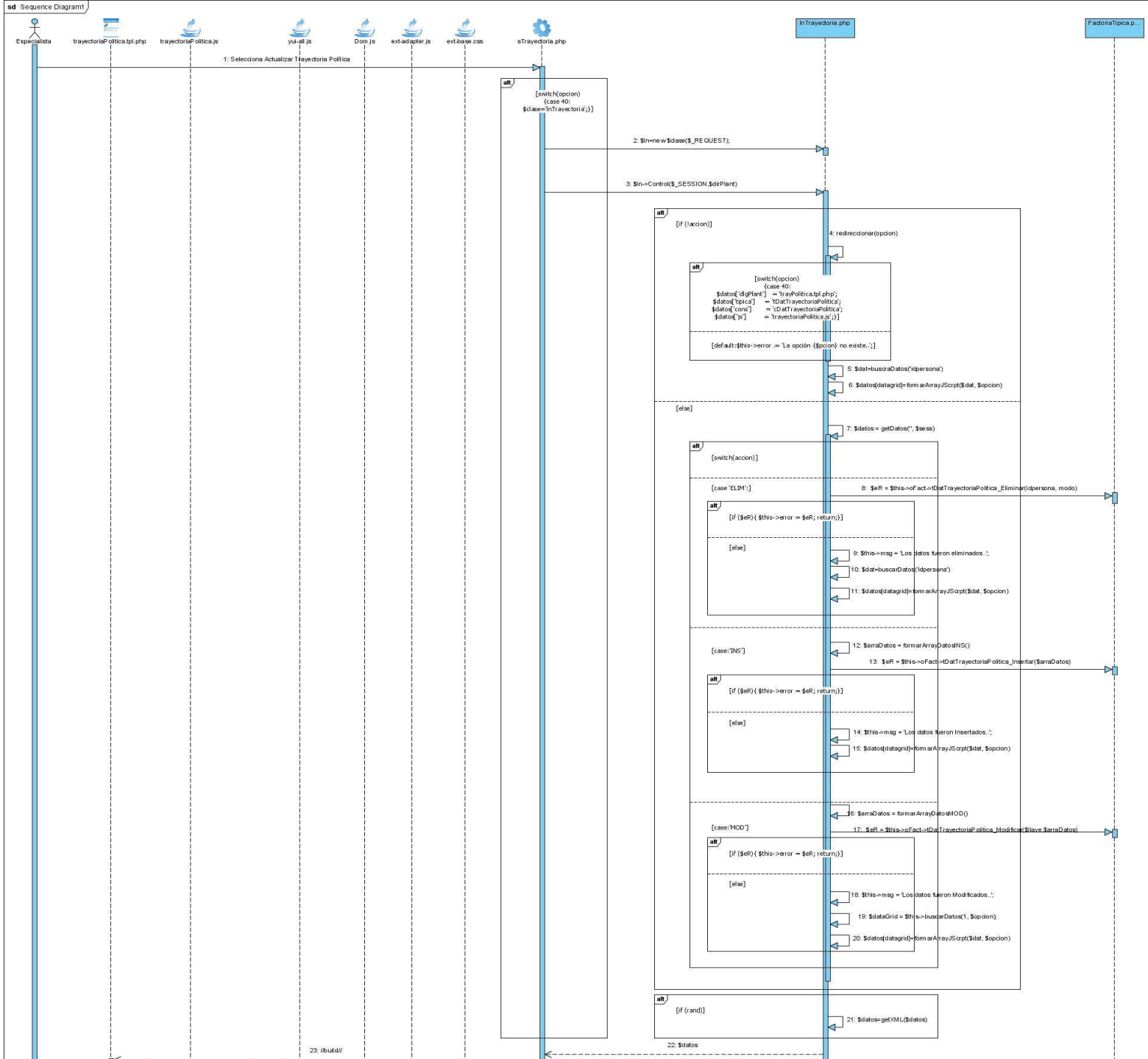
Un Diagrama de Interacción muestra una interacción, que consta de un conjunto de objetos y sus relaciones, incluyendo los mensajes que se pueden enviar entre ellos. Normalmente, los diagramas de interacción contienen objetos, enlaces y mensajes.

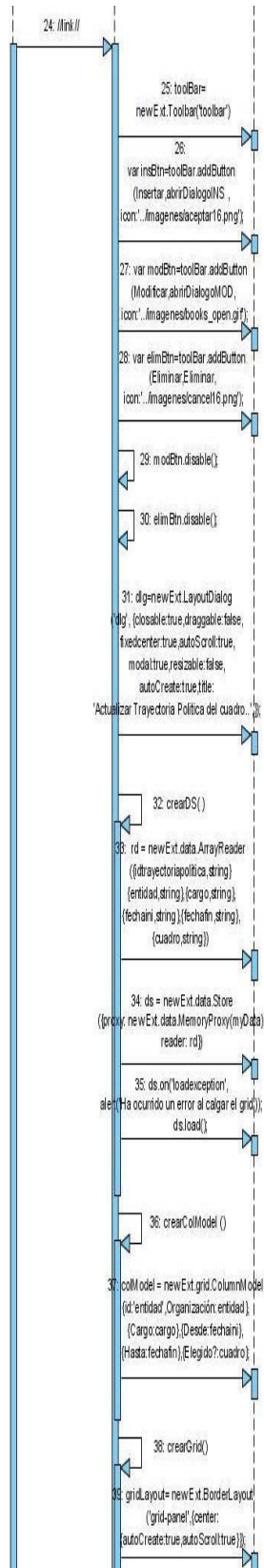
Este tipo de diagrama está compuesto por diagramas de colaboración y diagramas de secuencia. A continuación se desarrollarán los diagramas de secuencia.

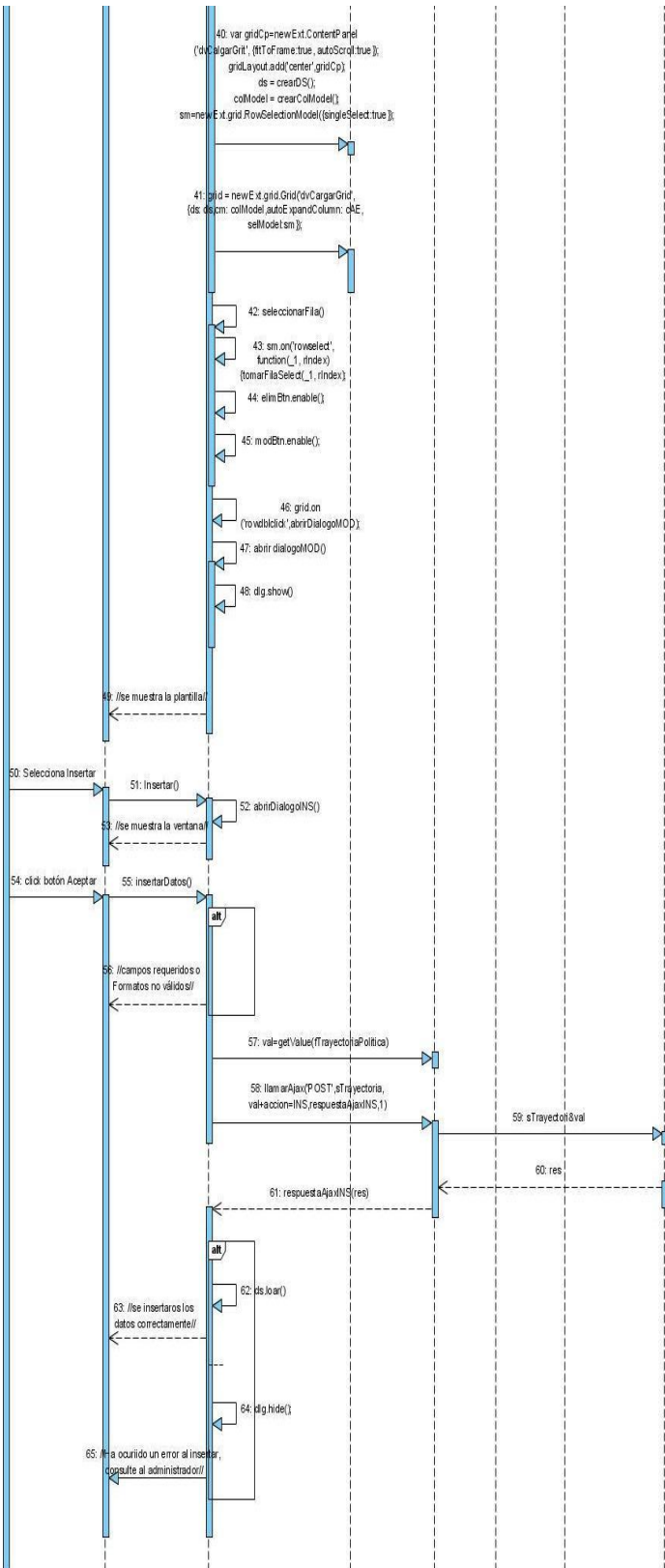
Diagramas Secuencia

Los diagramas de secuencia se destacan por la ordenación temporal de los mensajes. Algunas de sus características es que poseen una línea de vida del objeto y un foco de control. A continuación se presentarán los principales diagramas de secuencia desarrollados, se escogieron los más críticos de cada uno de los paquetes de CU sistema.

