

**Universidad de las Ciencias Informáticas
Facultad 7**



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

Título: Sistema de Gestión de Información en el
Proceso de Formación Posgraduada en Salud.
Módulo de Especialidades.

Autores: Maryelines Labrada Madrigal
Gabriel Armenteros Travieso

Tutor: Ing. Reinier Alonso González
Asesora: Lic. Yaima Margarita Riverí Ruiz

Ciudad de La Habana, Julio de 2008.

“Año 50 de la Revolución.”

DECLARACIÓN DE AUTORÍA

Declaramos ser autores de la presente tesis y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmamos la presente a los 11 días del mes de julio del año 2008.

Maryelines Labrada Madrigal

Autora

Gabriel Armenteros Travieso

Autor

Ing. Reinier Alonso González

Tutor

DATOS DE CONTACTO

Ing. Reinier Alonso González. (ralonso@uci.cu). Graduado de Ingeniero en Ciencias Informáticas en la Universidad de las Ciencias Informáticas en su primera graduación. Actualmente profesor en adiestramiento en el departamento de Técnicas de Programación de la Facultad 7 y Jefe del Polo de desarrollo para la Salud.

Lic. Yaima Margarita Riveri Ruiz (maggie@uci.cu): Licenciada en Educación. Especialidad Lengua Inglesa. Profesor Instructor. En cinco años de labor docente, ha impartido las asignaturas de la disciplina Idiomas Extranjeros. Es Jefa de la asignatura Idioma Extranjero IV.

“En Cuba, nadie ha hecho tanto en tan poco tiempo”

Fidel Castro

AGRADECIMIENTOS

En primer lugar le estamos muy agradecidos a nuestro Comandante en Jefe Fidel Castro, por habernos dado la posibilidad de estudiar en este centro de altos estudios.

A nuestro oponente Karel Gómez y a nuestra asesora Yaima Riverí por su labor abnegada y su ayuda incondicional.

A nuestro tutor Reinier Alonso y a la profesora Yenisel Valido por su apoyo y dedicación.

A todos los que de una forma u otra fueron partícipes de nuestra formación profesional.

De Maryelínes:

Le agradezco a mi familia por haber depositado confianza en mí. Especialmente a la persona que más amo: mi madre Ledis, que es mi razón de ser.

A mi padrastro Alberto que ha dedicado una gran parte de su vida a darme amor y cariño, apoyándome incondicionalmente en cada paso que di.

A mis hermanos que a pesar de ser pequeños estuvieron preocupados por mí. A mis tías, especialmente a Kenia y Zenia por su preocupación y por darme valor para seguir adelante en cualquier circunstancia. A mis abuelos, primos, primas y vecinos por su preocupación y apoyo espiritual.

A aquellas personas que me han brindado su compañía y amistad durante todas las etapas de estudios, especialmente a Yisel que ha sido como una hermana, apoyándome en los buenos y malos momentos. A mi compañero de tesis Gabriel que supo sobrellevar mis etapas de desesperación. A mis compañeros de aula, profesores, A todos ellos muchas gracias.

De Gabriel

Les agradezco con todas las fuerzas de mi corazón a mis padres y mi hermano Reinier, que han sido mi fuente de inspiración y siempre me han dado su apoyo incondicional para seguir adelante.

A mis abuelas Paula y Ohilda que las quiero mucho y sé que ellas a mi también, a mi primo Yoanis quien es como un hermano para mí. A toda mi familia en general, la del Mariel y la del Cerro que siempre han estado presente.

A todos mis amigos: los de la Humboldt: Adonis, Alioth, Carlos, Osmani, Pedro, Sergio y Ariel y a los del Henequén Sulien, Dayron, Osmel (el pina) y Brady que siempre se han preocupado por mis estudios.

Agradezco también a Raciél, Pavel, Yaimi y a todos mis compañeros de aula que me brindaron su amistad durante estos 5 años. Siempre los tendré presente. A mi compañera de tesis Maryi que sin ella nada hubiera sido posible.

Muchas gracias...los quiero.

DEDICATORIA

*A mis queridísimos padres, Ledis y Alberto,
por el beso y el regaño oportuno.
Con todo el amor.
Maryi.*

*A mis padres que son las personas que más quiero en la vida,
por haber puesto todo su empeño en mí.
Los sueños sí se hacen realidad.
Los quiere mucho.
Gabriel*

RESUMEN

En el presente trabajo se realiza una investigación sobre la **Gestión de Información en el Proceso de Formación Posgraduada** que se lleva a cabo en las Secretarías Docentes de las Facultades o Institutos Médicos. Esta información es almacenada en una base de datos central que recoge los datos concernientes a los programas de Capacitación, Superación, Doctorado, Maestría y Especialidades.

Por ello este trabajo tiene como objetivo informatizar la actividad de posgrado para área de las Especialidades. Para garantizar la actualización, procesamiento y obtención de información. Con propósito de lograr rapidez, confiabilidad y calidad en la transmisión de la información. La aplicación se desarrolla sobre tecnologías web, corriendo sobre un servidor LAMP (*Linux, Apache, MySQL, PHP*), utilizando herramientas de software libre.

Con el desarrollo propuesto para el módulo Especialidades se proporciona una interfaz amigable, con flexibilidad y claridad para la navegación. Además se logra rapidez y efectividad en el proceso de gestión de la información para la formación posgraduada. Se facilita el manejo de los datos del programa en dependencia de la solicitud del usuario.

PALABRAS CLAVE: *Gestión, Posgrado, Especialidades, Residencia, Módulo.*

TABLA DE CONTENIDOS

INTRODUCCIÓN.....	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA.....	6
1.1 Sistema Nacional de Salud	6
1.2 Surgimiento de la educación de posgrado.....	7
1.3 Comienzo de las actividades del programa de las Especialidades en Cuba.....	8
1.4 Tendencias y tecnologías actuales.....	9
1.5 Leguajes de programación utilizados	16
1.6 Sistema Gestor de base de datos utilizado (SGBD)	17
1.7 Servidor Web Apache 2.2	18
1.8 Metodología utilizada para el desarrollo del software	18
1.9 Herramientas utilizadas.....	20
CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA	25
2.1 Objeto de estudio	25
2.2 Modelo de negocio.....	25
2.3 Descripción de los procesos del negocio.....	28
2.4 Propuesta del sistema.....	29
2.5 Especificación de los Requerimientos de Software	30
2.6 Definición de los casos de uso del sistema	34
CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA.....	45
3.1 Modelo de Análisis	45
3.2 Modelo de Diseño	51
3.3 Descripción de las clases.....	58
3.4 Diagramas de Interacción	64
3.5 Diseño de la Base de Datos	65
3.6 Definiciones del diseño	73
3.7 Tratamiento de errores.....	74
3.8 Seguridad.....	74
CAPÍTULO 4: IMPLEMENTACIÓN.....	76
4.1 Dependencias y Relaciones con otros Sistemas	76
4.2 Integración del módulo Especialidades al Sistema	78
4.3 Modelo de Implementación	78
CONCLUSIONES.....	87
RECOMENDACIONES.....	88
REFERENCIAS BIBLIOGRÁFICAS.....	89
BIBLIOGRAFÍA.....	92
ANEXO 1: MODELO DE NEGOCIO.....	95
ANEXO 2: MODELO DE DISEÑO.....	103

INTRODUCCIÓN

La informática en sus diferentes manifestaciones tiene asegurado un papel protagónico en el futuro de la sociedad. En Cuba, desde muy temprano se ha identificado la conveniencia y necesidad de dominar e introducir en la práctica social las Tecnologías de la Información y las Comunicaciones (TICs). Lo que permitirá lograr una cultura informática en la sociedad, como una de las características imprescindibles del hombre nuevo.

Durante los últimos 20 años, un grupo de instituciones cubanas han desarrollado sistemas encaminados a lograr determinados niveles de informatización de la salud. Estas soluciones carecían de integración y de una definición generalizable, aparte de que no existían los recursos tecnológicos necesarios para su ejecución en el Sistema Nacional de Salud (SNS). A partir de 1997 se concibe una primera estrategia de informatización como respuesta del sector de la salud a los lineamientos estratégicos para la informatización de la sociedad cubana. Su finalidad consistía en coordinar esfuerzos para el desarrollo de este proceso en el SNS. [1]

La informatización del Sistema Nacional de Salud está dada por el conjunto de métodos, técnicas, procederes y actividades gerenciales dirigidas al manejo de la información en salud. Esta comprende la información sobre el estado de salud de la población, el conocimiento de las ciencias de la salud y la información en general para la toma de decisiones clínico-epidemiológicas, operativas y estratégicas.

La informática se ha convertido en una herramienta de uso cotidiano por el personal de salud en todo el país. Con la creación de INFOMED, servicio de Intranet que comunica a todos los centros de salud, a través de una densa red de computadoras, posibilita la comunicación, consultas e intercambio científico entre todos los médicos, enfermeros y técnicos así como el acceso a todas las bases de datos e información médica.

Con el propósito de lograr una cultura informática, el Ministerio de Salud Pública (MINSAP) se ha visto en la ardua tarea de aplicar la informatización a todas las esferas y procesos. Se prioriza así el área de docencia, para la gestión de la información referente a la formación de recursos humanos tanto de pregrado como de posgrado. Para ello ha convocado un grupo de instituciones propias del sector del Ministerio de Informática y Comunicaciones (MIC), a trabajar

integradamente para definir de conjunto la estrategia a desarrollar. En su desarrollo e implementación participan diferentes instituciones como SOFTEL, la Universidad de Ciencias Informáticas (UCI), INFOMED y las Direcciones Nacionales del Ministerio de Salud Pública implicadas directamente en los primeros productos.

Se trabaja al mismo tiempo en tres direcciones estratégicas: desarrollo de los sistemas y aplicaciones, completamiento y capacitación de los recursos humanos necesarios y organización de las estructuras institucionales encargadas de garantizar el funcionamiento de las aplicaciones. De esta manera el MINSAP asume su proceso en el marco de la informatización de la sociedad cubana para expresar con eficiencia y calidad la atención médica al pueblo. [2]

Este ministerio cuenta con un departamento de posgrado que tiene asociados los programas de Maestrías, Doctorados, Capacitación, Superación Profesional y las Especialidades; esta última constituye el factor fundamental de la investigación en cuestión. Las Especialidades o Residencias Médicas son un programa de la educación de posgrado, que les proporcionan a los profesionales una mayor superación profesional, continuidad de su carrera y mejoramiento en su vínculo laboral.

Teniendo en cuenta que el posgrado constituye el nivel más elevado del Sistema Nacional de Educación, los especialistas que forman parte de estas canteras, lo hacen con el propósito de alcanzar un nivel cultural avanzado. Se contribuye de esta manera a elevar la productividad, eficiencia y calidad del trabajo de los recién egresados universitarios.

Durante el período de adiestramiento laboral la adquisición, actualización, complementación y profundización de los conocimientos y las habilidades específicas son elementos necesarios para asumir con eficiencia las funciones inherentes a los cargos para los que sean destinados en su centro de trabajo. Así mismo se forman los cuadros científicos al más alto nivel de desarrollo de cada rama de actividad, de acuerdo con las necesidades del país.

Las actividades del programa de las especialidades se planifican y ejecutan de modo que se logren los resultados esperados con mínimo gasto de tiempo, movimiento y recursos humanos, materiales y financieros. Para ello se elabora el plan de desarrollo profesional que responde a las necesidades de las instituciones y de cada profesional. No puede ser espontáneo, sino

dirigido a cumplir con la misión, la visión y los objetivos de cada unidad, municipio, provincia y/o el sistema nacional de salud cubano.

Todo centro de educación médica debe satisfacer las necesidades de superación de sus profesionales. Deberá solicitar a las instituciones docentes y asistenciales, a las direcciones municipales y provinciales de Salud y al MINSAP, las principales temáticas para la confección de los programas. Las instituciones y direcciones identifican los problemas de salud y otros que generen necesidad de superación. Para ello es necesario aplicar las técnicas de aprendizaje, expresadas en diferentes temáticas que responden a los programas concebidos. Se planifican las actividades de superación profesional a través de cursos, entrenamientos, talleres, diplomados y otras formas.

La información del programa de las Especialidades se almacena en las Secretarías Docentes de las Facultades Independientes o Institutos Médicos Superiores. Aquí se toma el control total de la información comprendida por: datos generales, de la carrera, de la especialidad, registro de evaluaciones y trámites del profesional. También se manejan las estadísticas asociadas a estos datos. En el decursar de los años, el cúmulo de datos se torna difícil de controlar, lo que dificulta la obtención de cierta información.

El proceso de almacenamiento en estos centros se realiza por el método tradicional. Utiliza modelos en papel y en algunos casos hojas de cálculos. Pero se ha comprobado que no pueden mantener actualizado el registro de información, pues en la mayoría de los casos es almacenada de diferentes formas. Una vez procesada la información, se envía a otros niveles por vía electrónica o telefónica corriendo el riesgo de perderse o duplicarse. Esto conlleva a que el nivel de errores y demora en la entrega de datos sea considerable constituyendo una pérdida de tiempo.

Actualmente el país carece de un sistema informático capaz de gestionar la información para la formación posgraduada, debido a ello no existe disponibilidad de información única, ni confiabilidad en tiempo real para la toma de decisiones.

A partir de lo expuesto anteriormente se plantea el **Problema a resolver**: ¿Cómo facilitar el proceso de gestión de la información relacionada con la formación posgraduada para el área de especialidades en el Ministerio de Salud Pública de Cuba?

Como **Objeto de Estudio** se ha definido como el Proceso de gestión de la información referente a la formación posgraduada en el Ministerio de Salud Pública de Cuba.

El **Campo de Acción** está enmarcado en el Proceso de gestión de la información para el área de especialidades.

De acuerdo al problema planteado anteriormente se propone como **Objetivo General:** Automatizar la gestión de la información del profesional de salud en el área de especialidades para el Departamento de Posgrado en el Ministerio de Salud, mediante una aplicación web.

Para darle cumplimiento al objetivo general se llevan a cabo las siguientes **Tareas de la Investigación:**

1. Analizar los aspectos conceptuales relacionados con la formación de posgrado en el MINSAP y específicamente con el programa de especialidades.
2. Aplicar las tecnologías y arquitectura definidas por el MINSAP y la dirección del proyecto para el desarrollo de la solución.
3. Documentar los artefactos para el Modelo del negocio, Modelo del sistema, Modelo de análisis y Modelo del diseño que se emplean en el desarrollo del módulo de Especialidades.
4. Implementar el Módulo de Especialidades.
5. Integrar el Módulo de Especialidades al “Sistema de Gestión de Información en el Proceso de Formación Posgraduada en Salud”.