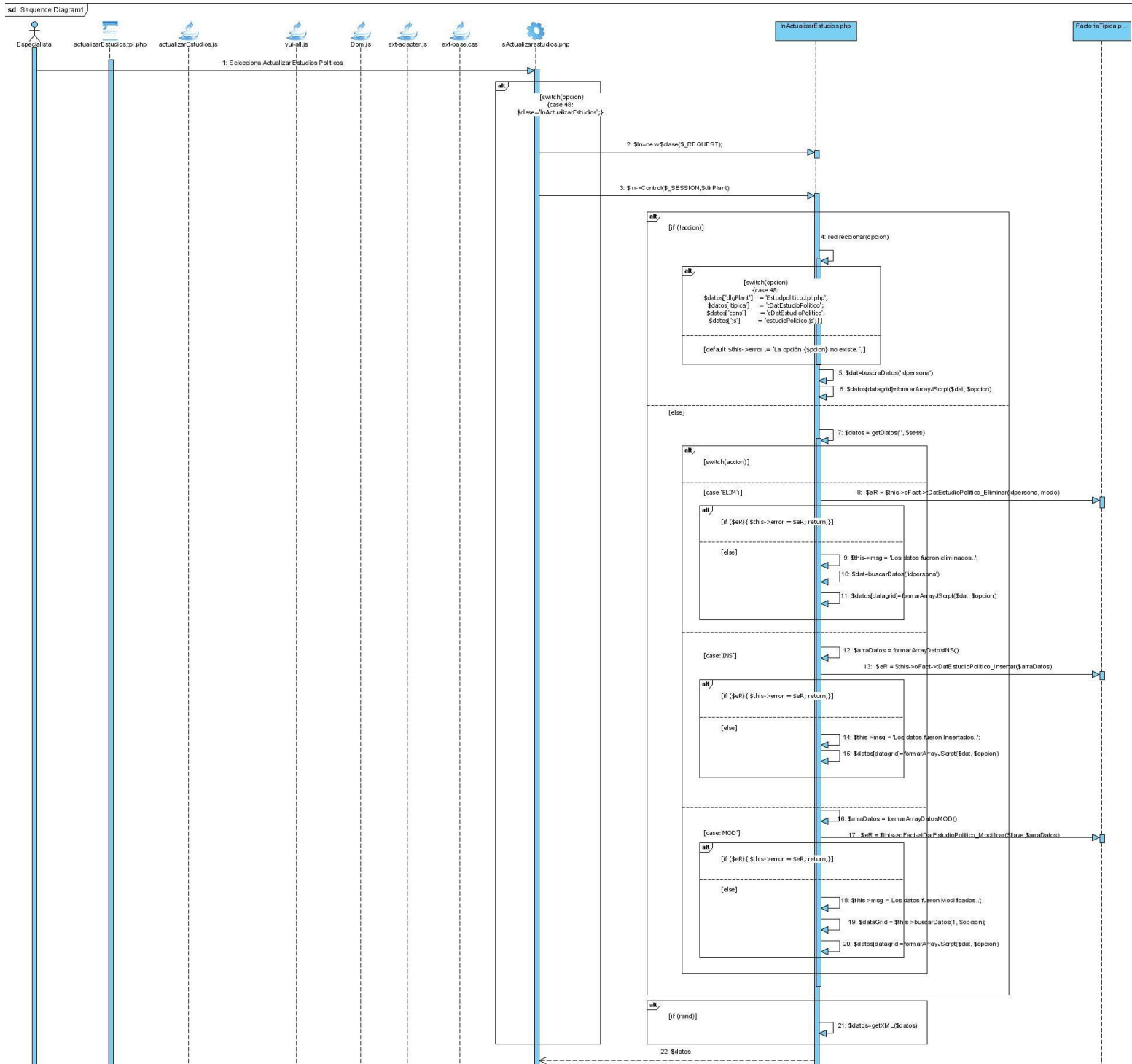
3.4.1 Diagrama de Secuencia Actualizar Trayectoria Política

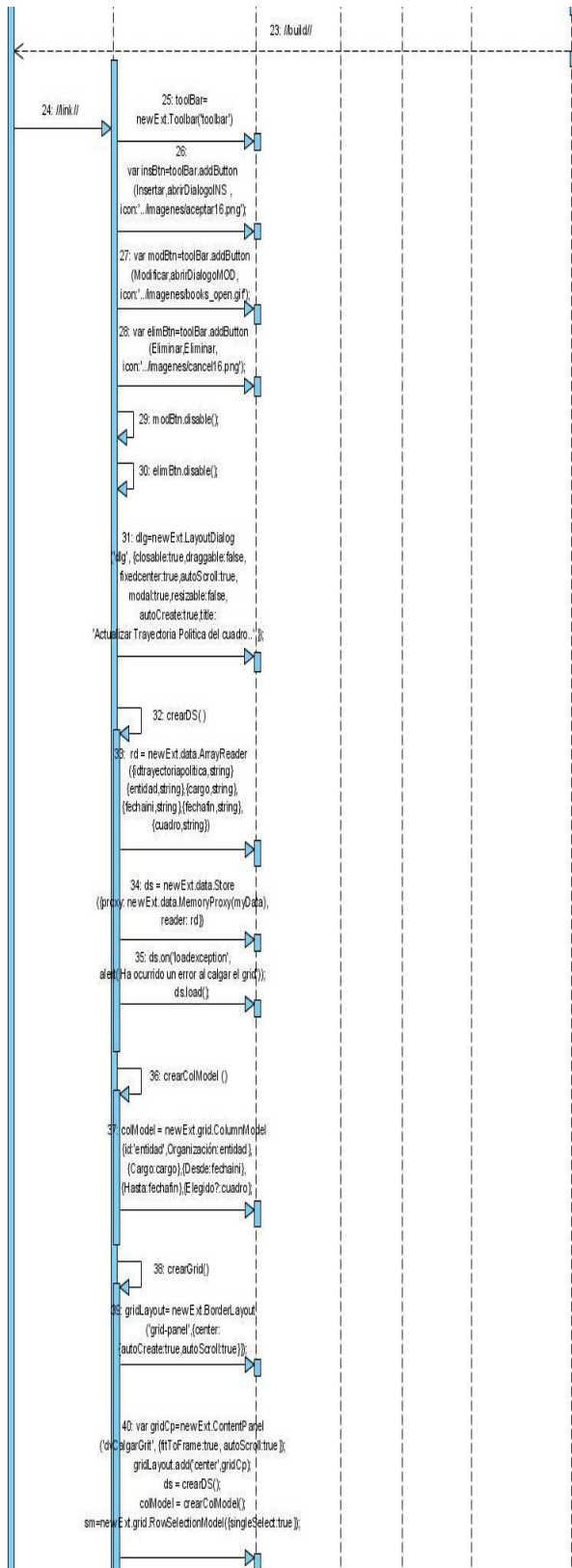


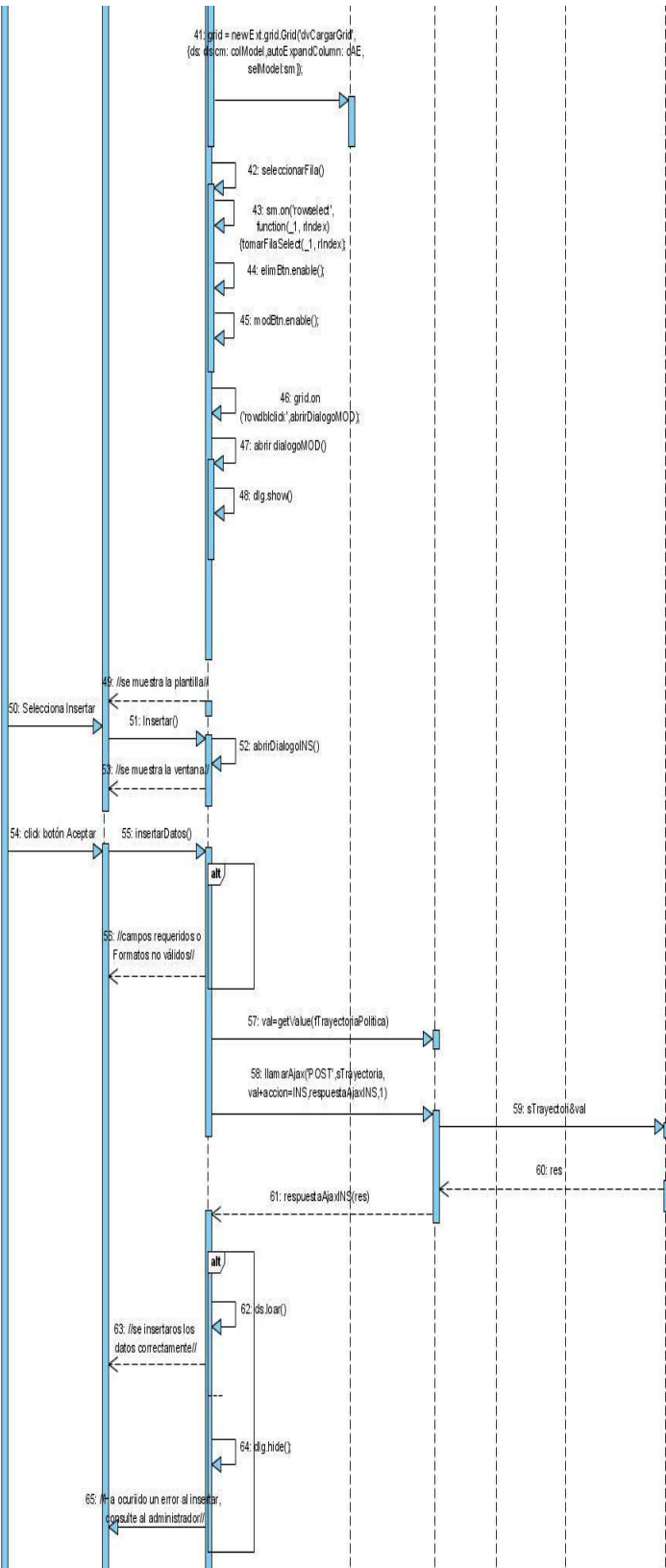




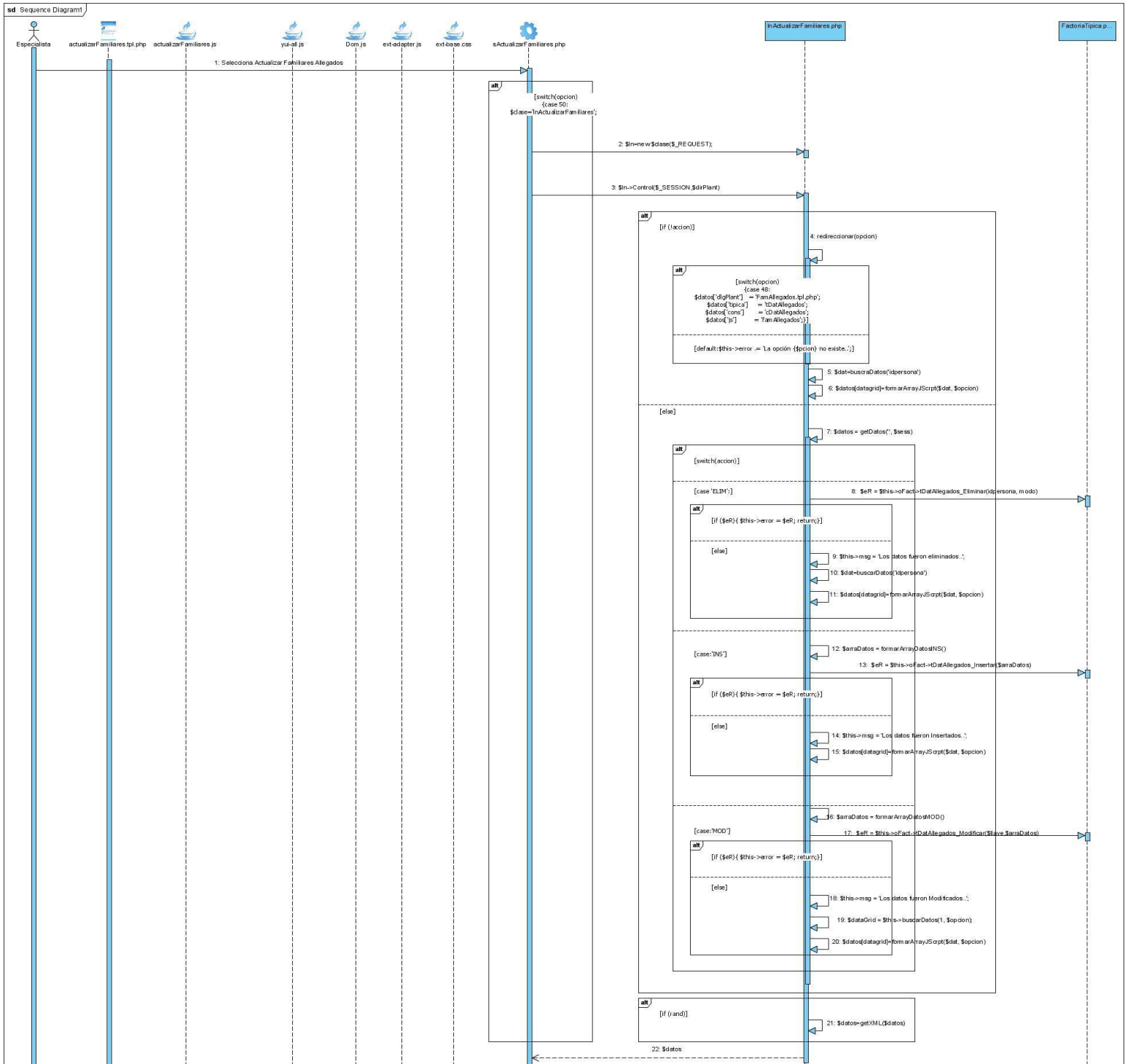
3.4.2 Diagrama de Secuencia Actualizar Estudios