Con el desarrollo de módulo de Especialidades se logra mayor rapidez y efectividad en el proceso de gestión de la información para la formación posgraduada. Se facilita el manejo de los datos del programa y se le proporciona al cliente una interfaz amigable, con flexibilidad y claridad para la navegación.

El desarrollo del presente trabajo está estructurado por **cuatro capítulos** que incluye todo el trabajo investigativo. En el primer capítulo "**Fundamentación Teórica**", se explican temas relacionados con el estado del arte a nivel nacional. Se hace un estudio de las tendencias, tecnologías, lenguajes de programación, metodología y herramientas utilizadas para el desarrollo del módulo. Seguidamente el capítulo dos "**Características del Sistema**", aquí se describe el modelo del negocio, los requerimientos funcionales y no funcionales así como los casos de uso del sistema.

En el tercer capítulo "**Análisis y Diseño del Sistema**", se centra en la modelación detallada y la construcción de la estructura de la aplicación. Se muestran los diagramas de clases, los diagramas de interacción y el modelo de datos con la descripción de sus tablas y campos. En el cuarto y último capítulo "**Implementación**", se fundamenta la integración de los módulos desarrollados que conforman el departamento de posgrado y los sistemas externos que se utilizan para un mejor funcionamiento de la aplicación. Se implementan las clases y componentes en términos de subsistemas. Se muestra cómo quedará desplegado el sistema en todas las unidades docentes del país.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

Este capítulo tiene como objetivo fundamental, abordar los temas relacionados con el estado del arte a nivel nacional. Se hace un estudio de las tecnologías que se van a utilizar en el desarrollo del módulo, con el objetivo de cubrir las necesidades que existen donde se aplicará la solución. También se describen los lenguajes de programación, la metodología y las herramientas de trabajos.

1.1 Sistema Nacional de Salud

La forma y los métodos que sirven de base para la organización de la atención a la salud en un país determinado, es lo que se conoce como SNS. La Organización Mundial de la Salud lo define como: “Un complejo de elementos interrelacionados que contribuyen a la salud en los hogares, los lugares de trabajo, los lugares públicos y las comunidades, así como el medio ambiente físico y psicosocial en el sector de salud y otros sectores afines. Además es el conjunto de unidades administrativas, de producción, investigación y servicios, responsabilizado con la atención integral de la salud de una población”. [3]

El SNS cubano está estructurado en tres niveles de dirección, los cuales se encuentran identificados con la estructura político – administrativa del país. Estos niveles son los siguientes:

- ✓ **El Nivel Nacional**, representado por el Ministerio de Salud Pública (MINSAP), como órgano rector con funciones metodológicas, normativas, de coordinación y de control en la aplicación de las políticas del Estado y el Gobierno en cuanto a la salud pública, el desarrollo de las Ciencias Médicas y la Industria Médico Farmacéutica.
- ✓ **El Nivel Provincial**, representado por las Direcciones Provinciales de Salud, directamente subordinadas, administrativa y financieramente a la Asamblea Provincial del Poder Popular (órgano de gobierno a esa instancia).
- ✓ **El Nivel Municipal**, representado por las Direcciones Municipales de Salud y dependientes, administrativa y financieramente de la Asamblea Municipal del Poder Popular (órgano de gobierno a esa instancia).

1.1.1 Informatización del Sistema Nacional de Salud

La informatización del SNS está apoyada en estrategias y políticas trazadas por la dirección del país y el MINSAP, siendo esta una tarea de vital importancia. Con este proceso, se pretende crear un sistema informático para el sector, al que se integran módulos que comprende el Departamento de Posgrado. Permite que todas las unidades de salud del país alcancen un nivel de informatización elevado en las actividades que realicen, influyendo directamente en el aumento gradual de la eficiencia del personal de salud y en la calidad de los servicios que se brinden a la población. Algunas de las políticas a tener en cuenta para la informatización del SNS son las siguientes: [4]

- ✓ Alinearse con tecnologías de punta, estándares de calidad desarrollados en el mundo adecuándolos a condiciones particulares.
- ✓ Todas las inversiones y proyectos que se desarrollen para el SNS deben considerar el elemento informático desde su concepción inicial.
- ✓ La superación y especialización de la informática en salud será una actividad básica para la formación de los recursos humanos.
- ✓ Los productos se desarrollan basados en tecnología LAMP (Linux, Apache, MySQL y PHP) para garantizar sus sustentabilidad en el tiempo donde se emplean estándares internacionales para productos relacionados con la salud pública.

1.2 Surgimiento de la educación de posgrado

La educación de posgrado es hoy, en la mayor parte del mundo, una actividad académica marginal, a veces inexistente. Un indicador de su marginalidad es el reducido espacio que ella ocupa en los medios de comunicación social y en los discursos políticos. Como consecuencia de ello, son ínfimas las asignaciones presupuestarias que la mayoría de los gobiernos o las universidades le asignan. La educación de posgrado tiene sus antecedentes en la antigüedad, se puede afirmar que ella nace y se formaliza en Alemania cuando en 1808 el filólogo y estadista

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

Alejandro de Humboldt (1767-1835) funda la Universidad de Berlín y en ella se establece el Doctorado en Filosofía --el famoso y ya anacrónico Ph.D.-- como el título académico más alto que otorga la Universidad. [5]

En Cuba, el Diplomado se incorporó al Sistema de Superación Profesional a partir de la emisión de la Resolución NO. 6-96 del Ministerio de Educación Superior (MES), que puso en vigor el Reglamento de la Educación de Posgrado de la República de Cuba.

Por tanto, el surgimiento del posgrado es la respuesta al perfeccionamiento del sistema de superación profesional, menguado desde fines de la década de los 80. Desde el punto de vista funcional, constituye parte del proceso de desarrollo profesional que posibilita a los graduados de los diferentes centros docentes la adquisición de los métodos y técnicas profesionales; así como ser parte de una estrategia académica para obtener a continuación Especialidades, Maestría o Doctorado. Se mantiene la correspondencia con el programa académico contemplado y acreditado para estos fines. [6]

1.3 Comienzo de las actividades del programa de las Especialidades en Cuba

La primera actividad de posgrado que de forma programada se iniciara posterior al triunfo de la Revolución, fue Residencia o Especialidad Médica, surgida en 1962. Con anterioridad los pocos médicos que ofrecían atención médica especializada lo hacían por dedicación personal a la misma. Teniendo en cuenta la experiencia acumulada, en el año 1967 se aprobó oficialmente la estructuración de las Residencias para la formación de Especialistas en las diferentes ramas de la Estomatología; y se ponen en ejecución los primeros Programas en 1968. En la década del 70, la evolución hacia nuevas formas de organización en el SNS, introdujo el modelo de atención comunitaria y las Especialidades se ubican en el nivel secundario de atención.

Mientras todo esto ocurre entre 1981- 1985, se efectuó una revisión de los programas, los cuales comenzaron a aplicarse en este último año. En estos programas se introdujeron cambios en el orden metodológico de la enseñanza de posgrado y se le incorporaron aquellos aspectos derivados de las transformaciones ocurridas durante esos años en el SNS. [7]

1.4 Tendencias y tecnologías actuales

Como factor fundamental en este epígrafe se encuentran los patrones de diseño y de arquitectura utilizados para el desarrollo del módulo. Se hace énfasis en el modelo cliente servidor, modelo-vista-controlador, arquitectura en tres capas, y arquitectura orientada a servicios por la secuencia de pasos lógicos a seguir, dando a conocer los argumentos necesarios de su utilización.

1.4.1 Modelo Cliente-Servidor

En la actualidad la información en el Ministerio de Salud se encuentra distribuida por las diferentes unidades de salud del país. Es inevitable la unificación de esta información, pues da la posibilidad de combinarlo y mostrarle al usuario final una interfaz única y bien definida.

Para el desarrollo de sistemas de información en el que las transacciones se dividen en procesos independientes, que cooperan entre sí, se necesita de la aplicación de este modelo. Se denomina cliente al proceso que inicia el diálogo o solicita los recursos; y servidor al proceso que responde a las solicitudes. Las aplicaciones se dividen de forma que el servidor contiene la parte que debe ser compartida por varios usuarios, y en el cliente permanece sólo lo particular de cada usuario. [8]

Se pueden apreciar algunas características del modelo cliente-servidor, como: el cliente no necesita conocer la lógica del servidor. Sólo su interfaz externa. Este no depende de la ubicación física del servidor, ni del tipo de equipo físico en el que se encuentra, ni de su sistema operativo.

Los clientes generalmente realizan funciones como: Manejo de la interfaz de usuario, captura y validación de los datos de entrada y generación de consultas e informes sobre las bases de datos. Los servidores por su parte gestionan los periféricos compartidos, controlan el acceso concurrente a bases de datos compartidas y enlazan las comunicaciones con otras redes de área local o extensa.

1.4.2 Patrones de arquitectura y diseño

Los patrones son unidades de información que describen un problema particular y recurrente del diseño, que surgen en un contexto específico. El objetivo de los patrones es crear un lenguaje

común para la comunidad de los desarrolladores, que les permita dar solución al problema planteado. Existen varios tipos de patrones, que dependen del contexto en el que se aplican o de la etapa en la que se encuentra el proceso. Se pueden citar algunos como: de arquitectura, de diseño, de negocios y de análisis.

1.4.3 Arquitectura Orientada a Servicios (SOA)

Dada la situación presentada es necesario aplicar esta arquitectura por el contexto en que está enmarcada. La Arquitectura Orientada a Servicios (en inglés Service-Oriented Architecture o SOA), es un concepto de arquitectura de software que define la utilización de servicios para dar soporte a los requerimientos de software del usuario. SOA proporciona una metodología y un marco de trabajo, puede dar soporte a las actividades de integración y consolidación.

En un ambiente SOA, los nodos de la red hacen disponibles sus recursos a otros participantes, como servicios independientes a los que pueden tener acceso. La mayoría de las definiciones de SOA identifican la utilización de Servicios Web (empleando SOAP y WSDL) en su implementación, no obstante se puede implementar una SOA utilizando cualquier tecnología basada en servicios.

Se pueden citar algunas ventajas de la Arquitectura Orientada a Servicio como: la administración unificada, transparencia de locación, reusabilidad simplificada, buenos niveles de escalabilidad y paralelismo en el desarrollo. [9]

1.4.3.1 Tecnología Servicio Web

Uno de los mecanismos para el intercambio de mensajes en una arquitectura SOA son los Servicios Web, Son una colección de protocolos y estándares que sirven para intercambiar datos entre aplicaciones. Las aplicaciones de software desarrolladas en lenguajes de programación diferentes, y ejecutadas sobre cualquier plataforma, pueden utilizar los Servicios Web para intercambiar datos en redes de ordenadores como Internet. [10]

Ventajas

- ✓ Los Servicios Web fomentan los estándares y protocolos basados en texto, que hacen más fácil acceder a su contenido y entender su funcionamiento.
- ✓ Permiten que servicios y software de diferentes compañías, ubicadas en diferentes lugares geográficos, puedan ser combinados fácilmente para proveer servicios integrados.

Inconvenientes de los Servicios Web

- ✓ Omite medidas de seguridad basadas en firewall, cuyas reglas de seguridad tratan de bloquear o auditar la comunicación entre ambos lados de la barrera. Esto sucede porque usan protocolos de comunicación HTTP.
- ✓ El rendimiento es bajo con otros modelos de computación pues no presentan como objetivo la concisión ni la eficiencia de procesamiento.
- ✓ No es tan eficiente como otros estándares abiertos de computación distribuida para realizar transacciones.

Un programa cliente que se conecta a un servicio web puede leer el WSDL para determinar las funciones que están disponibles en el servidor. Los tipos de datos especiales se incluyen en el archivo WSDL en forma de XML Schema. El cliente puede usar SOAP para hacer la llamada a una de las funciones listadas en el WSDL.

En la implementación de este módulo se reutilizaron funciones de estos Servicios Web, lo que hace posible la reducción considerable de tiempo y costo del desarrollo. Esto incrementa la robustez del nuevo sistema, al utilizarse software ya probado.

1.4.4 Arquitectura en tres capas

La programación por capas es un estilo de programación en la que el objetivo primordial es la separación de la lógica de negocios de la lógica de diseño. Un ejemplo básico de esto es separar la capa de datos de la capa de presentación al usuario. La ventaja principal de este estilo, es que el desarrollo se puede llevar a cabo en varios niveles. En caso de existir algún cambio, sólo se ataca

al nivel requerido sin tener que revisar entre códigos mezclados. Algunas de las ventajas que proporciona esta arquitectura: la reutilización de capas y facilidad de estandarización. [11]

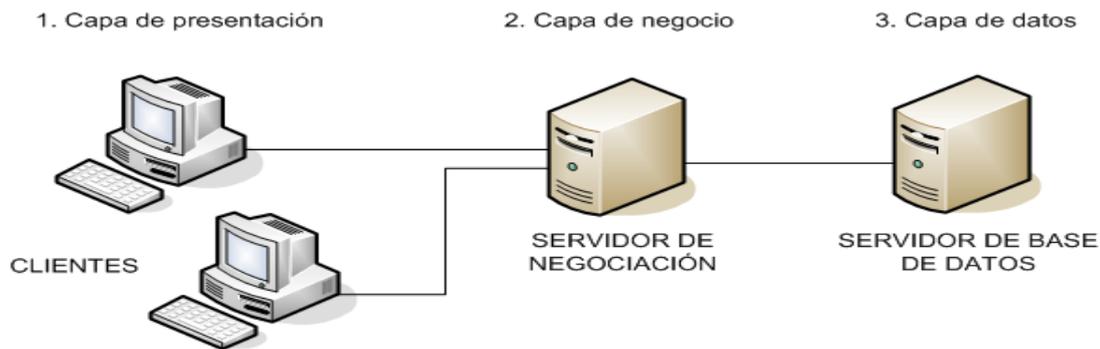


Fig. 1 Arquitectura en tres capas

A continuación se dan detalles de las capas:

- ✓ **Capa de presentación:** es la que ve el usuario (hay quien la denomina capa de usuario). Presenta el sistema al usuario, le comunica y captura la información del usuario. Realiza un filtrado previo para comprobar que no hay errores de formato. Esta capa se comunica únicamente con la capa de negocio.
- ✓ **Capa de negocio:** es donde residen los programas que se ejecutan. Se reciben las peticiones del usuario y se envían las respuestas tras el proceso. Se denomina capa de negocio (e incluso de lógica del negocio) pues es aquí donde se establecen todas las reglas que deben cumplirse. Esta capa se comunica con la capa de presentación, para recibir las solicitudes y presentar los resultados, y con la capa de datos, para solicitar al gestor de base de datos para almacenar o recuperar datos.
- ✓ **Capa de datos:** es donde residen los datos y es la encargada de acceder a los datos. Está formada por uno o más gestores de bases de datos que realizan todo el almacenamiento de información. Recibe solicitudes de almacenamiento o recuperación de información desde la capa de negocio.