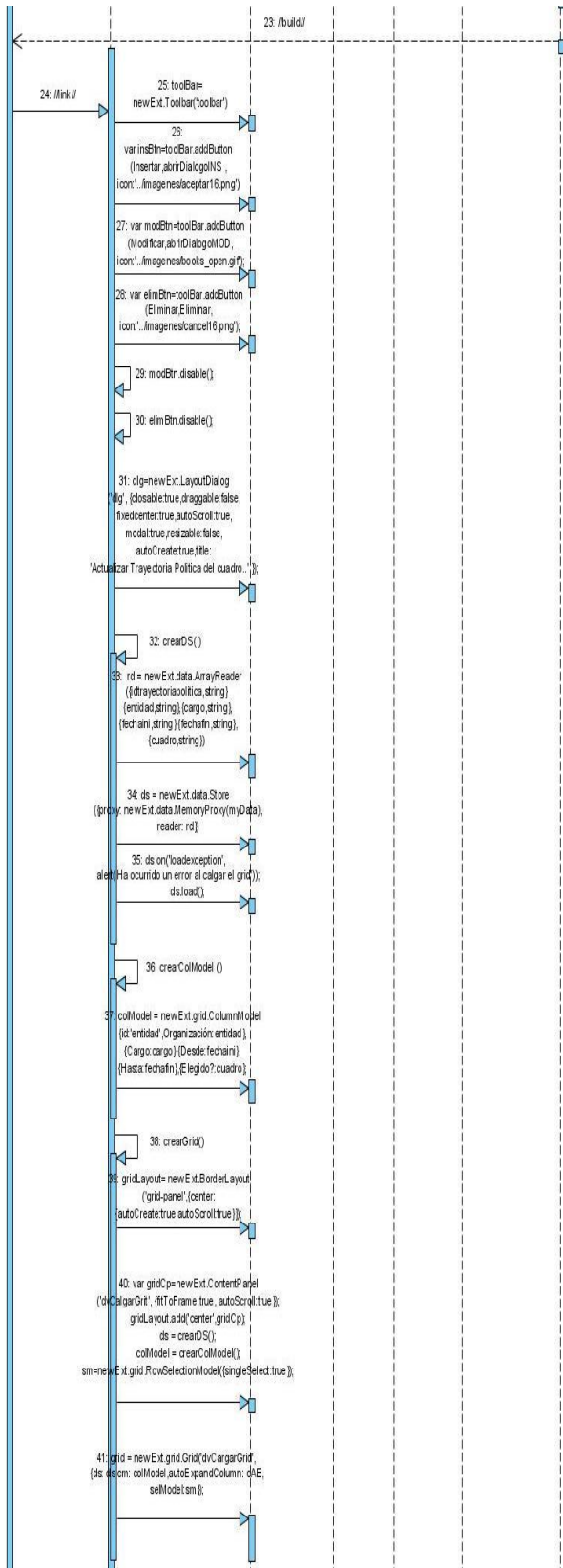


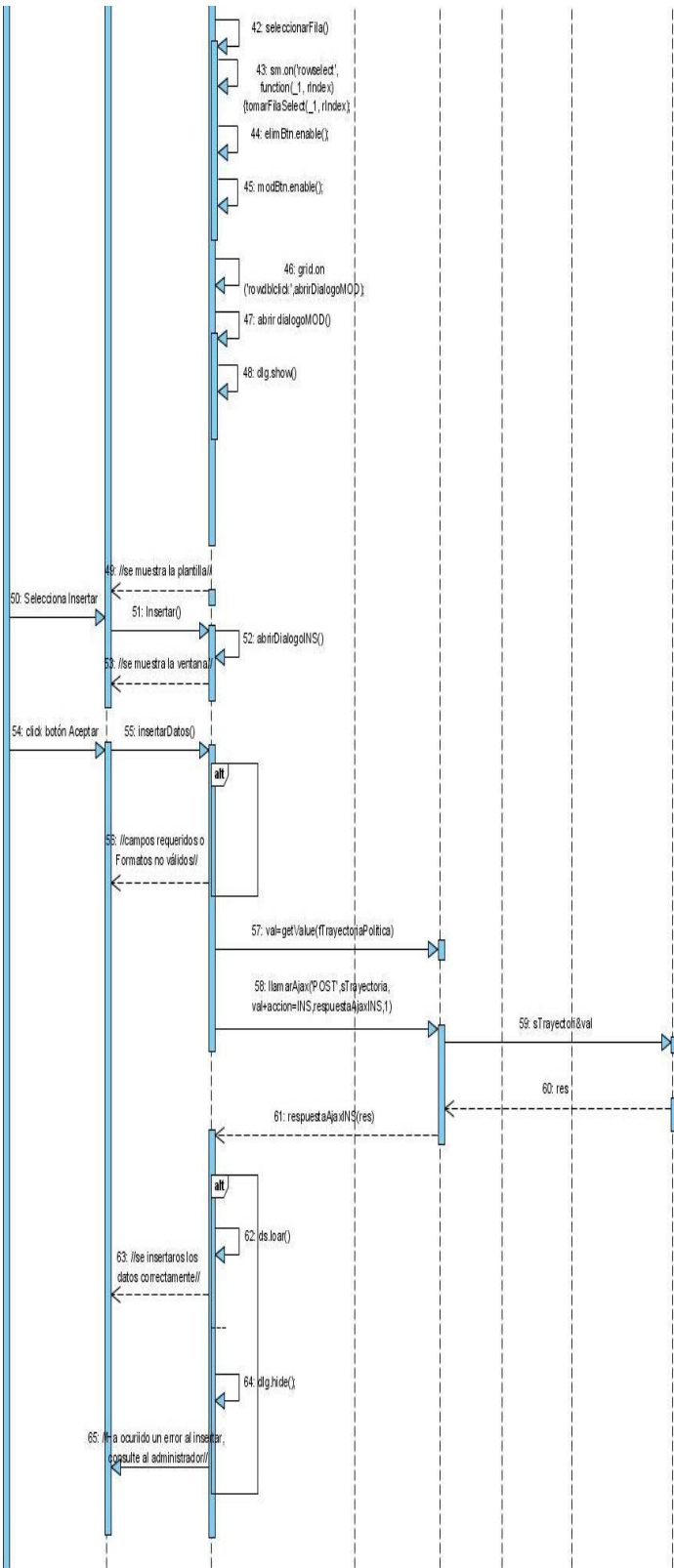




3.4.3 Diagrama de Secuencia Actualizar Familiares







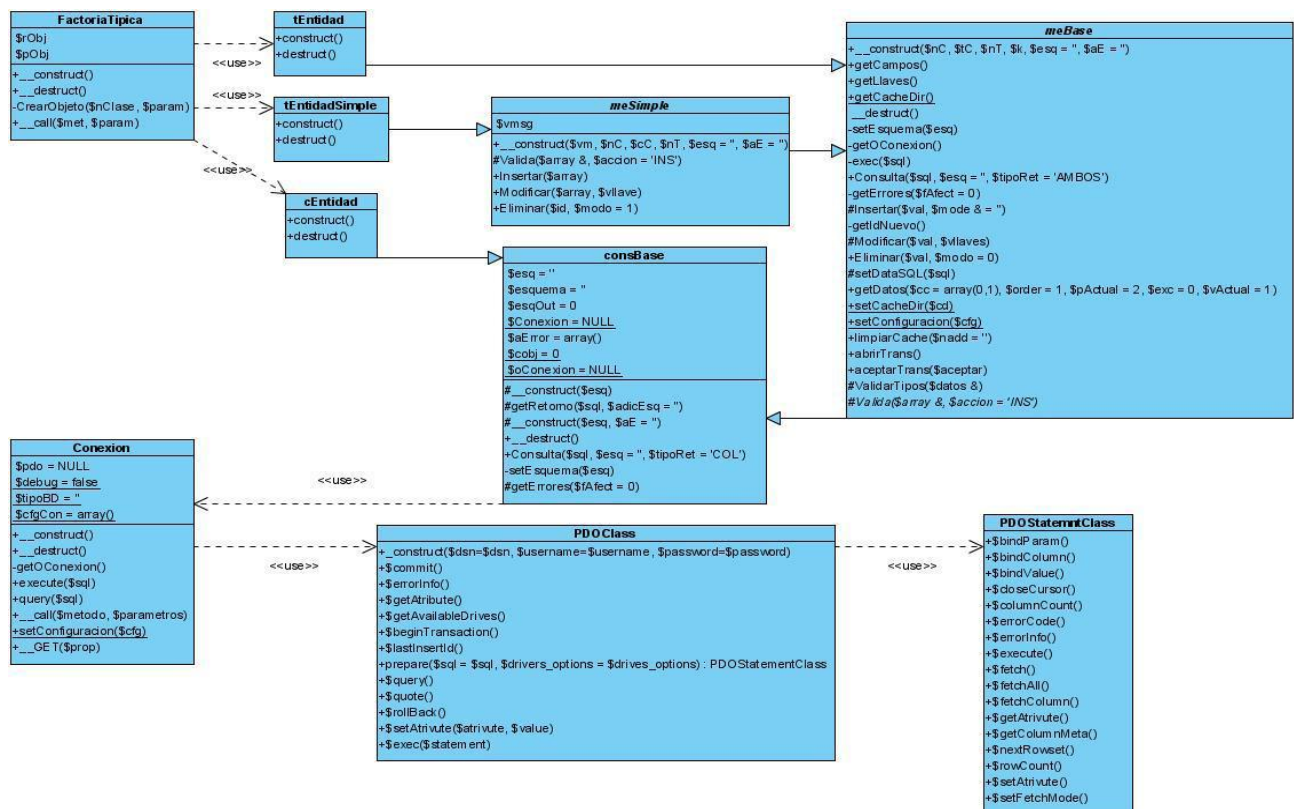
3.5 Mecanismo de diseño de Acceso a Datos

El modelo de acceso a datos tiene una estrecha relación con los estándares de base de datos, los que ayudan a identificar grupos o tipos de entidades según el prefijo, por ejemplo: las entidades de datos tienen prefijo “dat_” y los nomencladores o clasificadores “nom_” y las tablas de los históricos “his_”. Para poder identificar patrones que puedan convertirse luego en clases abstractas, como meSimple, es preciso realizar un estudio detallado de todas las entidades con las que se va a relacionar el modelo de acceso a datos.

Dentro de los dos grupos más grandes, específicamente en el de los nomencladores, existe un patrón básico al que se ha llamado “entidades simples”, las cuales son fáciles de identificar pues siempre tienen la estructura siguiente: identidad, denentidad, y actual. Por tanto, las operaciones básicas tendrán un comportamiento exacto en el 99% de los casos y de ahí es que surge meSimple.

El mecanismo de acceso a datos utilizado contiene un conjunto de clases que interactúan para dar acceso y manipulación de los datos de la persistencia desde el nivel más bajo, es decir, utilizando los objetos nativos brindados por el entorno de desarrollo PHP como son la extensión PDO que define una ligera interfaz, para el acceso a bases de datos., siguiendo así hasta la abstracción del acceso a datos a través de meBase de la cual heredan las clases particulares del sistema como Típicas y meSimples.

Para dar la responsabilidad a una clase que encapsulara las instancias de estos objetos se definió la clase Factoría Típica, que está concebida para instanciar todas las típicas del sistema para utilizarlas de algún modo.



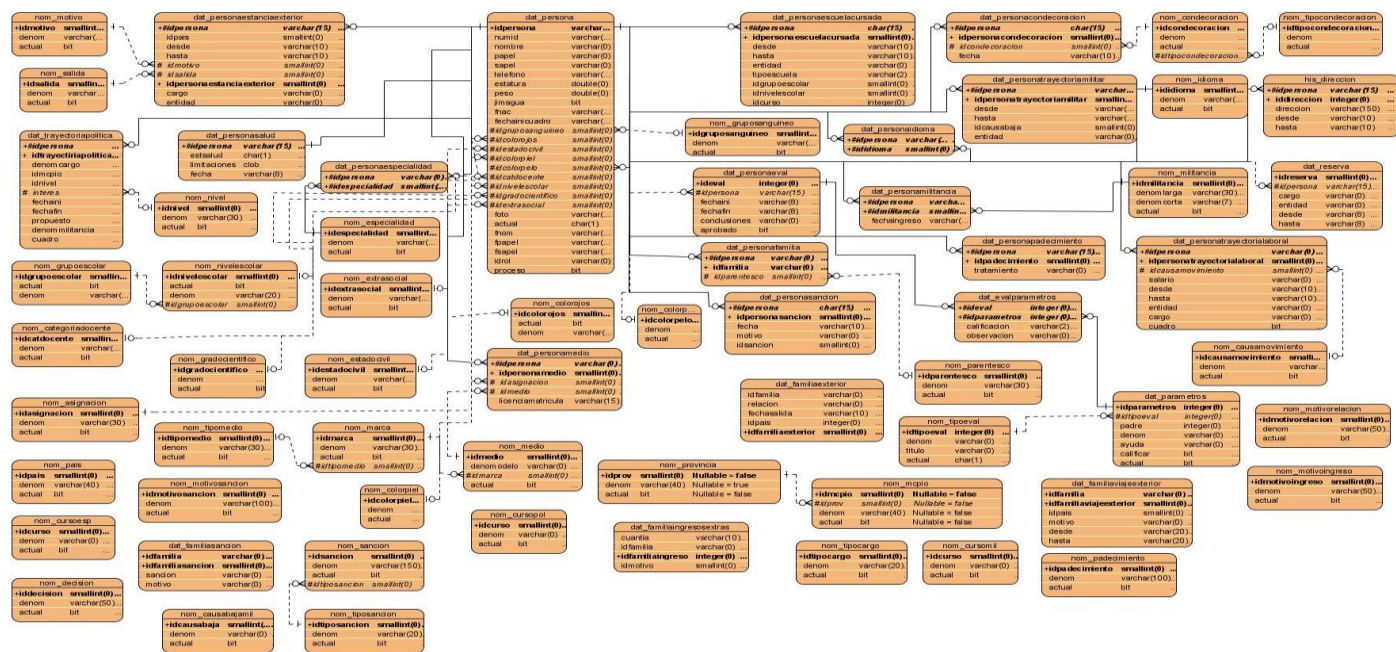
3.5.1 Descripción de las clases

Clases	Descripción
[Plantillas]	Son las clases que contienen en su estructura todo el contenido estático de una página Web. Estas serán empleadas para generar las páginas clientes.
[Lógica del negocio]	Son las clases con prefijo ln y encapsulan en sus métodos todas las Operaciones específicas para cada caso de uso del sistema.
Global	Fichero con código PHP que contiene instancias y funciones que serán empleadas por todas las páginas servidoras.
[Típicas]	Son las clases con prefijo t, existe una típica para cada entidad de la base de datos. Es una clase que representa a las clases típicas en general de la aplicación. Para la implementación de esta clase se decidió aplicar el patrón de diseño Table Data Gateway, que consiste en crear una instancia por cada tabla existente en la BD. Sus métodos consisten en las operaciones básicas que se realizan sobre estas tablas, insertar, modificar y eliminar. Hereda de la clase abstracta meBase.
Típica Simple	Es una clase que representa a las clases típicas (nomencladores simples) en general de la aplicación. Estas típicas son de una implementación muy sencilla, pues la mayoría de las líneas que normalmente habían que codificar quedaron encapsuladas en la clase base de las mismas. Para la implementación de esta clase se decidió aplicar el patrón de diseño Table Data Gateway, que consiste en crear una instancia por cada tabla existente en la BD. Sus métodos consisten en las operaciones básicas que se realizan sobre estas tablas, insertar, modificar y eliminar. Hereda de la clase abstracta meSimple.
Factoría Típica	Es la clase encargada de gestionar la creación y utilización de objetos de tipo [Típicas]. Clase que implementa la interfaz del modelo de persistencia con el resto de los subsistemas. A través de esta clase se crean y se manipulan los objetos de las típicas simples, los nomencladores y las demás típicas. Es una puerta entre la capa de Acceso a Datos y la capa de Lógica de Negocio. Implementa un método de instanciación de clases típicas
consBase	Es la clase base de la cual heredan todas las de acceso a datos. Implementa de forma general las operaciones a realizar con consultas. Además le aporta el contenido dinámico a las plantillas.
mEntidad	Es la clase base de la cual heredan todas las típicas. Implementa de forma general las operaciones básicas a realizar con las entidades del sistema y maneja las conexiones a la Base de Datos. Emplean las clases PDO y PDOStatement.

meSimples	Clase abstracta heredera de meBase y es además la clase base de la cual heredan todas las Típicas que responden a los nomencladores básicos o simples. Redefine las operaciones básicas con la funcionalidad de Validación dada. Redefine las operaciones básicas que pudieran realizarse a una entidad (insertar, eliminar, modificar) para los nomencladores simples.
meBase	Es una clase abstracta; clase base de la cual heredan meSimples y las Típicas complejas. Hereda la operación de consulta de consBase. Implementa de forma general las operaciones básicas que pudieran realizarse a las entidades del sistema (insertar, modificar y/o eliminar).
PDO y PDOStatement	Son clases pertenecientes a la librería PHP que permiten la gestión de la Conexión y la consulta a la Base de Datos.
Conexión	Es la clase responsable de establecer la conexión con el servidor de la BD a través de un objeto PDO de la librería de PHP.
Miscelánea	Clase que implementa una serie de funciones de carácter general para usar en las lógicas de negocio. Incorpora contenido dinámico las listas.
cClaseconsulta	Es una clase que representa a las clases consultas en general de la aplicación. Clase que hereda de la clase abstracta consBase. Existe una clase consulta para cada entidad de la base de datos.

3.6 Diseño de la Base de datos

En el diseño de la Base de datos (BD) se muestra las relaciones existentes entre todas las entidades de la DB, con todos sus atributos, tipo y esquema al que pertenecen.



Descripción de las tablas

A continuación se mostrarán las tablas de la BD del esquema cuadro que se utilizan con la descripción, atributos que poseen, tipo y una breve descripción de estos.

3.6.1 Tabla dat_persona

Nombre: dat_persona		
Descripción: Se almacenan los datos de la persona		
Atributo	Tipo	Descripción
idpersona	VARCHAR	Identificador de la persona
numid	VARCHAR	Carnet de Identidad de la persona
nombre	VARCHAR	Nombre de la persona
papel	VARCHAR	Primer apellido de la persona
sapel	VARCHAR	Segundo apellido de la persona
telefono	VARCHAR	Teléfono de la persona
estatura	DOUBLE PRECISION	Estatura de la persona
peso	DOUBLE PRECISION	Peso de la persona
jimagua	BIT	Si es jimagua
fnac	VARCHAR	Fecha de nacimiento de la persona
fechainiccuadro	VARCHAR	Fecha de inicio del cuadro
idgruposanguineo	SMALLINT	Llave foránea para identificar el grupo sanguíneo
idcolorojos	SMALLINT	Llave foránea para identificar el Color de los ojos
idestadocivil	SMALLINT	Llave foránea para identificar el estado civil de la persona
idcolorpiel	SMALLINT	Llave foránea para identificar el color de la piel de la persona
idcolorpelo	SMALLINT	Llave foránea para identificar el color del pelo de la persona
idcatdocente	SMALLINT	Llave foránea para identificar la categoría de la persona
idnivelescolar	SMALLINT	Llave foránea para identificar el nivel escolar de la persona
idgradocientifico	SMALLINT	Llave foránea para identificar el grado científico de la persona

idextrasocial	SMALLINT	Llave foránea para identificar el estrato social de la persona
foto	VARCHAR	Foto de la persona
fnom	VARCHAR	Nombre de la persona
fpapel	VARCHAR	Primer apellido de la persona
fsapel	VARCHAR	Segundo apellido de la persona
idrol	VARCHAR	Identificador del rol
proceso	BIT	Si se le esta haciendo el proceso de cuadro
sexo	CHAR	Sexo a la que pertenece la persona

3.6.2 Tabla dat_familiaexterior.

Nombre: dat_familiaexterior		
Descripción: Se almacenan Los datos de familia que se encuentra en el exterior		
Atributo	Tipo	Descripción
idfamilia	VARCHAR	Identificador de la familia
relacion	VARCHAR	Relación que tiene con la persona
idfamiliaexterior	VARCHAR	Identificador de la tabla dat_familiaexterior

3.6.3 Tabla dat_familiasancion.

Nombre: dat_familiasancion		
Descripción: Se almacenan los datos de los familiares sancionados		
Atributo	Tipo	Descripción
idpersona	VARCHAR	Identificador de la persona
idpersonatrayectoriamilitar	SMALLINT	Identificador de la trayectoria militar
desde	VARCHAR	Fecha desde que comenzó la sanción
Hasta	VARCHAR	Fecha hasta que es la sanción
identidad	BIGINT	Identidad de la persona
idcausabaja	SMALLINT	Identificador de la causa baja
entidad	VARCHAR	Identificador de la Entidad
idgrado	SMALLINT	Identificador del grado

3.6.4 Tabla dat_familiaviajeexterior.

Nombre: dat_familiaviajeexterior		
Descripción: Se almacenan los datos de la persona		
Atributo	Tipo	Descripción
idfamilia	VARCHAR	Identificador de los familiares
idfamiliaviajeexterior	SMALLINT	Identificador de la tabla
idpais	SMALLINT	Identificador del país
motivo	VARCHAR	Motivo por el cual viaja
desde	VARCHAR	Fecha desde que comenzó el viaje
Hasta	VARCHAR	Fecha hasta que es el viaje

3.6.4 Tabla dat_parametros.

Nombre: dat_parametros		
Descripción: Se almacenan los parámetros		
Atributo	Tipo	Descripción
idparametros	INTEGER	Identificador de la Tabla
idtipoeval	INTEGER	Identificador de la evaluación
padre	INTEGER	Nombre de la persona
denom	VARCHAR	Denominación de los parametros
ayuda	VARCHAR	Ayuda que se propone

3.6.5 Tabla dat_personacondecoracion.

Nombre: dat_personacondecoracion		
Descripción: Se almacenan las condecoraciones de la persona		
Atributo	Tipo	Descripción
idpersona	VARCHAR	Identificador de la persona
idpersonacondecoracion	SMALLINT	Identificador de la tabla
idcondecoracion	SMALLINT	Identificador de la condecoración
fecha	VARCHAR	Fecha

3.6.6 Tabla dat_personaescuelacursada.

Nombre: dat_personaescuelacursada		
Descripción: Se almacenan los datos de las escuelas cursadas por la persona		
Atributo	Tipo	Descripción
idpersona	VARCHAR	Identificador de la persona
idpersonaescuelacursada	SMALLINT	Identificador de la tabla
desde	VARCHAR	Fecha en que comenzó
Hasta	VARCHAR	Fecha donde termino
identidad	SMALLINT	Identidad de la persona
entidad	VARCHAR	Entidad a la que pertenece esa persona
tipoescuela	VARCHAR	Tipo de escuela
idgruposcolar	SMALLINT	Identificador del Grupo escolar
idnivelescolar	SMALLINT	Identificador del nivel escolar
idcurso	INTEGER	Identificador del curso

3.6.7 Tabla dat_personaespecialidad.

Nombre: dat_personaespecialidad		
Descripción: Se almacenan las llaves		
Atributo	Tipo	Descripción
idpersona	VARCHAR	Identificador de la persona
idespecialidad	SMALLINT	Identificador de la especialidad

3.6.8 Tabla dat_personaestanciaexterior.

Nombre: dat_personaestanciaexterior		
Descripción: Se almacenan los datos de la persona		
Atributo	Tipo	Descripción
idpersona	VARCHAR	Identificador de la persona
idpais	SMALLINT	Identificador del país
desde	VARCHAR	Fecha en que comenzó
Hasta	VARCHAR	Fecha donde termino
idsalida	SMALLINT	Identificador del Motivo de la salida

idpersonaestanciaexterior	SMALLINT	Identificador de la tabla
cargo	VARCHAR	Cargo que ocupa
idcargo	SMALLINT	Identificador del cargo
entidad	VARCHAR	Entidad a la que pertenece esa persona

3.6.9 Tabla dat_personaeval.

Nombre: dat_personaeval		
Descripción: Se almacenan los datos de la evaluación de la persona		
Atributo	Tipo	Descripción
idpersona	VARCHAR	Identificador de la persona
ideval	INTEGER	Identificador de la tabla
fechaini	VARCHAR	Fecha de inicio
fechafin	VARCHAR	Fecha de fin
conclusiones	VARCHAR	conclusiones
aprobado	BIT	Aprobado

3.6.10 Tabla dat_personafamilia.

Nombre: dat_personafamilia		
Descripción: Se almacenan los datos de la persona		
Atributo	Tipo	Descripción
idpersona	VARCHAR	Identificador de la persona
idfamilia	VARCHAR	Identificador de la familia
idparentesco	VARCHAR	Identificador del parentesco

3.6.11 Tabla dat_personaidioma.

Nombre: dat_personaidioma		
Descripción: Se almacenan los datos de la persona		
Atributo	Tipo	Descripción
idpersona	VARCHAR	Identificador de la persona
idioma	SMALLINT	Identificador del idioma

3.6.12 Tabla dat_personamedio.

Nombre: dat_personamedio		
Descripción: Se almacenan los datos de la persona		
Atributo	Tipo	Descripción
idpersona	VARCHAR	Identificador de la persona
idpersonamedio	SMALLINT	Identificador de la tabla
idasignacion	SMALLINT	Identificador de la asignacion
idmedio	SMALLINT	Identificador del medio
licenciamatricula	VARCHAR	Matricula del medio

3.6.13 Tabla dat_personamilitancia.

Nombre: dat_personamilitancia		
Descripción: Se almacenan los datos de la militancia de la persona		
Atributo	Tipo	Descripción
idpersona	VARCHAR	Identificador de la persona
idmilitancia	SMALLINT	Identificador de la militancia
fechaingreso	VARCHAR	Fecha de ingreso

3.6.14 Tabla dat_personapadecimiento.

Nombre: dat_personapadecimiento		
Descripción: Se almacenan los padecimientos de la persona		
Atributo	Tipo	Descripción
idpersona	VARCHAR	Identificador de la persona
idpadecimiento	SMALLINT	Identificador del padecimiento
tratamiento	VARCHAR	Tratamiento

3.6.15 Tabla dat_personasalud.

Nombre: dat_personasalud		
Descripción: Se almacenan los datos de la persona		
Atributo	Tipo	Descripción
idpersona	VARCHAR	Identificador de la persona
estsalud	CHAR	Estado de salud
limitaciones	TEXT	limitaciones
fecha	VARCHAR	fecha

3.6.16 Tabla dat_personasancion.

Nombre: dat_personasancion		
Descripción: Se almacenan las Sanciones de la persona		
Atributo	Tipo	Descripción
idpersona	VARCHAR	Identificador de la persona
idpersonasancion	SMALLINT	Identificador de la tabla
fecha	VARCHAR	fecha
motivo	VARCHAR	Motivo de la sanción
idsancion	SMALLINT	Identificador de la sanción