1.4.4.1 Modelo Vista Controlador (MVC)

Este patrón de arquitectura de software permite separar los datos de una aplicación, la interfaz de usuario y la lógica de negocio en tres componentes distintos, esto proporciona múltiples vistas sobre un mismo modelo de datos. El patrón MVC se usa frecuentemente en aplicaciones web donde se utilicen diferentes interfaces de usuario. El código que provee los datos a la página es dinámico. Los tres elementos esenciales de este patrón son los siguientes: [12]

- ✓ **Modelo:** Administra el comportamiento y los datos del dominio de la aplicación. Responde a requerimientos de información sobre su estado, usualmente formulada desde la vista, respondiendo a instrucciones de cambio de estos datos, desde el controlador.
- ✓ **Vista:** Este presenta el modelo en un formato adecuado para interactuar. Usualmente un elemento de interfaz de usuario.
- ✓ **Controlador:** Este responde a eventos, con frecuencia acciones del usuario e invoca cambios en el modelo y probablemente en la vista.

Tanto la vista como el controlador dependen del modelo, el cual no depende de otros conceptos o clases. Esta separación permite construir y probar el modelo independientemente de la representación visual. En aplicaciones web, por ejemplo, la separación entre la vista (navegador) y el controlador (componentes del lado del servidor que manejan los requerimientos a través de HTTP) está claramente definida. Entre las ventajas del estilo señaladas por Microsoft están las siguientes:

- ✓ **Soporte de vistas múltiples:** Dado que la vista se encuentra separada del modelo, la interfaz de usuario puede mostrar múltiples vistas de los datos de manera simultánea.
- ✓ **Adaptación al cambio:** Los requerimientos no funcionales de interfaz de usuario tienden a cambiar con mayor rapidez que las reglas del negocio. Los clientes pueden preferir distintas opciones de representación, pero como el modelo no depende de la vista, agregar nuevas opciones o modificar las ya existentes generalmente no afecta al modelo.

Entre las desventajas se han señalado:

- ✓ **Complejidad:** El patrón introduce nuevos niveles de indirección y por lo tanto aumenta ligeramente la complejidad de la solución. También se profundiza la orientación a eventos del código de la interfaz de usuario, que pueden llegar a ser difíciles de depurar.
- ✓ **Costo de actualizaciones frecuentes:** Desacoplar el modelo de la vista no significa que los desarrolladores del modelo puedan ignorar la naturaleza de las vistas.

1.4.5 ¿Por qué utilizar SOA y la Arquitectura en 3 Capas?

La Arquitectura Orientada a Servicios permite la integración de la aplicación con las que se utilizan en el Sistema Nacional de Salud independientemente de la plataforma, lenguaje y arquitectura en los que se desarrollen. Los principios y estándares propuestos por SOA, garantizan la implementación de un sistema capaz de cumplir con las exigencias que la aplicación requiere.

Exponer procesos de negocio como servicios, es la clave a la flexibilidad de la arquitectura. Esto permite que otras piezas de funcionalidad (incluso también implementadas como servicios) hagan uso de otros servicios de manera natural, sin importar su ubicación física. Así un sistema evoluciona con la adición de nuevos servicios y su mejoramiento, donde cada servicio evoluciona de una manera independiente.

La Arquitectura en 3 capas, mediante el modelo cliente-servidor permite separar la lógica de negocios de todo lo que tenga que ver con el origen de datos. Esto brinda una estructura más organizada a los desarrolladores, y proporciona un menor tiempo de solución de errores en caso de efectuarse cambios.

1.4.6 Tecnología AJAX

Acrónimo de Asynchronous Java Script And XML (*Java Script y XML asíncronos, donde XML es un acrónimo de eXtensible Markup Language*), es un conjunto de tecnologías para crear aplicaciones interactivas. Estas 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. [13]

AJAX es una combinación de tres tecnologías ya existentes:

- ✓ **XHTML** (o **HTML**) y hojas de estilos en cascada (**CSS**) para el diseño que acompaña a la información. **Document Object Model (DOM)** accedido con un lenguaje de scripting por parte del usuario, especialmente implementaciones **ECMAScript** como **Java Script** y **JScript**, para mostrar e interactuar dinámicamente con la información presentada.
- ✓ El objeto **XMLHttpRequest** para intercambiar datos asincrónicamente con el servidor web. En algunos frameworks y en algunas situaciones concretas, se usa un objeto **iframe** en lugar del **XMLHttpRequest** para realizar dichos intercambios.
- ✓ **XML** es el formato usado comúnmente para la transferencia de vuelta al servidor. Cualquier formato puede funcionar, incluyendo **HTML** pre formateado, texto plano, **JSON** y hasta **EBML**.

1.4.7 Motor de plantilla Smarty

El mecanismo de plantillas sirve para separar el código de presentación del resto del código de una aplicación web. Consiste en codificar todo lo que tenga que ver con la presentación en una serie de plantillas de código **HTML** u otro lenguaje de presentación, con expresiones sencillas intercaladas para comunicarse con el resto de la aplicación y poder mostrar datos dinámicos. Un motor de plantillas es el que se encarga de hacer la traducción a **HTML**. De esta forma, las vistas simplemente se ocupan de extraer la información necesaria del modelo y comunicársela a las plantillas, y estas simplemente de dar un formato visual a esa información, añadiendo la información estática pertinente. Puede ser usado sin un entorno de desarrollo. [14]

1.4.8 Frameworks de diseño

Yahoo User Interface (YUI): Es una serie de librerías escritas en **Java Script**, para la construcción de aplicaciones interactivas. Liberadas bajo licencia **BSD** por parte de la compañía **Yahoo**. Dichas librerías son utilizadas para el desarrollo web específicamente para ser usadas como la

programación de aplicaciones de escritorio, con componentes vistosos y personalizables y con una amplia implementación con AJAX. [15]

1.5 Leguajes de programación utilizados

Para la implementación del sistema se utilizaron lenguajes de programación entre los que se encuentran los del lado del servidor y del lado del cliente. Entre los lenguajes del lado del servidor se pueden citar algunos: ColdFunction, PERL, ASP, PHP, JSP y los módulos CGIs. Estos lenguajes desarrollan la lógica de negocio dentro del servidor. Además se encargan de los accesos al almacenamiento físico de los datos dentro de los distintos Sistemas de Gestión de Bases de Datos. Entre los lenguajes que trabajan del lado del cliente se encuentran el Java Script, XSLT y el Visual Basic Script. Estos dos últimos al combinarse con el HTML forman lo que se conoce como DHTML, es decir, salida estándar dinámica o HTML dinámico.

Esta distinción entre los lenguajes ha sido necesaria debido a que el protocolo HTTP es un protocolo sin estado. No guarda información sobre conexiones anteriores y al finalizar la transacción los datos se pierden. Cada petición/respuesta es una operación distinta, por lo que la web trabaja en modo desconectado; o sea, un usuario a través de un navegador hace una petición de una página a un servidor web. Este obtiene la petición, la procesa y le envía la respuesta al cliente, el cual la recepciona y se desconecta.

1.5.1 PHP

Las siglas significan "PHP Hypertext Pre-processor" (*inicialmente PHP Tools, o, Personal Home Page Tools*), y se trata de un lenguaje interpretado usado para la creación de aplicaciones para servidores, o creación de contenido dinámico para sitios web. Es gratuito e independiente de plataforma, muy rápido, con una gran librería de funciones y mucha documentación. En el caso de estar montado sobre un servidor Linux o Unix, es uno de los más rápidos, dado que se ejecuta en un único espacio de memoria.

Es un lenguaje de programación usado generalmente para la creación de contenido para sitios web. Permite la conexión a diferentes tipos de servidores de bases de datos tales como MySQL, Postgres, Oracle, ODBC y Microsoft SQL Server. Tiene la capacidad de ser ejecutado en la

mayoría de los sistemas operativos tales como UNIX, Windows y Mac OS X, y puede interactuar con los servidores web más populares como Apache. [16]

1.5.2 Java Script

Es un lenguaje interpretado, es decir, que no requiere compilación, orientado a las páginas web. Dirigido por eventos, por lo que estará listo para actuar en cuanto un evento sea ejecutado, implementa una sencilla interfaz de objetos/propiedades/métodos. Se integra dentro del código HTML de las páginas web. Se ejecuta en el navegador al mismo tiempo que las sentencias van descargándose junto con el código HTML. Brinda rapidez a la aplicación web a la hora de las validaciones de los formularios. Gracias a su compatibilidad con la mayoría de los navegadores modernos, es el lenguaje de programación del lado del cliente más utilizado. [17]

1.5.3 HTML

El HTML, acrónimo inglés de HyperText Markup Language (lenguaje de marcas hipertextuales), es un lenguaje de etiquetas diseñado para estructurar textos y presentarlos en forma de hipertexto, que es el formato estándar de las páginas web. Gracias a Internet y a los navegadores del tipo Internet Explorer, Opera, Firefox o Netscape, el HTML se ha convertido en uno de los formatos más populares que existen para la construcción de documentos, es una aplicación de SGML conforme al estándar internacional ISO 8879. [18]

1.6 Sistema Gestor de base de datos utilizado (SGBD)

Es una interfaz entre el usuario y las bases de datos, se trata de un software de propósito general que ayuda en los procesos de definición, construcción y manipulación de una base de datos. Es un conjunto de programas que permite crear y mantener una base de datos, asegurando su integridad, confidencialidad y seguridad. Algunos de los sistemas más conocidos son: PostgreSQL, MySQL, Microsoft Access, Microsoft SQL Server y Oracle. Estos sistemas deben permitir la especificación de tipos, estructuras y restricciones de datos; la construcción de base de datos que permita guardar la información en algún medio controlado por el gestor y debe permitir la manipulación para realizar consultas, actualizarla y generar informes. [19]

1.6.1 MySQL 5.0

MySQL es una aplicación que permite gestionar archivos llamados de base de datos. Como base de datos relacionales, utiliza múltiples tablas para almacenar la información. Destaca por su adaptación a diferentes entornos de desarrollo, lo que permite su interacción en distintos sistemas operativos. Esto ha favorecido su desarrollo y continuas actualizaciones. Ha sido diseñado con el objetivo de aumentar la velocidad. Consume muy pocos recursos de CPU y memoria lo que proporciona un alto rendimiento. Tiene buena integración con PHP y facilita el control de acceso usuarios-tablas-permisos. [20]

1.7 Servidor Web Apache 2.2

Está diseñado para ser un servidor web potente y flexible que pueda funcionar en la más amplia variedad de plataformas y entornos. Las diferentes plataformas y entornos, hacen que a menudo sean necesarias diferentes características o funcionalidades. Apache se ha adaptado siempre a una gran variedad de entornos a través de su diseño modular. Este diseño permite a los administradores de sitios web elegir las características que van a ser incluidas en el servidor, seleccionando los módulos que se cargan, ya sea al compilar o al ejecutar el servidor. [21]

Las principales características que tiene este Servidor es que se puede utilizar tanto en sistemas Unix como en sistemas Windows, existiendo versiones diferentes para cada uno de los sistemas. Es de código abierto y totalmente gratuito. Tiene una arquitectura modular, que permite la construcción del propio servidor mediante paquetes de pequeño tamaño. De esos paquetes hay algunos que son obligatorios.

1.8 Metodología utilizada para el desarrollo del software

Para el desarrollo del software se debe contar con un proceso bien detallado y para esto se necesita aplicar una metodología que sea capaz de llevar a cabo el control total del producto. De esta forma se cumple con el objetivo principal de asegurar la producción de software de alta calidad, con el fin de satisfacer las necesidades de los usuarios finales. Esto se realiza dentro de un cronograma y con presupuestos bien definidos.

Algunas de las metodologías que se ponen en práctica a la hora de desarrollar un software son las siguientes: metodología orientada a objetos planteada por Rational (la metodología RUP), la metodología XP, y alguna mixtura de metodología Ágil. Se ha definido como metodología de desarrollo la metodología basada en RUP.

1.8.1 El ciclo de vida RUP

El ciclo de vida RUP es una implementación del desarrollo en espiral. Fue creado ensamblando los elementos en secuencias semi-ordenadas. El ciclo de vida organiza las tareas en fases e iteraciones. Esta metodología divide el proceso de desarrollo en ciclos, teniendo un producto final al culminar cada uno de ellos. Estos a su vez se dividen en fases que finalizan con un hito donde se debe tomar una decisión importante:

- ✓ **Concepción:** se hace un plan de fases, se identifican los principales casos de uso y se identifican los riesgos.
- ✓ **Elaboración:** se hace un plan de proyecto, se completan los casos de uso y se eliminan los riesgos.
- ✓ **Construcción:** se concentra en la elaboración de un producto totalmente operativo y eficiente y en el manual de usuario.
- ✓ **Transición:** se instala el producto en el cliente y se entrena a los usuarios. Como consecuencia suelen surgir nuevos requisitos a ser analizados.
- ✓ **Mantenimiento:** una vez instalado el producto, el usuario realiza requerimientos de ajuste, esto se hace de acuerdo a solicitudes generadas como consecuencia del interactuar con el producto.

RUP es un producto de Rational (IBM). Se caracteriza por ser iterativo e incremental, estar centrado en la arquitectura y guiado por los casos de uso. Incluye artefactos, que son los productos tangibles del proceso: el modelo de casos de uso, el código fuente. Incluye los roles, que son el papel desempeñado de una persona en un determinado momento, una persona puede desempeñar distintos roles a lo largo del proceso.

Provee a cada miembro del equipo un fácil acceso a una base de conocimiento con guías, plantillas y herramientas para todas las actividades críticas del desarrollo de software. Esta

metodología permite que todos los integrantes de un equipo de trabajo conozcan y compartan el proceso de desarrollo, una base de conocimientos y los distintos modelos de cómo desarrollar el software utilizando un lenguaje de modelado común: UML.

1.8.2 UML

Lenguaje Unificado de Modelado (UML, por sus siglas en inglés, *Unified Modeling Language*) es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad. Es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema de software. UML ofrece un estándar para describir un "plano" del sistema (modelo), incluyendo aspectos conceptuales tales como procesos de negocios, funciones del sistema y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y componentes de software reutilizables.

Se utiliza para definir un sistema de software, para detallar los artefactos en el sistema y para documentar y construir. En otras palabras, es el lenguaje en el que está descrito el modelo. Se puede aplicar en una gran variedad de formas para dar soporte a una metodología de desarrollo de software (tal como el Proceso Unificado Racional), pero no especifica en sí mismo qué metodología o proceso utilizar. [22]

1.9 Herramientas utilizadas

Para lograr mejor arquitectura del sistema y tener conocimiento de la problemática planteada, se definieron las herramientas necesarias para desarrollar el módulo de Especialidades. En la primera etapa de desarrollo se utilizó el Rational Rose Enterprise Edition 2003 para realizar los diagramas como guía y argumentos de la documentación. Para la creación de páginas web, se utilizó Dreamweaver 8, pues sus características soportan el lenguaje de marca XHTML. Para la edición de páginas PHP, se utilizó el Zend Studio 5. Para la administración y desarrollo de servidores de base de datos se utilizó la poderosa herramienta EMS SQL Manager, ya que soporta MySQL. Por último se utilizó el subversion para almacenar las actualizaciones que se realizan en el transcurso del desarrollo del software.

1.9.1 Rational Rose Enterprise Edition 2003

Rational Rose es la herramienta CASE desarrollada por los creadores de UML (*Booch, Rumbaugh y Jacobson*), que cubre todo el ciclo de vida de un proyecto: concepción y formalización del modelo, construcción de los componentes, transición a los usuarios y certificación de las distintas fases entregables.

Es una herramienta con plataforma independiente que ayuda a la comunicación entre los miembros de equipo, a monitorear el tiempo de desarrollo y a entender el entorno de los sistemas. Una de las grandes ventajas que posee es que utiliza la notación estándar en la arquitectura de software (UML). Esta permite a los arquitectos de software y desarrolladores visualizar el sistema completo utilizando un lenguaje común, además los diseñadores pueden modelar sus componentes e interfaces en forma individual y luego unirlos con otros componentes del proyecto. [23]

Rational permite completar una gran parte de flujos fundamentales en concreto:

- ✓ Modelado del negocio.
- ✓ Captura de requisitos.
- ✓ Análisis y diseño.
- ✓ Implementación.
- ✓ Control de cambios y gestión de configuración.

1.9.2 Dreamweaver 8

Es el programa de este tipo más utilizado en el sector del diseño y la programación web, por sus funcionalidades, su integración con otras herramientas como Adobe Flash y, recientemente, por su soporte de los estándares del World Wide Web Consortium [24].

Existen varias funciones típicas que se emplean como un editor de código fuente, entre las que se pueden destacar el administrador de sitios, que sirve para agrupar los archivos según el proyecto al que pertenezcan. Funciona como cliente FTP integrado, que permite subir los archivos editados inmediatamente al sitio en Internet. Posee la función de auto completar y resaltado de la sintaxis para instrucciones en HTML y lenguajes de programación como PHP, JSP o ASP.

1.9.3 Zend estudio

Se trata de un programa de la casa Zend, impulsores de la tecnología de servidor PHP, orientada a desarrollar aplicaciones web, en lenguaje PHP. Proporciona una serie de ayudas que pasan desde la creación y gestión de proyectos hasta la depuración de código. Zend Studio consta de dos partes en las que se dividen las funcionalidades de parte del cliente y las del servidor. Las dos partes se instalan por separado: la del cliente contiene la interfaz de edición y la ayuda. Permite además hacer depuraciones simples de scripts. Aunque para disfrutar de toda la potencia de la herramienta de depuración habrá que disponer de la parte del servidor, que instala Apache y el módulo PHP o, en caso de que estén instalados, los configura para trabajar juntos en depuración. [25]

1.9.4 EMS SQL Manager

Es una poderosa herramienta para la administración y desarrollo de servidores de base de datos MySQL y PostgreSQL. Dentro de las funciones incluidas se encuentra gestión de privilegios para usuarios, ejecución de Scripts, acceso a peticiones visuales previamente desarrolladas e importación/exportación de datos. Gracias a un entorno muy accesible, EMS SQL Manager permite ingresar y ajustar diversos aspectos de las bases de datos.

1.9.5 Controlador de versiones

Subversion es un software de sistema de control de versiones. Es software libre, bajo una licencia de tipo Apache/BSD. Una característica importante de Subversion es que, a diferencia de CVS, los archivos versionados no tienen un número de revisión independiente cada uno. En cambio, todo el repositorio tiene un único número de versión que identifica un estado común de todos los archivos en cierto punto del tiempo.

El repositorio es un sistema centralizado para compartir información. Esta almacena información en forma de un árbol de ficheros, una jerarquía típica de ficheros y directorios. Cualquier número de clientes se conectan al repositorio, y luego leen o escriben esos ficheros. Al escribir datos, el cliente hace que la información esté disponible para los otros. Al leer los datos, el cliente recibe la información de los demás.

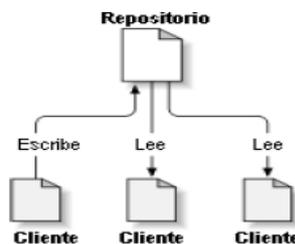


Fig. 2 Un sistema típico cliente/servidor

Cuando un cliente lee datos de un repositorio, normalmente ve únicamente la última versión del árbol de ficheros. Pero el cliente también tiene la capacidad de ver estados previos del sistema de ficheros. Por ejemplo, un cliente puede hacer preguntas históricas, como "¿qué contenía este directorio el último miércoles?", o "¿quién fue la última persona que cambió este fichero, y qué cambios hizo?" Esta es la clase de preguntas que forman el corazón de cualquier sistema de control de versiones: son sistemas que están diseñados para guardar y registrar los cambios a los datos a lo largo del tiempo. [26]

Ventajas

- ✓ Se sigue la historia de los archivos y directorios a través de copias y renombrados.
- ✓ Permite selectivamente el bloqueo de archivos. Se usa en archivos binarios que, al no poder fusionarse fácilmente, conviene que no sean editados por más de una persona a la vez.
- ✓ Cuando se usa integrado a Apache permite utilizar todas las opciones que este servidor provee a la hora de autenticar archivos (SQL, LDAP, PAM).

1.9.5.1 TortoiseSVN.

Es software libre que maneja ficheros y directorios a lo largo del tiempo. Los ficheros se almacenan en un repositorio central. Esto permite que pueda recuperar versiones antiguas de sus ficheros y

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

examinar la historia de cuándo y cómo cambiaron sus datos, y quién hizo el cambio. Esta es la razón por la que mucha gente piensa que Subversion, y los sistemas de control de versiones en general, son una especie de “máquinas del tiempo”.

Algunos sistemas de control de versiones también son sistemas de manejo de configuración del software. Estos sistemas están diseñados específicamente para manejar árboles de código fuente, y tienen muchas características que son específicas para el desarrollo de software, tales como el entendimiento nativo de los lenguajes de programación, o proporcionan herramientas para compilar software. Subversion, sin embargo, no es uno de estos sistemas; es un sistema general que puede ser utilizado para manejar cualquier colección de ficheros, incluyendo código fuente. [27]

En este capítulo se profundizó en el conocimiento de la informatización del Sistema Nacional de Salud y del surgimiento del posgrado hasta la actualidad. Se analizaron algunas actividades desarrolladas en Cuba con respecto al programa de las Especialidades. Se ofreció una breve panorámica de cómo se realiza la Gestión de la Información en el MINSAP. Finalmente se realizó un estudio de las tecnologías utilizadas en el desarrollo del sistema propuesto. Se fundamentó en la elección de los lenguajes de programación, la metodología de desarrollo, el sistema gestor de base de datos y las herramientas utilizadas.

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

En este capítulo se realiza un estudio de los aspectos relacionados con el objeto de estudio donde se definen la situación problemática, los conceptos asociados al dominio del módulo y los procesos a automatizar. Se plantean los requerimientos funcionales y no funcionales de la aplicación a desarrollar y se modela la misma en términos de casos de uso de sistema.

2.1 Objeto de estudio

2.1.1 Situación Problemática

La información del programa de las Especialidades se recoge en las Secretarías Docentes de las Facultades Independientes o Institutos Médicos Superiores. Aquí se toma el control total de la información comprendida por: datos generales, de la carrera, de la especialidad, registro de evaluaciones y trámite del profesional. También se manejan las estadísticas de este tipo de datos. Año tras año se vuelve más difícil controlar este cúmulo de datos, pues crece a grandes volúmenes. Dicha información es procesada por el método tradicional, utilizando el formato papel y en algunos casos la herramienta Excel. No obstante se ha comprobado que las soluciones propuestas no han sido factibles, pues en la mayoría de los casos no es posible la actualización del registro de la formación posgraduada.

Una vez procesada la información, se envía a otros niveles por vía electrónica o telefónica corriendo el riesgo de perderse o duplicarse. Esto conlleva a que el nivel de errores y demora en la entrega de datos sea considerable, lo que constituye una pérdida de tiempo. Por lo anteriormente expuesto se puede afirmar que actualmente el país carece de un sistema informático capaz de gestionar la información para la formación posgraduada. Por esta razón, no existe disponibilidad de información única, confiable en tiempo real para la toma de decisiones.

2.2 Modelo de negocio

El proceso del modelamiento del negocio, es el principal requisito para tener el conocimiento básico y necesario para afrontar la problemática presentada y darle solución con la mayor calidad

posible. Con el proceso de modelamiento del negocio se da una visión clara y precisa, capaz de definir procesos, roles y asignar responsabilidades para los casos de usos correspondientes. Todo esto, a través del modelo de casos de uso del negocio y el modelo de objeto. Para profundizar más acerca de la descripción de los casos de uso y los diagramas de actividades, se puede ver el [ANEXO 1](#).

2.2.1 Actores del Negocio

ACTOR	DESCRIPCIÓN
Profesional de Salud.	Este actor es el interesado en que se le realice la matrícula en la especialidad. Se le gestiona su formación docente. Tales son: sus evaluaciones en los exámenes realizados y los programas (cursos, estancias, rotación) asignados. Puede solicitar movimientos según las causas expuestas.
Metodólogo de Posgrado.	Este actor es el interesado en conocer los reportes que le competen según su nivel de acceso (Nacional, Provincial y Municipal).

2.2.2 Trabajadores del Negocio

ACTOR	DESCRIPCIÓN
Editor Nacional.	Este actor se encarga de adicionar, eliminar y modificar los codificadores referentes a las especialidades y los programas en cuestión. Realiza cambios de acuerdo a las exigencias del MINSAP.
RPS (Registro Personal de la Salud).	Este actor será consultado para acceder a los datos de los profesionales que son insertados en la base de datos del sistema. Se pueden recoger datos de su carrera, que se utilizan posteriormente, para manejar los datos de su formación posgraduada.

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

SAAA(Single Authentication, Authorization and Accounting).	Sistema externo que permite la reutilización de funciones con el objetivo de garantizar la seguridad del sistema. Los usuarios tienen que estar previamente autenticados para acceder a los datos.
Secretario Docente.	Este actor es el encargado de realizar la matrícula de los profesionales al programa de las especialidades. Gestiona las evaluaciones de los mismos y registra los movimientos de acuerdo a las causas expuestas. Antes de realizar cualquiera de estas operaciones se procede a buscar el profesional. Es un tipo de editor que trabaja a un nivel provincial, o lo que es lo mismo a nivel de facultades.
RU (Registro de Ubicación).	Sistema externo que admite la reutilización de funciones. Permite conocer la ubicación residente de los profesionales de la salud, como la provincia y el municipio al que corresponden.
RUS (Registro de Unidades de Salud).	Sistema externo que admite la reutilización de funciones. Permite consultar los datos de las unidades asistenciales a las que están vinculados los profesionales de la salud.
Usuario.	Este actor puede ser editor y visualizador. Tiene la función de autenticarse en el sistema. De acuerdo al nivel de acceso que tenga se otorgan los permisos necesarios.
Visualizador.	Este actor tiene la función de ver los informes de acuerdo a los diferentes niveles de acceso. Estos pueden ser: nacional, provincial y municipal.