3.6.17 Tabla dat_personatrayectorialaboral.

Nombre: dat_personatrayectorialaboral		
Descripción: Se almacenan los datos de la trayectoria laboral de la persona		
Atributo	Tipo	Descripción
idpersona	VARCHAR	Identificador de la persona
idpersonatrayectorialaboral	SMALLINT	Identificador de la tabla
idcausamovimiento	SMALLINT	Identificador de la causa de movimiento
salario	VARCHAR	Salario
desde	VARCHAR	Fecha en que comenzó
Hasta	VARCHAR	Fecha donde termino
entidad	VARCHAR	Entidad a la que pertenece esa persona
cargo	VARCHAR	Cargo que Ocupa
idcargo	SMALLINT	Identificador del cargo
cuadro	BIT	Si es Cuadro

3.6.18 Tabla dat_personatrayecto.

Nombre: dat_personatrayecto		
Descripción: Este campo es para almacenar la trayectoria militar de una persona		
Atributo	Tipo	Descripción
idpersona	VARCHAR	Identificador de persona
idpersonatrayectoriamilitar	SMALLINT	Identificador de persona con trayectoria militar

desde	VARCHAR	Esto es fecha de inicio
hasta	VARCHAR	Esto es fecha de final
identidad	BIGINT	Identificador de entidad
idcausabaja	SMALLINT	Identificador de causa para dar baja
entidad	VARCHAR	Entidad a la que pertenece esa persona
idgrado	SMALLINT	Identificador de grado

3.6.18 Tabla dat_ personatrayecto.

Nombre: dat_reserva		
Descripción: Este campo es para almacenar si una persona es reserva		
Atributo	Tipo	Descripción
idreserva	SMALLINT	Identificador de reserva
idpersona	VARCHAR	Identificador de persona
idcargo	SMALLINT	Identificador de cargo
cargo	VARCHAR	Cargo que ocupa
entidad	VARCHAR	Entidad a la que pertenece esa persona
desde	VARCHAR	Esto es fecha de inicio
hasta	VARCHAR	Esto es fecha de final

3.6.19 Tabla dat_ familiaingresosextras.

Nombre: dat_familiaingresosextras		
Descripción: Se almacenan los datos de los familiares con ingresos monetarios extras		
Atributo	Tipo	Descripción
idfamiliaingreso	INTEGER	Identificador de la tabla
idfamilia	VARCHAR	Identificador de los familiares
idmotivo	SMALLINT	Motivo por el cual lo recibe
cuantia	VARCHAR	Primer apellido de la persona

3.6.20 Tabla dat_ trayectoriapolitica.

Nombre: dat_trayectoriapolitica		
Descripción: Este campo es para almacenar si una persona es reserva		
Atributo	Tipo	Descripción
idpersona	VARCHAR	Identificador de persona
idtrayectirapolitica	SMALLINT	Identificador de trayectoria política

denomcargo	VARCHAR	Denominación de cargo
idcargo	SMALLINT	Identificador de cargo
idmcpio	SMALLINT	Identificador de municipio
interes	SMALLINT	Datos de interes de la persona
idnivel	SMALLINT	Identificador de nivel
fechaini	VARCHAR	Esto es fecha de inicio
fechafin	VARCHAR	Esto es fecha de final
propuesto	BIT	Propuesta de cargo de la persona
denommilitancia	VARCHAR	Denominación militar
cuadro	BIT	cuadro

3.6.21 Tabla his_direccion.

Nombre: his_direccion		
Descripción: Tabla histórica para acumular la dirección de la persona		
Atributo	Tipo	Descripción
idpersona	VARCHAR	Identificador de la persona
iddireccion	INTEGER	Identificador de la tabla
direccion	VARCHAR	Dirección
desde	VARCHAR	Fecha en que comenzó
Hasta	VARCHAR	Fecha donde termino
iddpa	BIGINT	identificador

3.6.22 Tabla nom_asignación.

Nombre: nom_asignación		
Descripción: Este campo es para almacenar una asignación		
Atributo	Tipo	Descripción
idasignación	SMALLINT	Identificador de asignación
denom	VARCHAR	Valores que puede tomar el nomenclador
actual	BIT	Esto es para ver si esta activado

3.6.23 Tabla nom_asignación.

Nombre: nom_categoriadocente		
Descripción: Este campo es para almacenar una categoría docente		
Atributo	Tipo	Descripción
idcatdocente	SMALLINT	Identificador de categoría
denom	VARCHAR	Valores que puede tomar el nomenclador
actual	BIT	Esto es para ver si esta activado

3.6.24 Tabla nom_causabajamil.

Nombre: nom_causabajamil		
Descripción: Este campo es para almacenar la causa de la baja		
Atributo	Tipo	Descripción
idcausabaja	SMALLINT	Identificador de la causa de baja
denom	VARCHAR	Valores que puede tomar el nomenclador
actual	BIT	Esto es para ver si esta activado

3.6.25 Tabla nom_causamovimiento.

Nombre: nom_causamovimiento		
Descripción: Este campo es para almacenar la causa de movimiento de la persona		
Atributo	Tipo	Descripción
idcausamovimiento	SMALLINT	Identificador de la causa de movimiento
denom	VARCHAR	Valores que puede tomar el nomenclador
actual	BIT	Esto es para ver si esta activado

3.6.26 Tabla nom_colorajo.

Nombre: nom_colorajo		
Descripción: Este campo es para almacenar el color de los ojos de la persona		
Atributo	Tipo	Descripción
idcolorajos	SMALLINT	Identificador del color de los ojos
denom	VARCHAR	Valores que puede tomar el nomenclador
actual	BIT	Esto es para ver si esta activado

3.6.27 Tabla nom_colorpelo.

Nombre: nom_colorpelo		
Descripción: Este campo es para almacenar el color del pelo de la persona		
Atributo	Tipo	Descripción
idcolorpelo	SMALLINT	Identificador del color del pelo
denom	VARCHAR	Valores que puede tomar el nomenclador
actual	BIT	Esto es para ver si esta activado

3.6.28 Tabla nom_colorpiel.

Nombre: nom_colorpiel		
Descripción: Este campo es para almacenar el color de la piel de la persona		
Atributo	Tipo	Descripción
idcolorpiel	SMALLINT	Identificador del color de la piel
denom	VARCHAR	Valores que puede tomar el nomenclador
actual	BIT	Esto es para ver si esta activado

3.6.29 Tabla nom_condecoración.

Nombre: nom_condecoración		
Descripción: Este campo es para almacenar la condecoración de la persona		
Atributo	Tipo	Descripción
idcondecoración	SMALLINT	Identificador de la condecoración
denom	VARCHAR	Valores que puede tomar el nomenclador
actual	BIT	Esto es para ver si esta activado

3.6.30 Tabla nom_cursoesp.

Nombre: nom_cursoesp		
Descripción: Este campo es para almacenar los cursos especiales de la persona		
Atributo	Tipo	Descripción
idcurso	SMALLINT	Identificador del curso especial
denom	VARCHAR	Valores que puede tomar el nomenclador
actual	BIT	Esto es para ver si esta activado

3.6.31 Tabla nom_cursomil.

Nombre: nom_cursomil		
Descripción: Este campo es para almacenar los cursos militares de la persona		
Atributo	Tipo	Descripción
idcurso	SMALLINT	Identificador del curso militares
denom	VARCHAR	Valores que puede tomar el nomenclador
actual	BIT	Esto es para ver si esta activado

3.6.32 Tabla nom_cursopol.

Nombre: nom_cursopol		
Descripción: Este campo es para almacenar los cursos político de la persona		
Atributo	Tipo	Descripción
idcurso	SMALLINT	Identificador del curso político
denom	VARCHAR	Valores que puede tomar el nomenclador
actual	BIT	Esto es para ver si esta activado

3.6.33 Tabla nom_decision.

Nombre: nom_decision		
Descripción: Este campo es para almacenar las decisiones de la persona		
Atributo	Tipo	Descripción
iddecisión	SMALLINT	Identificador de decisión
denom	VARCHAR	Valores que puede tomar el nomenclador
actual	BIT	Esto es para ver si esta activado

3.6.34 Tabla nom_especialidad.

Nombre: nom_especialidad		
Descripción: Este campo es para almacenar la especialidad de la persona		
Atributo	Tipo	Descripción
idespecialidad	SMALLINT	Identificador de la especialidad
denom	VARCHAR	Valores que puede tomar el nomenclador
actual	BIT	Esto es para ver si esta activado

3.6.35 Tabla nom_estadocivil.

Nombre: nom_estadocivil		
Descripción: Este campo es para almacenar el estado civil de la persona		
Atributo	Tipo	Descripción
idestadocivil	SMALLINT	Identificador del estado civil
denom	VARCHAR	Valores que puede tomar el nomenclador
actual	BIT	Esto es para ver si esta activado

3.6.36 Tabla nom_extrasocial.

Nombre: nom_extrasocial		
Descripción: Este campo es para almacenar el extra social de la persona		
Atributo	Tipo	Descripción
idextrasocial	SMALLINT	Identificador del extra social
denom	VARCHAR	Valores que puede tomar el nomenclador
actual	BIT	Esto es para ver si esta activado

3.6.37 Tabla nom_gradocientifico.

Nombre: nom_gradocientifico		
Descripción: Este campo es para almacenar el grado científico de la persona		
Atributo	Tipo	Descripción
idgradocientifico	SMALLINT	Identificador del grado científico
denom	VARCHAR	Valores que puede tomar el nomenclador
actual	BIT	Esto es para ver si esta activado

3.6.38 Tabla nom_grupoescolar.

Nombre: nom_grupoescolar		
Descripción: Este campo es para almacenar el grupo escolar de la persona		
Atributo	Tipo	Descripción
idgrupoescolar	SMALLINT	Identificador del grupo escolar
denom	VARCHAR	Valores que puede tomar el nomenclador
actual	BIT	Esto es para ver si esta activado

3.6.39 Tabla nom_gruposanguineo.

Nombre: nom_gruposanguineo		
Descripción: Este campo es para almacenar el grupo sanguineo de la persona		
Atributo	Tipo	Descripción
idgruposanguineo	SMALLINT	Identificador del gruposanguineo
denom	VARCHAR	Valores que puede tomar el nomenclador
actual	BIT	Esto es para ver si esta activado