2.2.3 Diagrama de Casos de Uso del Negocio

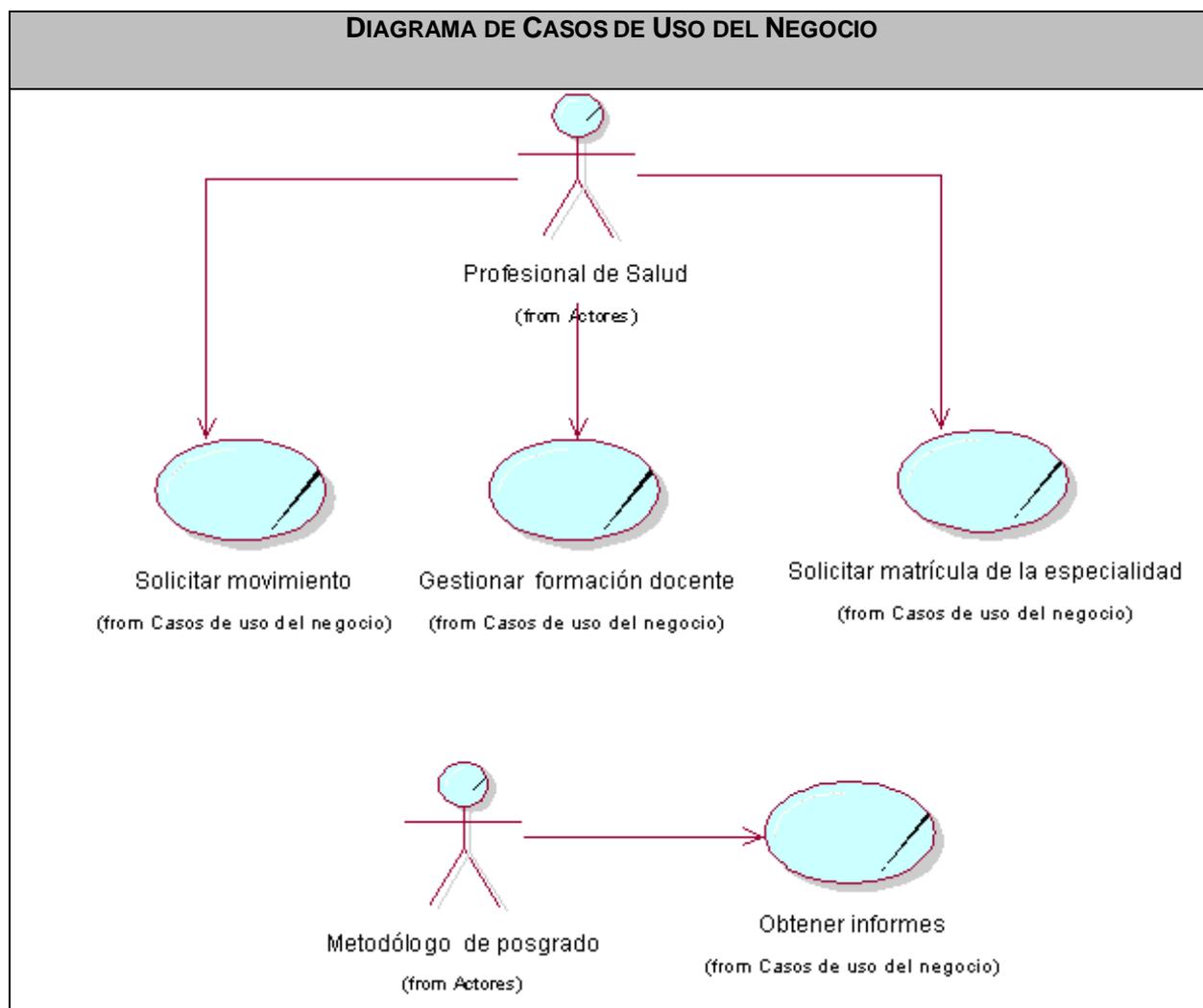


Fig. 2.1 Diagrama de casos de uso del negocio

2.3 Descripción de los procesos del negocio

➤ Solicitar matrícula de la especialidad

El proceso comienza cuando el profesional de nuevo ingreso se dirige a la Facultad o Instituto Médico al que pertenece a matricularse en una especialidad. Según su vía de ingreso, una vez realizado los trámites para recibir su aprobación, se recogen los datos generales, académicos y de

la especialidad que le fue asignada. Toda su información es almacenada en la Facultad o Instituto Médico donde fue matriculado.

➤ **Control de la gestión de formación docente**

El proceso comienza, una vez hecha la matrícula, al presentarse la necesidad de gestionar la formación docente de un profesional con el objetivo de llevar un control de sus evaluaciones. Aquí se recogen las notas de los exámenes de promoción, estatal y el trabajo de terminación de la especialidad. Según en el tipo de especialidad que fue matriculado, se registran cursos, estancias y rotaciones por las que transita el profesional durante su superación.

➤ **Solicitar movimiento**

El calificativo movimiento consiste en las bajas por las diferentes causas estipuladas por el MES. Entre ellas se encuentran bajas académicas, bajas por deserción, bajas por inasistencia, bajas por sanción disciplinaria y bajas voluntarias.

El proceso de bajas definitivas está dado por el traslado de especialidad. También puede ser causado por negligencias del profesional. El proceso de baja temporal es la solicitud de licencia de matrícula (interrupción de su formación docente) por un tiempo definido según las causas presentadas. Por último está el proceso de traslado que constituye, como su nombre lo indica, traslado de centro, según sus causas, que pueden ser problemas personales, condiciones de trabajo, entre otras.

2.4 Propuesta del sistema

Con la realización de este trabajo se propone el desarrollo de un módulo que interactúe con la RED INFOMED que brinde sus servicios a todos los involucrados en el Dpto. de posgrado, donde tienen permisos autorizados para interactuar con la información. Existen privilegios de flexibilidad en los aspectos relacionados con la configuración. Se cuenta con un nodo compuesto por dos servidores; uno de aplicación y uno de base de datos. Este contiene toda la información y está disponible según el usuario que la necesite.

La solución propuesta posee tecnologías de avanzada, desarrolladas sobre la web, lo que posibilita el acceso de información según el nivel de acceso. Para el acceso a la información el módulo contará con una administración asignada por roles, visualizadores y editores. Los editores según el nivel de acceso pueden modificar cualquier información que le corresponde, es decir, solamente de la facultad a la que pertenecen. Los visualizadores, como su nombre indica, sólo tienen derecho a visualizar su información y la de los niveles inferiores.

2.5 Especificación de los Requerimientos de Software

Los requerimientos de software son el principal eslabón para llevar a cabo un desarrollo con calidad y eficiencia. Con la descripción de los requerimientos del sistema se logró el nivel de entendimiento entre el usuario y los desarrolladores, con el objetivo de satisfacer las necesidades de los clientes.

2.5.1 Requerimientos Funcionales

Los requerimientos funcionales son capacidades o condiciones que el sistema debe cumplir. Para el módulo se definen los siguientes:

LISTADO DE REQUERIMIENTOS FUNCIONALES	
RF1- Autenticar usuario.	RF11- Matricular profesional en especialidad.
RF2- Buscar profesional.	RF12- Buscar matriculado.
RF3- Listar especialidades.	RF13- Matricular por programas.
RF4- Adicionar especialidades.	RF14- Asignar evaluación a examen de especialidad.
RF5- Modificar especialidades.	RF15- Registrar movimientos.
RF6- Eliminar especialidades.	RF16- Solicitar traslado.
RF7- Listar programas.	RF17- Atender solicitud de traslado.
RF8- Adicionar programas.	RF18- Realizar baja.
RF9- Modificar programas.	RF19- Causar baja de especialidad.
RF10- Eliminar programas.	

Tabla 2.1 Requerimientos Funcionales del Sistema

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

DEFINICIÓN DE CASOS DE USO.	
CU-1: Autenticar. RF1- Autenticar usuario.	CUS-2: Buscar profesional. RF2- Buscar profesional.
CU-3: Gestionar codificador. RF3- Listar especialidades. RF4- Adicionar especialidades. RF5- Modificar especialidades. RF6- Eliminar especialidades. RF7- Listar programas. RF8- Adicionar programas. RF9- Modificar programas. RF10- Eliminar programas.	CU-4: Registrar matrícula. RF11- Matricular profesional en especialidad. RF12- Buscar matriculado.
CU-5: Gestionar evaluación. RF13- Matricular por programas. RF14- Asignar evaluación a examen de especialidad.	CU-6: Registrar movimiento. RF15- Registrar movimientos. RF16- Solicitar traslado. RF17- Atender solicitud de traslado. RF18- Realizar baja. RF19- Causar baja de especialidad.

Tabla 2.2 Requerimientos por Casos de Uso del Sistema

Una vez identificados los requerimientos funcionales son agrupados en casos de uso del sistema. Téngase en cuenta que los casos de uso es la forma en que los actores usan el sistema. Es decir, son fragmentos de funcionalidades que el sistema ofrece, para aportar un resultado de valor a sus actores o una secuencia de acciones que el sistema puede llevar a cabo al interactuar con estos. Incluye alternativas dentro de la secuencia.

2.5.2 Requerimientos No Funcionales

Los requerimientos no funcionales son propiedades o cualidades que el producto debe tener.

1. Apariencia o interfaz externa.

- 1.1 El sistema debe tener una interfaz fácil de usar y amigable para que pueda ser utilizada fácilmente por el usuario.
- 1.2 Empleo de imágenes y colores identificados con el negocio donde se implantará el sistema.

2. Usabilidad.

- 2.1 El sistema podrá ser utilizado por personas que tengan un conocimiento mínimo en el manejo de las computadoras.
- 2.2 El sistema estará disponible mientras exista conexión con el servidor central.
- 2.3 Las funcionalidades del sistema están visibles en un menú a la izquierda del sistema, lo que hace más fácil su distribución y localización por el usuario.

3. Rendimiento.

- 3.1 Debe mantener un diseño similar en todas las páginas, las cuales estén poco cargadas de imágenes, para garantizar una respuesta más rápida del sistema.

4. Soporte.

- 4.1 Se necesita un servidor de base de datos que soporte grandes volúmenes de datos (MySQL).
- 4.2 Al ser una aplicación web, será visible en navegadores web como Internet Explorer y Mozilla Firefox, los cuales son los más utilizados hoy en día.

5. Portabilidad.

- 5.1 El sistema será multiplataforma. Podrá ser visualizado en sistemas operativos tales como versiones de Windows (98 o superiores) y Linux.

6. Software.

6.1 Para el cliente:

- 6.1.1 Sistema Operativo Windows 98 o superior y Linux.
- 6.1.2 Navegador web Internet Explorer, o Mozilla Firefox.

6.2 Para el servidor:

- 6.2.1 Sistema operativo Windows Advancer Server (2000 o superior) o Linux en cualquiera de sus distribuciones.

6.2.2 Un servidor Apache v 2.2 con módulo PHP 5 disponible.

6.2.3 Un servidor de bases de datos MySQL 5.

7. Hardware.

7.1 Para Servidor:

7.1.1 Server de rack de 19 con:

7.1.2 NIC de 1 Gbit Ethernet.

7.1.3 Array scsi de 3 Hard Disk > 70 Gb.

7.1.4 Fuente redundante.

7.1.5 Certificación del fabricante para correr Linux (Debian Sarge).

7.1.6 1 Gbyte RAM.

7.2 Para estaciones de trabajo:

7.2.1 Procesador 1.60 GHz o superior.

7.2.2 256 MB de memoria; a más memoria mayor rendimiento.

7.2.3 Monitor VGA o superior.

7.2.4 Ratón Microsoft o compatible.

8. Seguridad:

8.1 Se garantiza el acceso controlado a la información. Esto influye en la presentación de las interfaces a cada usuario, en dependencia del nivel de acceso que le será conferido.

8.2 Disponer de un mecanismo de seguridad basado en el modelo de SAAA (Single Authentication, Authorization and Accounting).

8.3 Confiabilidad: La información manejada por el sistema está protegida de acceso no autorizado.

8.4 Integridad: La información que en el sistema se gestiona debe mantenerse de forma íntegra.

8.5 Disponibilidad: Se utiliza el servicio de autenticación para dar acceso a los usuarios que interactúan con el sistema. Sólo los usuarios autorizados tienen acceso a la información en todo momento.

9. Mantenimiento.

9.1 El sistema debe estar bien documentado de forma tal que el tiempo de mantenimiento sea mínimo en caso de necesitarse.

2.6 Definición de los casos de uso del sistema

La representación gráfica del modelo de casos de uso del sistema es la interacción del sistema con el usuario o con otro sistema para conseguir un objetivo específico. Se describe el modelamiento de casos de uso del sistema, donde se explica la generalización de los actores del sistema.

2.6.1 Actores del Sistema

ACTOR	DESCRIPCIÓN
Usuario.	Este actor puede ser editor y visualizador y tiene la función de autenticarse en el sistema. De acuerdo al nivel de acceso que tenga se le dan los permisos necesarios.
Editor.	Este actor puede ser secretario docente o editor nacional y tiene los permisos de insertar, eliminar y modificar datos en el sistema.
Secretario Docente.	Este actor es el encargado de realizar la matrícula al programa de formación posgraduada de las especialidades. Gestiona las evaluaciones y registra los movimientos de acuerdo a las causas expuestas. Antes de realizar cualquiera de estas operaciones procede a una búsqueda del profesional. Es un tipo de editor que trabaja a un nivel provincial o lo que es lo mismo a nivel de facultades.
Editor Nacional.	Este actor se encarga de adicionar, eliminar y modificar los codificadores referentes a las especialidades, gestiona los cursos y ediciones y le hace los cambios de acuerdo a las exigencias del MINSAP.
Visualizador.	Este actor tiene la función de ver los informes de acuerdo a los diferentes niveles de acceso que pueden ser: nacional, provincial y municipal.

2.6.2 Generalización de actores del sistema

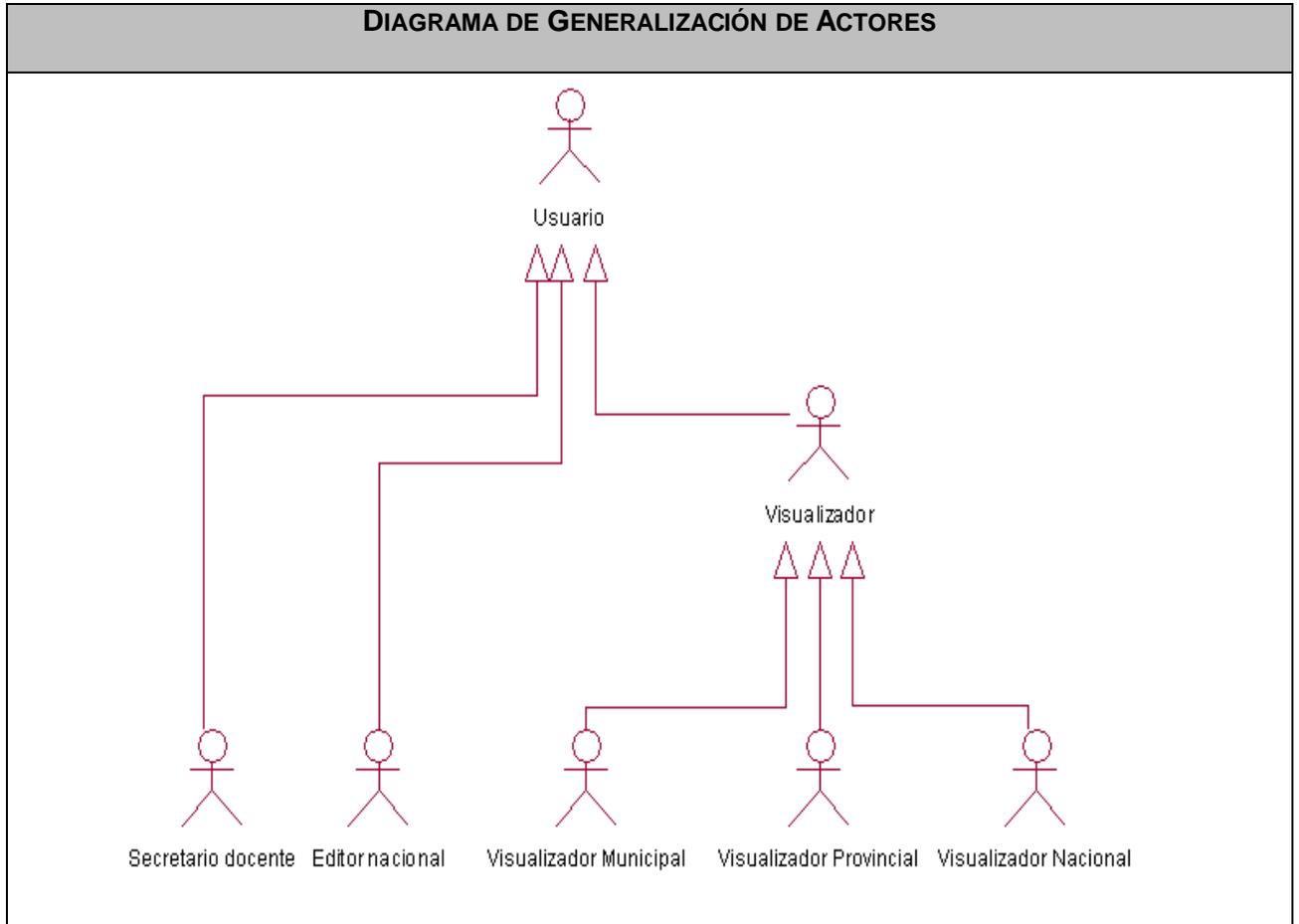


Fig 2.1 Diagrama de generalización de actores

2.6.3 Casos de Uso del Sistema

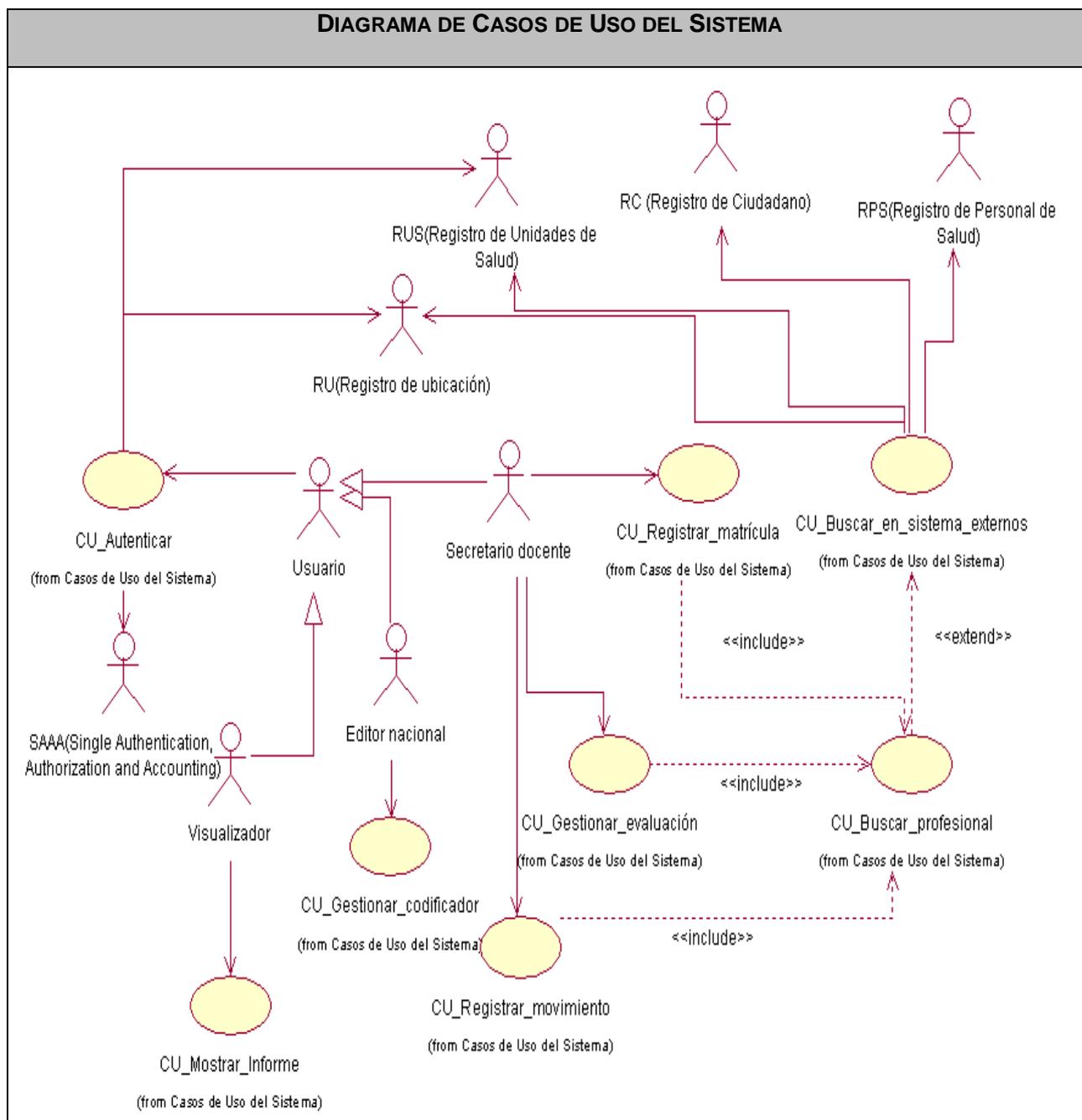


Fig 2.2 Diagrama de casos de uso del sistema

2.6.4 Descripción de los Casos de Uso

Caso de Uso.	CU_Autenticar.	
Actores.	Usuario.	
Propósito.	Gestionar el proceso de autenticarse en el sistema.	
Resumen.	El caso de uso se inicia cuando el usuario desea acceder al sistema. Para ello debe autenticarse mediante el SAAA donde debe estar previamente registrado, El RUS se utiliza para ubicar al usuario en su centro correspondiente. Finaliza cuando el usuario se autentica correctamente y el sistema lo envía a la interfaz en correspondencia con el rol que desempeñe.	
Pre-condiciones.	El usuario está registrado en el sistema.	
Referencias.	RF1.	
Prioridad.	Crítico.	
Flujo Normal de Eventos.		
Acción del Actor	Respuesta del Negocio	
1. El usuario escribe en el formulario nombre de usuario y contraseña correspondiente para acceder al sistema.	1.1 El sistema comprueba la validez de los datos y utiliza el SAAA, RU y el RUS para asignar los permisos al usuario según el rol que este tenga, ubicándolo en su centro correspondiente. 1.2 El sistema da acceso al usuario.	
Flujos Alternos.		
3.1 Si los datos no son correctos el sistema muestra mensajes de error.		
Pos-condiciones	El usuario accede al sistema.	

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

Caso de Uso.	CU_Buscar_profesional.
Actores.	CU_Registrar_matrícula, CU_Gestionar_evaluación y CU_Registrar_movimiento.
Propósito.	Buscar por diferentes criterios a un profesional determinado.
Resumen.	El Caso de Uso se inicia cuando se desea registrar la información de algún profesional, insertar alguna evaluación ó realizar algún trámite de movimiento. Para ello, primeramente se debe buscar ese profesional. Finaliza el caso de uso una vez que se encuentra el profesional.
Pre-condiciones.	El secretario docente autenticado satisfactoriamente por el sistema. El profesional debe estar registrado en el sistema.
Referencias.	RF2.
Prioridad.	Crítico.
Flujo Normal de Eventos.	
Acción del Actor	Respuesta del Negocio
<p>2. El secretario docente debe llenar alguno de los campos para buscar al profesional: nombre, apellidos y sexo. Luego pulsa el botón “Buscar”. En caso de no encontrar información, procede a la “Búsqueda avanzada”.</p> <p>3. El secretario docente selecciona el profesional que desea, haciendo clic en el hipervínculo del Nombre y Apellidos.</p>	<p>1.1 El sistema muestra la página de búsqueda.</p> <p>2.1 El sistema muestra un listado de los profesionales que cumplan con los criterios de búsqueda. Se muestra el código del profesional, sexo, nombre, apellidos, provincia y municipio.</p> <p>3.1 El sistema muestra la página correspondiente que invocó la búsqueda.</p>
Flujos Alternos	
2.1. Si no se encuentra profesional con los criterios de búsqueda deseados, el sistema muestra un mensaje indicando que no existe ningún profesional con estos datos.	
Pos-condiciones.	Se busca el profesional.

Caso de Uso.	CU_Registrar_matrícula.
---------------------	-------------------------

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

Actores.	Secretario docente.	
Propósito.	Registrar un nuevo profesional al sistema.	
Resumen.	El Caso de Uso se inicia cuando el secretario docente necesita registrar en el sistema los datos de un profesional. En una especialidad se registran todos los datos del mismo: datos generales, de la carrera y de la especialidad a la que se va matricular, mediante el uso de Servicios Web. Finaliza el caso de uso una vez registrados todos los datos. El profesional pasa a formar parte de la matrícula de una especialidad.	
Pre-condiciones.	El secretario docente autenticado satisfactoriamente por el sistema.	
Referencias.	RF11, 12.	
Prioridad.	Crítico.	
Flujo Normal de Eventos.		
Acción del Actor	Respuesta del Negocio	
<ol style="list-style-type: none"> 1. El secretario docente selecciona la opción "Matricular". 2. El secretario llena los campos que se encuentren vacíos. Marca la siguiente pestaña para continuar el registro del profesional. 3. El secretario llena los campos. Marca la siguiente pestaña para continuar con el registro del profesional. 4. El secretario llena los campos. Pulsa el botón "Registrar" para terminar la operación. 	<p>El sistema primero hace una búsqueda de este profesional. Si se encuentra en la base de datos del RPS sólo se debe actualizar en los campos sus datos y llenar los que no se recogen ahí.</p> <p>1.1 El sistema muestra el formulario "Datos generales" para comenzar la captación de datos.</p> <p>2.1 El sistema muestra el formulario "Datos de la carrera" para llenar estos datos del profesional.</p> <p>3.1 El sistema muestra el formulario "Datos de la especialidad" para llenar estos datos del profesional (es aquí donde se recogen los datos de la especialidad donde se va a matricular)</p>	

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

	4.1 El sistema registra al profesional en la especialidad; y muestra la confirmación.
Flujos Alternos.	
2.1, 3.1, 4.1. Si cometió algún error al llenar los campos del formulario anterior, puede regresar a la pestaña pulsando la misma y ahí están fijos los datos.	
Pos-condiciones.	Se registran los datos del profesional.

Caso de Uso.	CU_Gestionar_evaluación.
Actores.	Secretario docente.
Propósito.	Gestionar las evaluaciones de los profesionales.
Resumen.	El Caso de Uso se inicia cuando el encargado en la Secretaría docente necesita gestionar las evaluaciones en las especialidades de los profesionales en el sistema. Se registran todas las notas del mismo y los programas asociados a cada especialidad: nota de la tarjeta, examen teórico, examen práctico, Trabajo de Terminación de la Especialidad y la nota final y los programas (cursos, estancias y rotaciones). Finaliza el caso de uso una vez registrados todas las evaluaciones.
Pre-condiciones.	El profesional ya posee las evaluaciones pertinentes a una especialidad determinada.
Referencias.	RF13, 14.
Prioridad.	Crítico.
Flujo Normal de Eventos.	
Acción del Actor	Respuesta del Negocio
1 El secretario docente selecciona la opción "Registrar evaluaciones al profesional".	1.1 El sistema muestra la página de "exámenes de la especialidad y/o Asignar programa" para comenzar la captación de datos.
2 El secretario llena los campos de las evaluaciones.	
3 El secretario pulsa el botón "Registrar" para terminar la operación.	2.1 El sistema muestra las evaluaciones del

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

	<p>profesional en una especialidad específica.</p> <p>3.1 El sistema inserta las evaluaciones de un profesional para una especialidad determinada.</p>
Flujos Alternos.	
1.1, 2.1, 3.1 Si cometió algún error al llenar los campos de la página anterior puede regresar a esta pulsando el botón “Atrás”.	
Pos-condiciones	Se registran las evaluaciones del profesional.

Caso de Uso.	CU_Registrar_movimiento.
Actores.	Secretario docente (inicia).
Propósito.	Registrar el proceso relacionado con las bajas y traslados de los estudiantes.
Resumen.	Este caso de uso se encarga de todo el negocio relacionado con los movimientos que realice un profesional durante su formación: bajas y traslados.
Pre-condiciones.	El secretario docente autenticado satisfactoriamente por el sistema.
Referencias.	RF15, 16, 17, 18, 19.
Prioridad.	Crítico.
Flujo Normal de Eventos.	
Acción del Actor	Respuesta del Negocio
<p>1 El secretario docente selecciona la opción “Registrar movimiento al profesional” del menú principal de trabajo.</p> <p>2 El secretario docente selecciona la pestaña deseada de los tipos de movimientos: “Traslado”</p>	<p>Para realizar un movimiento a un profesional se debe buscar primero el profesional.</p> <p>1.1 El sistema muestra la página de movimientos con los datos generales del Profesional y las pestañas con los datos de los tipos de movimientos.</p>

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

<p>“Baja Definitiva”</p> <p>“Baja Temporal”</p>	
Sección “Traslado”	
<p>3. El secretario docente debe llenar todos los campos y pulsar el botón “Aceptar”.</p>	<p>2.1 El sistema muestra los campos a llenar: el tipo de traslado, el municipio, la provincia, fecha y las causas por las cuales se realiza este movimiento.</p> <p>3.1 El sistema ejecuta la acción y muestra un mensaje comunicando que el movimiento fue realizado satisfactoriamente.</p>
Sección “Baja Definitiva”	
<p>4. El secretario docente llena los campos y pulsa el botón “Aceptar”.</p>	<p>4.1 El sistema muestra los campos a llenar: fecha y causas expuestas por las cuales desea realizar su movimiento.</p> <p>5.1 El sistema realiza la baja del profesional y muestra un mensaje comunicando que el movimiento fue realizado satisfactoriamente.</p>
Sección “Baja Temporal”	
<p>7. El secretario docente llena todos los campos y pulsa el botón “Aceptar”.</p>	<p>6.1 El sistema muestra los campos a llenar para este trámite: fecha, tiempo (en años), causas, fecha probable de la reincorporación y la fecha real de reincorporación.</p> <p>7.1 El sistema ejecuta la acción y muestra un mensaje comunicando que el movimiento fue realizado satisfactoriamente.</p>
Flujos Alternos.	
Pos-condiciones	Se realizó el movimiento al profesional.