3.6.40 Tabla nom_idioma.

Nombre: nom_idioma		
Descripción: Este campo es para almacenar el idioma de la persona		
Atributo	Tipo	Descripción
ididioma	SMALLINT	Identificador del idioma
denom	VARCHAR	Valores que puede tomar el nomenclador
actual	BIT	Esto es para ver si esta activado

3.6.41 Tabla nom_marca.

Nombre: nom_marca		
Descripción: Este campo es para almacenar la marca de la persona		
Atributo	Tipo	Descripción
idmarca	SMALLINT	Identificador del marca
denom	VARCHAR	Valores que puede tomar el nomenclador
actual	BIT	Esto es para ver si esta activado
idtipomedio	SMALLINT	Identificador del tipo de medio

3.6.42 Tabla nom_mcpio.

Nombre: nom_mcpio		
Descripción: Este campo es para almacenar los municipios de la persona		
Atributo	Tipo	Descripción
idmcpio	SMALLINT	Identificador de los municipios
idprov	SMALLINT	Identificador de las provincias
denom	VARCHAR	Valores que puede tomar el nomenclador
actual	BIT	Esto es para ver si esta activado

3.6.43 Tabla nom_medio.

Nombre: nom_medio		
Descripción: Este campo es para almacenar los medio de la persona		
Atributo	Tipo	Descripción
idmedio	SMALLINT	Identificador de medio
denomodelo	VARCHAR	Denominación del modelo
idmarca	SMALLINT	Identificador de marca
actual	BIT	Esto es para ver si esta activado

3.6.44 Tabla nom_militancia.

Nombre: nom_militancia		
Descripción: Este campo es para almacenar la militancia de la persona		
Atributo	Tipo	Descripción
idmilitancia	SMALLINT	Identificador de militancia
denomlarga	VARCHAR	Denominación larga
denomcorta	VARCHAR	Denominación corta
actual	BIT	Esto es para ver si esta activado

3.6.45 Tabla nom_motivo.

Nombre: nom_motivo		
Descripción: Este campo es para almacenar los motivo de la persona		
Atributo	Tipo	Descripción
idmotivo	SMALLINT	Identificador de motivo
denom	VARCHAR	Valores que puede tomar el nomenclador
actual	BIT	Esto es para ver si esta activado

3.6.46 Tabla nom_motivoingreso.

Nombre: nom_motivoingreso		
Descripción: Este campo es para almacenar los motivo de ingreso de la persona		
Atributo	Tipo	Descripción
idmotivoingreso	SMALLINT	Identificador de motivo de ingreso
denom	VARCHAR	Valores que puede tomar el nomenclador
actual	BIT	Esto es para ver si esta activado

3.6.47 Tabla nom_motivorelacion.

Nombre: nom_motivorelacion		
Descripción: Este campo es para almacenar los motivo de relación de la persona		
Atributo	Tipo	Descripción
idmotivorelacion	SMALLINT	Identificador de motivo de relación
denom	VARCHAR	Valores que puede tomar el nomenclador
actual	BIT	Esto es para ver si esta activado

3.6.48 Tabla nom_motivosancion.

Nombre: nom_motivosancion		
Descripción: Este campo es para almacenar los motivo de sanción de la persona		
Atributo	Tipo	Descripción
idmotivosancion	SMALLINT	Identificador de motivo de sanción
denom	VARCHAR	Valores que puede tomar el nomenclador
actual	BIT	Esto es para ver si esta activado

3.6.49 Tabla nom_nivel.

Nombre: nom_nivel		
Descripción: Este campo es para almacenar el nivel de la persona		
Atributo	Tipo	Descripción
idnivel	SMALLINT	Identificador de nivel
denom	VARCHAR	Valores que puede tomar el nomenclador
actual	BIT	Esto es para ver si esta activado

3.6.50 Tabla nom_nivelescolar.

Nombre: nom_nivelescolar		
Descripción: Este campo es para almacenar el nivel escolar de la persona		
Atributo	Tipo	Descripción
idnivelescolar	SMALLINT	Identificador de nivel escolar
actual	BIT	Esto es para ver si esta activado
denom	VARCHAR	Valores que puede tomar el nomenclador
idgruposcolar	SMALLINT	Identificador de grupo escolar

3.6.51 Tabla nom_ padecimiento.

Nombre: nom_padecimiento		
Descripción: Este campo es para almacenar el padecimiento de la persona		
Atributo	Tipo	Descripción
idpadecimiento	SMALLINT	Identificador de padecimiento
denom	VARCHAR	Valores que puede tomar el nomenclador
actual	BIT	Esto es para ver si esta activado

3.6.52 Tabla nom_ pais.

Nombre: nom_pais		
Descripción: campo para almacenar el país de la persona		
Atributo	Tipo	Descripción
idpais	SMALLINT	Identificador de país
denom	VARCHAR	Valores que puede tomar el nomenclador
actual	BIT	Esto es para ver si esta activado

3.6.53 Tabla nom_ parentesco.

Nombre: nom_parentesco		
Descripción: Este campo es para almacenar el parentesco de la persona		
Atributo	Tipo	Descripción
idparentesco	SMALLINT	Identificador de parentesco
denom	VARCHAR	Valores que puede tomar el nomenclador
actual	BIT	Esto es para ver si esta activado

3.6.54 Tabla nom_ provincia.

Nombre: nom_provincia		
Descripción: Este campo es para almacenar la provincia de la persona		
Atributo	Tipo	Descripción
idprov	SMALLINT	Identificador de provincia
denom	VARCHAR	Valores que puede tomar el nomenclador
actual	BIT	Esto es para ver si esta activado

3.6.55 Tabla nom_ tipoeval.

Nombre: nom_ tipoeval		
Descripción: Este campo es para almacenar la evaluación de la persona		
Atributo	Tipo	Descripción
idtipoeval	SMALLINT	Identificador de evaluación
denom	VARCHAR	Valores que puede tomar el nomenclador
titulo	VARCHAR	Título de la evaluación
actual	BIT	Esto es para ver si esta activado

3.6.56 Tabla nom_ sancion.

Nombre: nom_ sancion		
Descripción: Este campo es para almacenar la sanción de la persona		
Atributo	Tipo	Descripción
idsancion	SMALLINT	Identificador de sanción
denom	VARCHAR	Valores que puede tomar el nomenclador
actual	BIT	Esto es para ver si esta activado
idtiposancion	SMALLINT	Identificador de tipo de sanción

3.6.57 Tabla nom_ tipocargo.

Nombre: nom_ tipocargo		
Descripción: Este campo es para almacenar el tipo de cargo de la persona		
Atributo	Tipo	Descripción
idtipocargo	SMALLINT	Identificador de tipo de cargo
denom	VARCHAR	Valores que puede tomar el nomenclador
actual	BIT	Esto es para ver si esta activado

3.6.58 Tabla nom_ tipocondecoracion.

Nombre: nom_ tipocondecoracion		
Descripción: Este campo es para almacenar el tipo de condecoración de la persona		
Atributo	Tipo	Descripción
idtipocondecoracion	SMALLINT	Identificador de tipo de condecoración
denom	VARCHAR	Valores que puede tomar el nomenclador
actual	BIT	Esto es para ver si esta activado

3.6.59 Tabla nom_salida.

Nombre: nom_salida		
Descripción: Este campo es para almacenar las salidas de la persona		
Atributo	Tipo	Descripción
idsalida	SMALLINT	Identificador de salida
denom	VARCHAR	Valores que puede tomar el nomenclador
actual	BIT	Esto es para ver si esta activado

3.6.60 Tabla nom_tipomedio.

Nombre: nom_tipomedio		
Descripción: Este campo es para almacenar el tipo de medio de la persona		
Atributo	Tipo	Descripción
idtipomedio	SMALLINT	Identificador de tipo de medio
denom	VARCHAR	Valores que puede tomar el nomenclador
actual	BIT	Esto es para ver si esta activado

3.6.61 Tabla nom_tiposancion.

Nombre: nom_tiposancion		
Descripción: Este campo es para almacenar el tipo de sanción de la persona		
Atributo	Tipo	Descripción
idtiposancion	SMALLINT	Identificador de tipo de sanción
denom	VARCHAR	Valores que puede tomar el nomenclador
actual	BIT	Esto es para ver si esta activado

3.7 Principios de diseño.

El diseño del sistema es la parte del mismo con que el usuario interactúa directamente; por lo que se consideró que es una parte fundamental dentro del proceso de desarrollo.

De modo general los usuarios solo están interesados en cómo se le muestra la interfaz de la aplicación y en cómo usarla, o sea que entre los mismos en el uso de las nuevas tecnologías prevalece un nivel medio; aunque en cuanto a la experiencia y conocimiento del proceso que se automatiza la capacidad de los estos es muy amplia y su interés en la aplicación informática es evaluada de positiva; por lo que fueron planteadas las siguientes características para el diseño de nuestro sistema:

- Los elementos mostrados en la pantalla no serán numerosos y en caso de repetirse alguno en otras interfaces, estos serán mostrados en la misma posición en todas para familiarizar al usuario con el sistema.
- Las páginas que muestran información, lo harán en el mismo orden y de manera similar.
- Cada elemento se diseñará siguiendo un patrón de tamaño, formas y colores (no serán ni fuertes ni brillantes).

Tratamiento de errores.