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

Caso de Uso.	CU_Gestionar_codificador.	
Actores.	Editor Nacional (inicia).	
Propósito.	Insertar, modificar o eliminar datos de los codificadores.	
Resumen.	El Caso de Uso se inicia cuando el Editor Nacional desea adicionar, editar y/o eliminar los datos de algún codificador. Finaliza el caso de uso una vez realizada la actualización.	
Pre-condiciones.	Editor Nacional debe estar autenticado en el sistema.	
Referencias.	RF3, 4, 5, 6, 7, 8, 9, 10.	
Prioridad.		
Flujo Normal de Eventos.		
Acción del Actor	Respuesta del Negocio	
<p>1. El Editor Nacional selecciona la opción “Gestionar Especialidades o Configuración de Programas”. Se dividen en dos tipos de codificadores del menú principal de trabajo.</p> <p>2. El Editor Nacional indica el codificador que desee agregar, actualizar o eliminar y presiona encima de él en caso que desee modificarlo o eliminarlo.</p>	<p>1.1 El sistema muestra la página Gestionar Especialidades y/o Configurar Programas que contiene los codificadores del sistema. Ya sean las especialidades o los tipos de programas que contiene una especialidad.</p> <p>2.1 El sistema muestra la página con los datos del codificador seleccionado (campos para introducir datos del codificador y listado de los mismos) y las opciones de adicionar, editar o eliminar, a través de los botones “Agregar”, “Editar” y “Eliminar” respectivamente.</p> <p>a) Si desea adicionar datos al codificador ir a sección “Agregar”.</p> <p>b) Si desea editar datos del codificador ir a sección “Listar”.</p> <p>c) Si desea eliminar datos al codificador ir a sección “Eliminar”.</p>	
Sección “Adicionar”		
3. El Editor Nacional llena los campos con	3.1 El sistema inserta los nuevos datos e	

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

los nuevos datos del codificador y pulsa el botón “Adicionar”.	inmediatamente los muestra en el listado de datos del codificador que aparece en la página.
Sección “Editar”	
4. El Editor Nacional debe marcar la fila correspondiente a los datos que desee modificar.	4.1 El sistema carga estos datos en los campos de edición.
5. El Editor Nacional realiza los cambios pertinentes.	5.1 El sistema muestra un mensaje de confirmación de la operación a realizar.
6. El Editor Nacional acepta la confirmación.	6.1 El sistema actualiza los datos, e inmediatamente los muestra en el listado de datos del codificador que aparece en la página.
Sección “Eliminar”	
7. El Editor Nacional debe marcar la fila correspondiente a los datos que desee eliminar y pulsar el botón “Eliminar”.	7.1 El sistema muestra un mensaje de confirmación de la operación a realizar.
8. El Editor Nacional acepta la confirmación.	8.1 El sistema elimina los datos e inmediatamente se dejan de mostrar en el listado de datos del codificador.
Flujos Alternos.	
Pos-condiciones	Se realiza la actividad para el codificador especificado.

En este capítulo se ha representado la información que debe manejar el sistema. Se identificaron los procesos y la justificación de cómo, cada uno de los actores, los lleva a cabo. Después de analizar los procesos de negocios y las características del sistema, se evidencia la necesidad de desarrollar un módulo para que sea integrado a un sistema basado en la Gestión de la Información en el Proceso de la Formación Posgraduada. Se aplican las tecnologías web de avanzada, para darle una solución factible a la problemática planteada.

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

En el capítulo anterior se definieron las funcionalidades necesarias del módulo que se propone. Se hace necesario definir cómo se desarrolla. Este capítulo está dedicado a la etapa de desarrollo donde se profundiza en el análisis del sistema, utilizando un lenguaje apropiado como UML para el entendimiento del mismo. Luego se da paso al diseño del sistema, etapa en la cual ocurre un refinamiento del análisis realizado.

En esta etapa se definen diagramas de clases del análisis de la aplicación. Se especifican qué clases de análisis toman parte del caso de uso y las relaciones entre ellas clasificándolas en clases interfaz, controladoras y entidad. Posteriormente en la etapa de diseño a través de los diagramas de interacción, se muestra una descripción gráfica de la interacción entre los actores y el sistema. Además se describen las clases que se representan en el diseño. Igualmente se muestra el Diagrama Modelo de Datos y la descripción de sus tablas.

3.1 Modelo de Análisis

En esta etapa se refinan y estructuran los requisitos obtenidos con anterioridad. Se profundiza en el dominio de la aplicación, lo que permite mayor comprensión del problema para modelar la solución. Los diagramas representados están organizados por paquetes para la comprensión de los mismos.