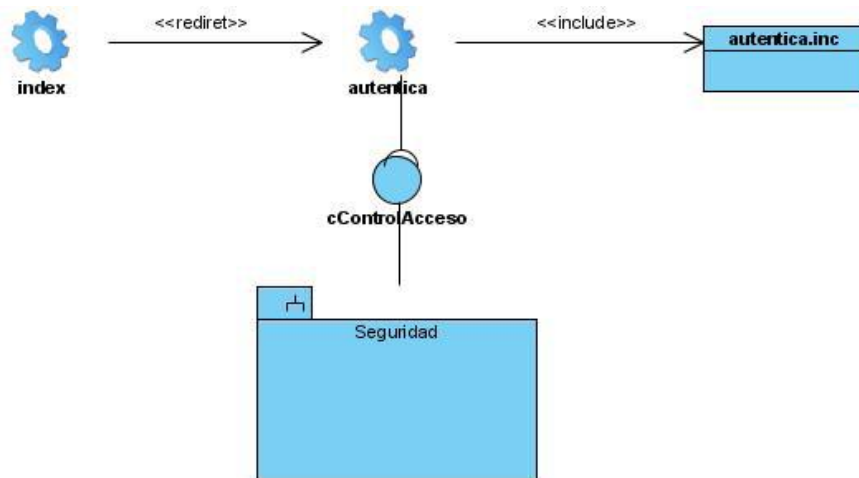
Identificar y controlar los posibles errores que se pueden presentar a la hora de interactuar con el software es de vital importancia para así garantizar un correcto funcionamiento del sistema; de manera general para el tratamiento de errores serán generados en todos los casos por funciones JavaScript para evitar la ejecución de procesos en la página cuando existan errores; además se incorpora la clase jsDOM que se encarga de la validación de la entrada de los datos en los formularios, así que todos los datos de entrada serán validados. Se mostraran mensajes de confirmación en el caso de las modificaciones y eliminaciones de manera que no se realice una operación no deseada. Se tratará de que el usuario introduzca la menor cantidad de datos ayudándolo mediante la utilización de componentes de para evitar incoherencias e incorrecciones en los mismos, Un segundo momento para manejar el control de errores se maneja en la capa de acceso a datos a través de las clases típicas. Estas implementan validaciones sobre los principales métodos (inserción, modificación y eliminación), que se llevan a cabo sobre cualquier entidad.

Interfaz de usuario.

Para garantizar que el usuario se adapte con facilidad al sistema que se les presenta; o sea que la interfaz que le proponemos le sea amigable, sencilla, agradable y fácil de usar fueron consideradas una serie de aspectos en la conformación de la misma. En todas las páginas principales se establecerá el mismo patrón de diseño, la barra de menú con sus opciones se encontrará en la parte superior de la aplicación, será horizontal, no deberá exceder los 2 niveles de profundidad y la denominación de las opciones principales deben estar visibles. Contará con un banner de presentación en la cabecera no muy grande que identificará a la aplicación. Se trabajará con las familias de fuentes: Verdana, Arial, Helvetica, Sans-serif; el tamaño de la fuente no deberá diferir mucho de 16 px y los colores los serán sobre tonalidades claras basados fundamentalmente en azul claro y blanco.

Seguridad

El diseño del sistema de Registro y control de cuadros cuenta con una serie de subsistema, dentro del mismo encontramos el de Seguridad; aprovechando la interfaz cControlacceso que el subsistema aporta para acceder al servicio Web que este brinda para la autenticación de los usuarios y apoyándose en su método público logueo se propone como solución para la implementación de la seguridad el uso de un servicio Web encargado del control de los accesos, autenticación y registro de los eventos que ocurren en el mismo.



Concepción de la ayuda.

La aplicación cuenta con un manual avanzado de uso para el usuario. Un material complementario que se pone a disposición de los usuarios para el correcto manejo de la misma, el mismo permite el dominio eficiente de cada una de las funcionalidades que brinda la aplicación y así un correcto aprovechamiento de sus potencialidades.

3.8 Conclusiones.

Los artefactos desarrollados en este flujo de trabajo fueron el diagrama de despliegue, el modelo de implementación, interfaz, subsistema de implementación y diagrama de componentes. Además de una descripción de la arquitectura utilizada.

CONCLUSIONES

Luego del estudio realizado sobre los resultados obtenidos en el análisis previo de esta investigación y la generación de los artefactos relacionados con el Diseño, se permitirá a los implementadores programar un sistema que dé como resultado realizar una mejor gestión del Registro y Control de Cuadros, permitiendo así que las actualizaciones se hagan en tiempo real con una mayor rapidez y eficiencia.

Se realizó una investigación sobre los diferentes patrones de diseño y arquitectura que intervendrán en el sistema; así como se delineó un conjunto de herramientas a utilizar. Se realizaron un conjunto de descripciones de las clases del Diseño, que se ajustan a los principios y patrones del diseño orientado a objetos

Las metas fueron alcanzadas y se logró el objetivo fundamental de la investigación: El análisis, y diseño de una aplicación Web que contribuya al desarrollo del registro y control de cuadros de una forma más eficiente y ayude a facilitar la gestión de la información.

RECOMENDACIONES

- Mejorar y/o enriquecer así como continuar el trabajo con nuevas propuestas que puedan surgir posteriormente.
- Continuar trabajando sobre la base de lo diseñado para finalizar así todas las funcionalidades que son necesarias para un desarrollo óptimo de la aplicación.
- Continuar la implementación y desarrollo del subsistema, dando continuidad al ciclo de desarrollo próximo, el cual contendrá otras funcionalidades.
- Continuar con la investigación para garantizar nuevas mejoras en futuras versiones del sistema.

BIBLIOGRAFÍA

[AUD 1991] – Andreu et Alemania.1991.

[LAU 2002] - Kenneth C.Laudon: New York University - Jane P.Laudon – “Information Systems (Prentice Hall)”, 2002.

[MÉNDEZ 2002] - Méndez, Leonardo. Sistemas de Información y Sistemas Administrativos, 2002.

- Jim Conallen, *Building Web Applications with UML*, Object Technology Series by Addison Wesley Longman, 1999
- Jim Conallen, *UML Extension for Web Applications 0.91*, 22-Mar-1999.
- Jim Conallen, *Modeling Web Application Architectures with UML*, Rational Software, June 1999
http://www.rational.com/media/uml/resources/documentation/27662_webapps.pdf
- Stan Ward, Per Kroll, *Building Web Solutions with the Rational Unified Process: Unifying the Creative Design Process and the Software Engineering Process*.
- JACOBSON, Ivar; RUMBAUGH, James; BOOCH, Grady, “El proceso unificado de desarrollo”.2000. Addison Wesley. Capítulos 9 Páginas 205-254.
- RUMBAUGH, James, JACOBSON, Ivar; BOOCH, Grady, “El lenguaje unificado de modelado. Manual de referencia”.2000. Addison Wesley. Capítulos 8 y 13 Páginas 75-80, 214, 216-218, 175-182 y 330.
- LARMAN, Craig “UML y patrones” 1999, Prentice Hall Iberoamericana. Capítulos 13, 16, 17, 18,19, 21, 34 y 35.
- Bruegge, B. Y Dutoit, A. “Ingeniería de Software Orientado a Objetos”. 2002. Prentice Hall – Pearson Educación. Capítulos 5 y 6. Páginas 146-149, 167-229.
- GAMMA, E.; HELM, R.; JOHNSON, R. y VLISSIDES, J. “Patrones de diseño”.2000.
<http://www.vico.org/pages/PatronsDisseny.html>.
- UML y la Modelación de datos.pdf. Whitepaper de Rational Rose.
- Rational Unified Process (de la Suite de Rational 2003).

GLOSARIO

CASE: Acrónimo inglés de Computer Aided Software Engineering, que viene a significar Ingeniería de Software Asistida por Ordenador.

Hipertexto: Hipertexto son datos que contienen enlaces (links) a otros datos. En el lenguaje Web, un documento de hipertexto no es solo algo que contiene datos, sino que además contiene enlaces a otros documentos. En Hipertexto se pueden hacer enlaces en cualquier lugar, no sólo al final. Cada enlace tiene una marca que lo destaca, puede estar resaltado, subrayado o puede estar identificado por un número. El hipertexto no está limitado a datos textuales, podemos encontrar dibujos del elemento especificado, sonido o vídeo referido al tema. Estos documentos que tienen gran variedad de datos, como sonido, vídeo, texto. El hipertexto es una herramienta potente para aprender y explicar. El texto debe ser diseñado para ser explorado libremente y así se consigue una comunicación de ideas más eficientes.

HTML: Acrónimo inglés de Hyper Text Markup Language (lenguaje de marcación de hipertexto), es un lenguaje de marcas diseñado para estructurar textos y presentarlos en forma de hipertexto, que es el formato estándar de las páginas web. Este lenguaje se basa en tags (instrucciones que le dicen al texto como deben mostrarse) y atributos (parámetros que dan valor al tag). Es el estándar usado en el World Wide Web.

HTTP: Es el protocolo de la Web (WWW), usado en cada transacción. Las letras significan Hyper Text Transfer Protocol, es decir, protocolo de transferencia de hipertexto. El hipertexto es el contenido de las páginas web, y el protocolo de transferencia es el sistema mediante el cual se envían las peticiones de acceder a una página web, y la respuesta de esa web, remitiendo la información que se verá en pantalla.

HTTPS: Es un protocolo de red basado en el protocolo HTTP, destinado a la transferencia segura de datos de hipertexto, es decir, es la versión segura de HTTP.

IDE: Siglas de: Integrated Drive Electronics ó Integrated development environment, es decir, un entorno integrado de desarrollo.

Internet: Es una red de redes a escala mundial de millones de computadoras interconectadas con el conjunto de protocolos TCP/IP. También se usa este nombre como sustantivo común y por tanto en minúsculas para designar a cualquier red de redes que use las mismas tecnologías que la Internet, independientemente de su extensión o de que sea pública o privada.

Intranet: Una intranet es una red de ordenadores de una red de área local.

Lenguajes Interpretados: Los lenguajes de programación, que incluye a aquellos Lenguajes cuyos programas son habitualmente lenguajes interpretados (o lenguajes de Script) forman un subconjunto de los ejecutados en un intérprete en vez de compilados. Sin embargo, la definición de un lenguaje de programación es independiente de cómo se ejecuten los programas en él escritos, ya sea mediante una compilación previa o a través de un intérprete.

PDF: Es la Extensión que corresponde a un tipo de fichero (un libro electrónico) creado con Adobe Acrobat.

RUP: El Proceso Racional Unificado o RUP (Rational Unified Process), es un proceso de desarrollo de software.

UML: Lenguaje Unificado de Modelado (UML, por sus siglas en inglés, Unified Modelling Language) es el lenguaje de modelado de sistemas de software más conocido en la actualidad; aún cuando todavía no es un estándar oficial, está apoyado en gran manera por la OMG (Grupo dedicado a la promoción de la tecnología orientada a objetos y su estandarización).