3.1.1 Diagramas de clases del análisis

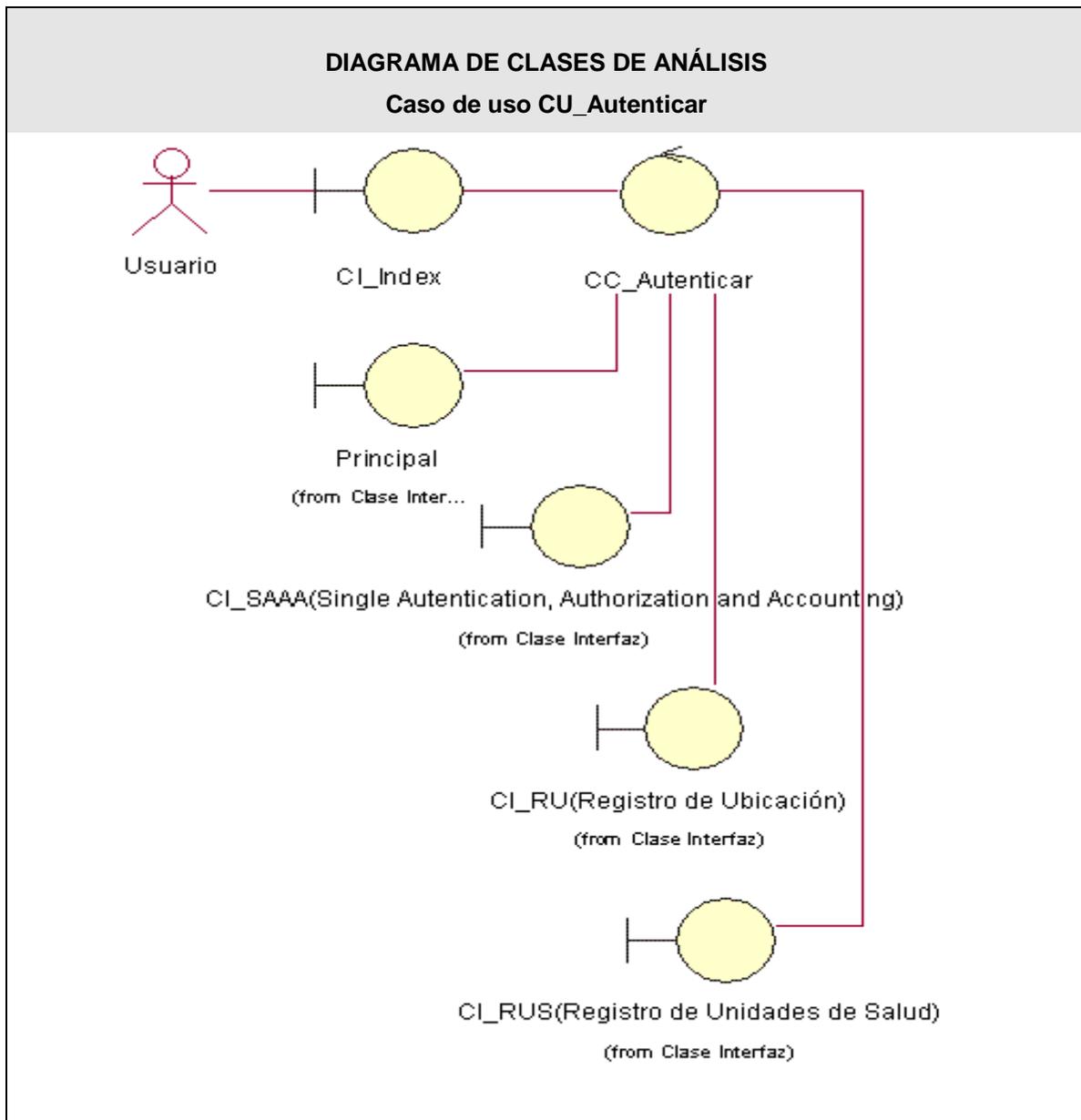


Fig. 3.1 Diagrama del Casos de uso CU_Autenticar

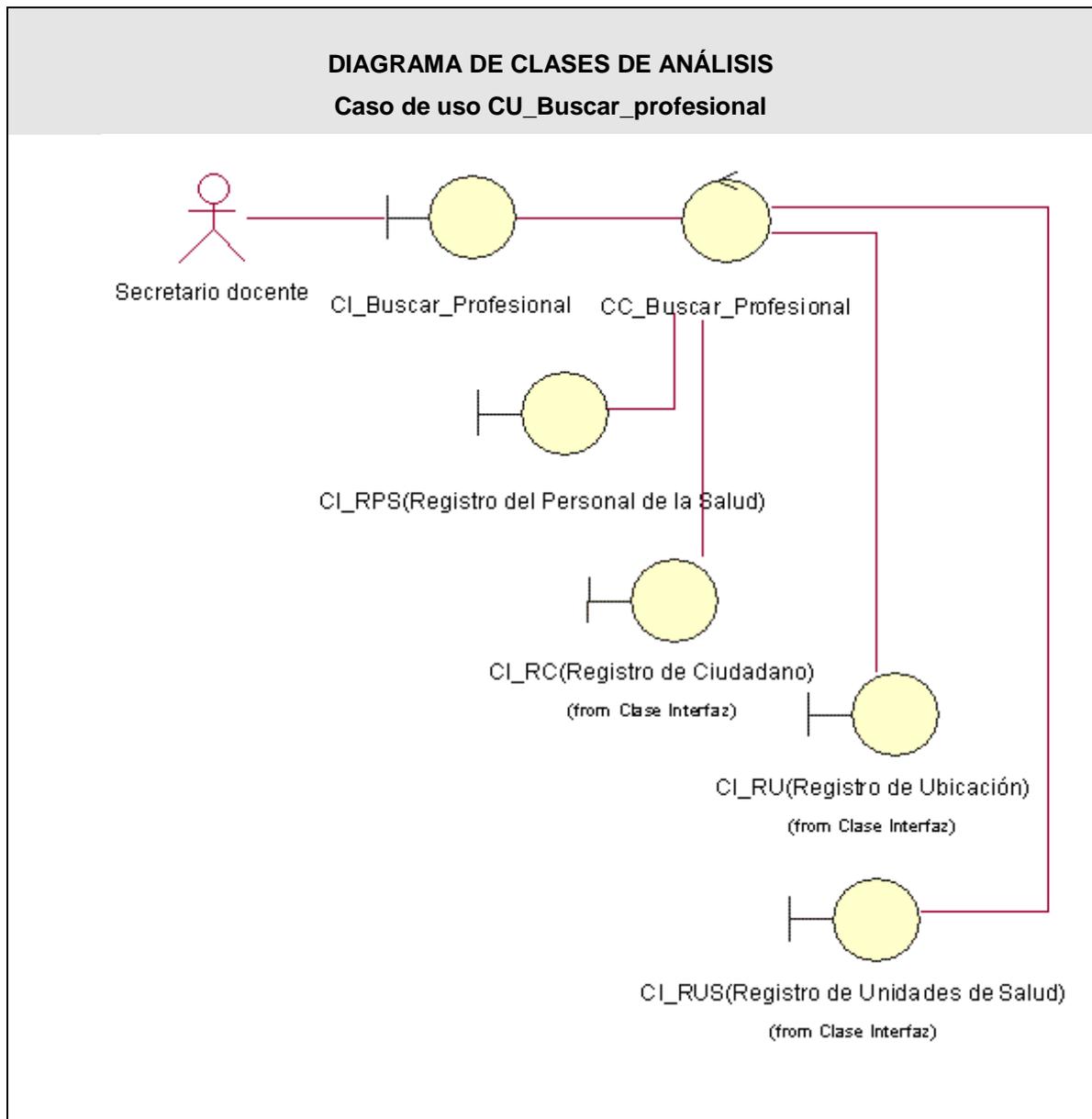


Fig. 3.2 Diagrama del Casos de uso CU_Buscar_profesional

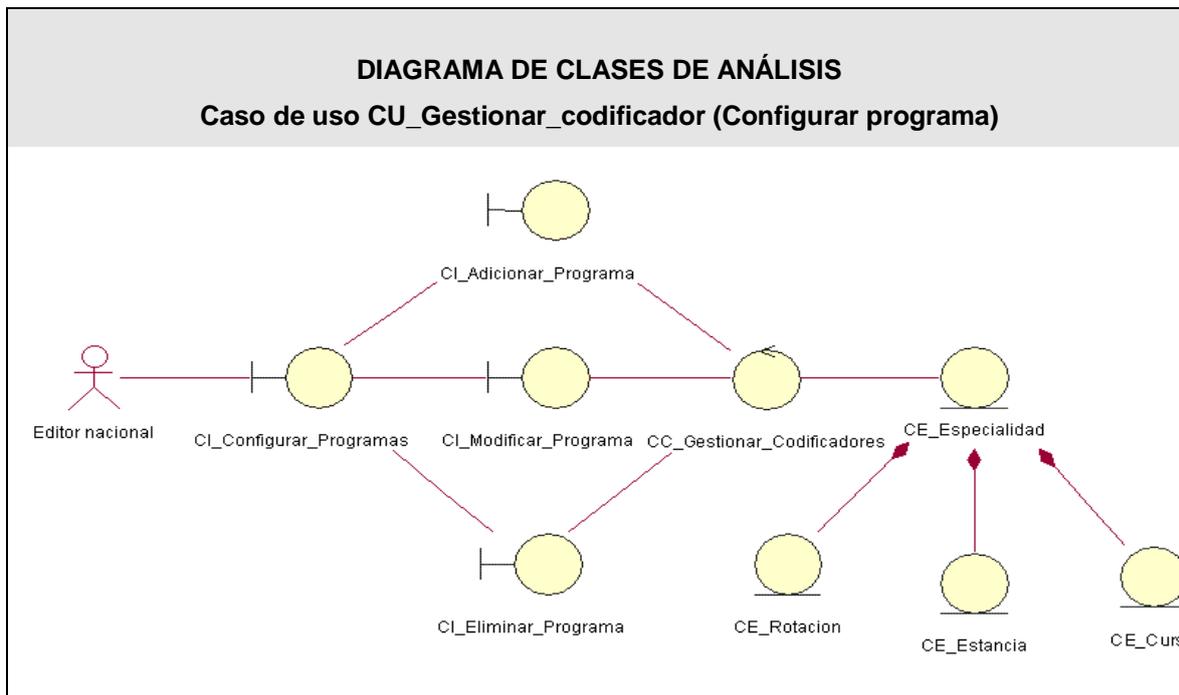


Fig. 3.3 Diagrama del Casos de uso CU_Gestionar_codificador

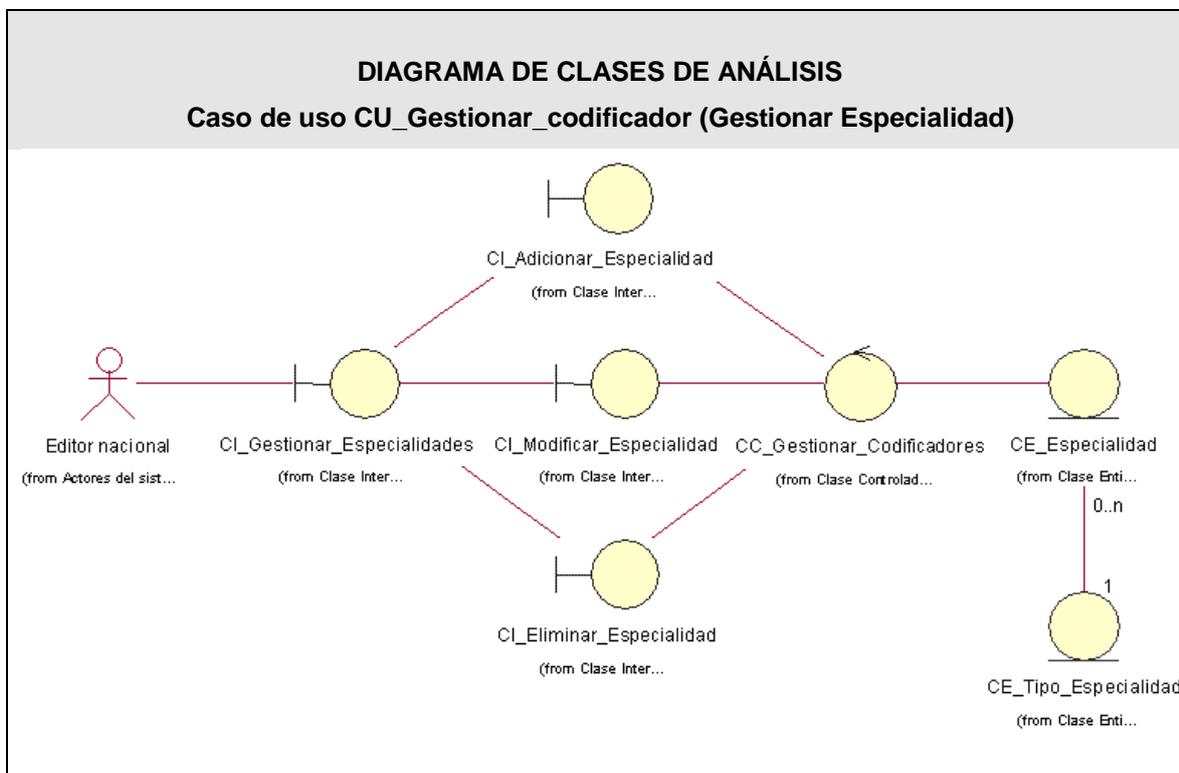


Fig. 3.4 Diagrama del Casos de uso CU_Gestionar_codificadores

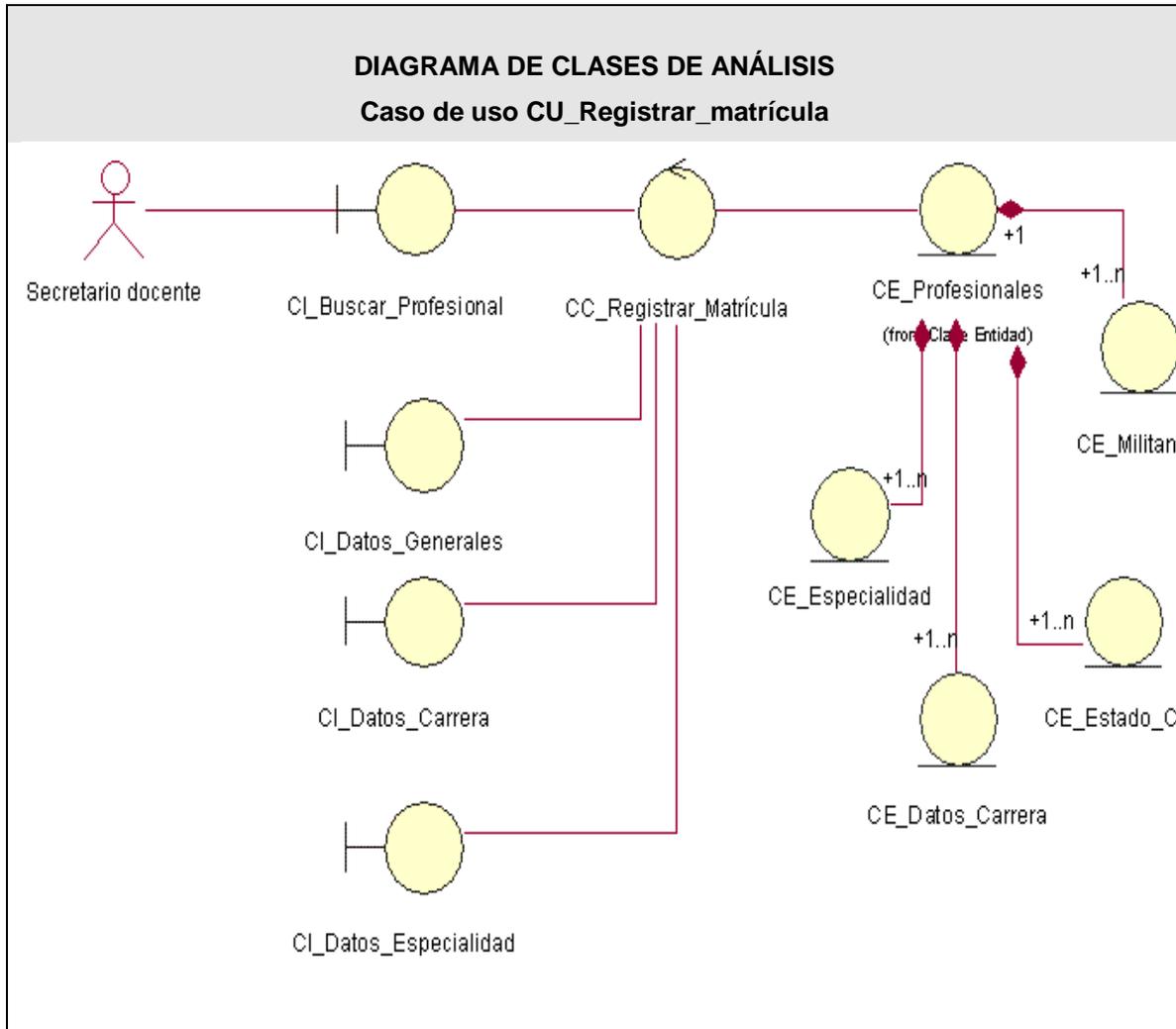


Fig. 3.5 Diagrama del Casos de uso CU_Registrar_matrícula

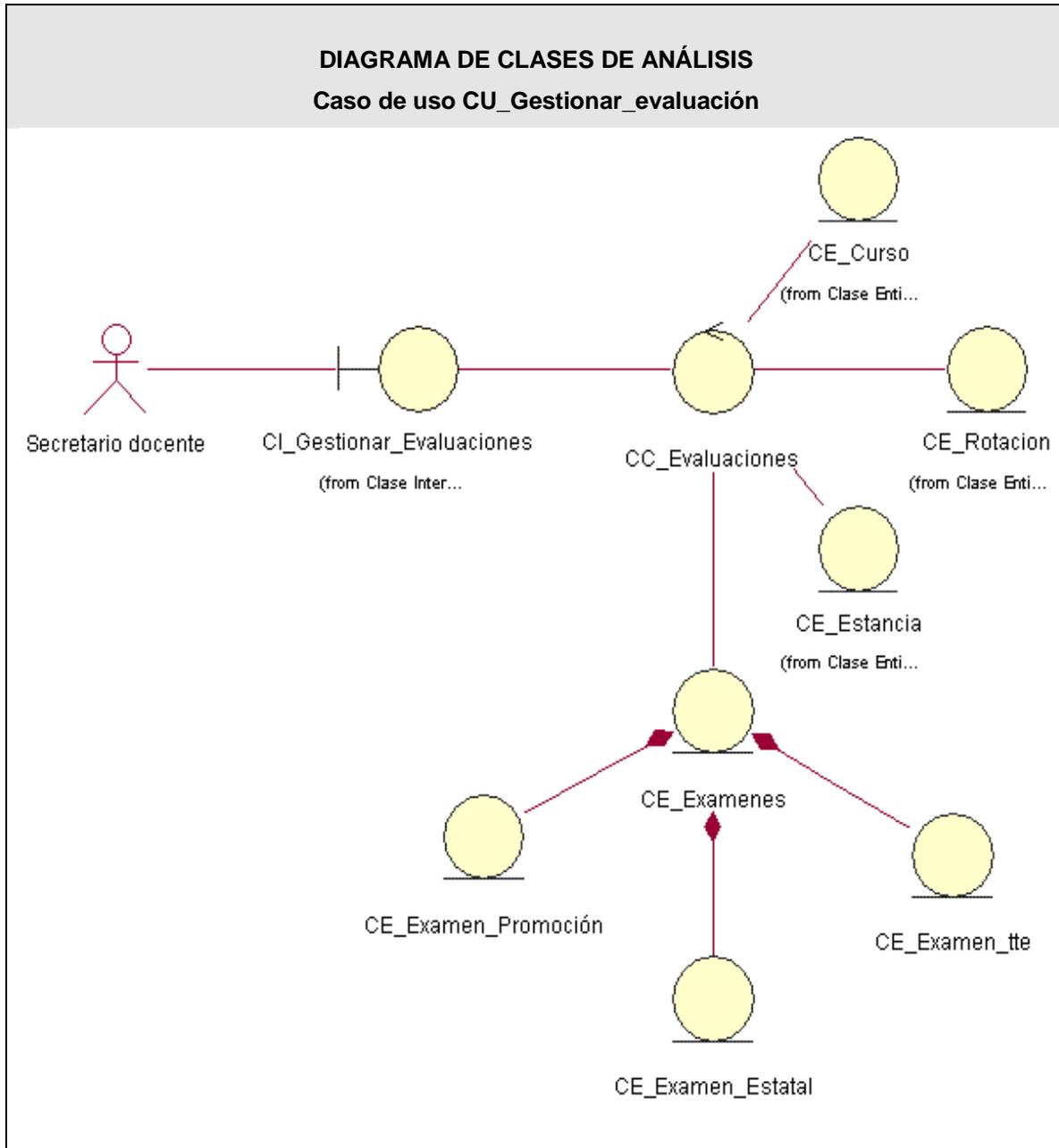


Fig. 3.6 Diagrama del Casos de uso CU_Gestionar_evaluación

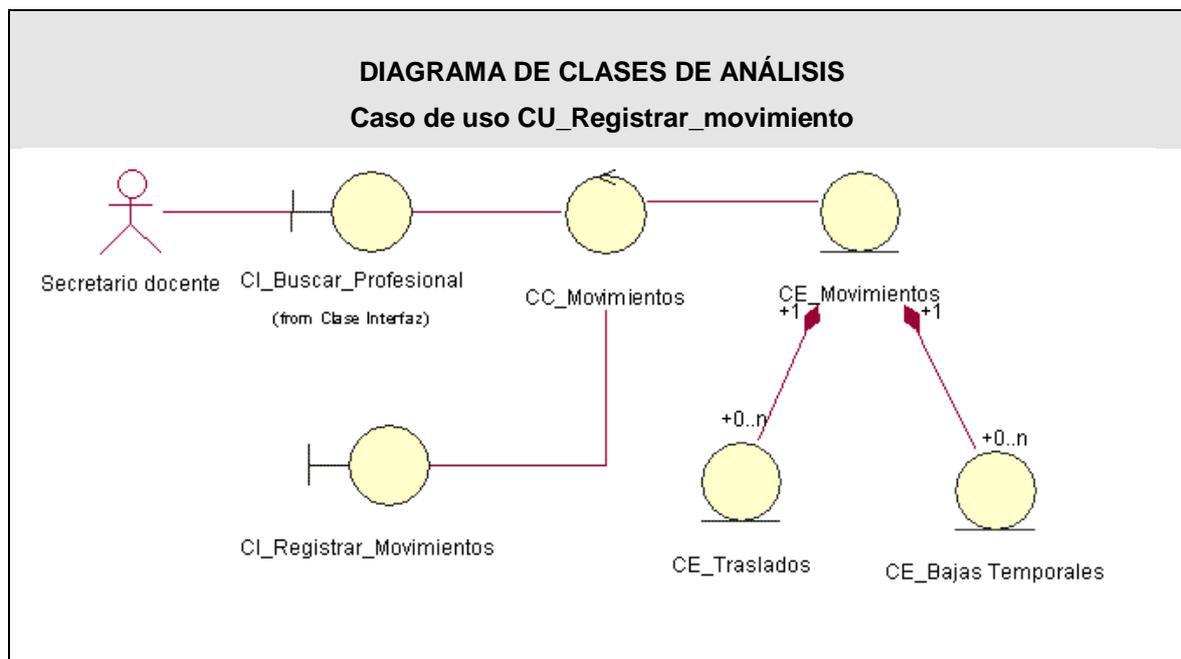


Fig. 3.7 Diagrama del Casos de uso CU_Registrar_movimiento

3.2 Modelo de Diseño

Esta etapa tiene como propósito modelar el sistema y encontrar su forma para que soporte todos los requisitos. Crea una entrada apropiada y un punto de partida para la implementación. El artefacto más importante que se obtiene es el modelo de diseño en el cual se desarrollan los casos de uso del sistema mediante diagramas de clases y de interacción.

3.2.1 Diagramas de clases de diseño

El modelo de diseño es el artefacto fundamental que se desprende del flujo de trabajo Análisis y Diseño y constituye una abstracción de la implementación. Es preciso hacer un buen diseño para tener un buen producto que satisfaga las necesidades del cliente y que sea un lenguaje fácil de interpretar para el implementador que convertirá el diseño en código. En estos diagramas muestran la relación entre las clases de cada caso de uso desarrollado.

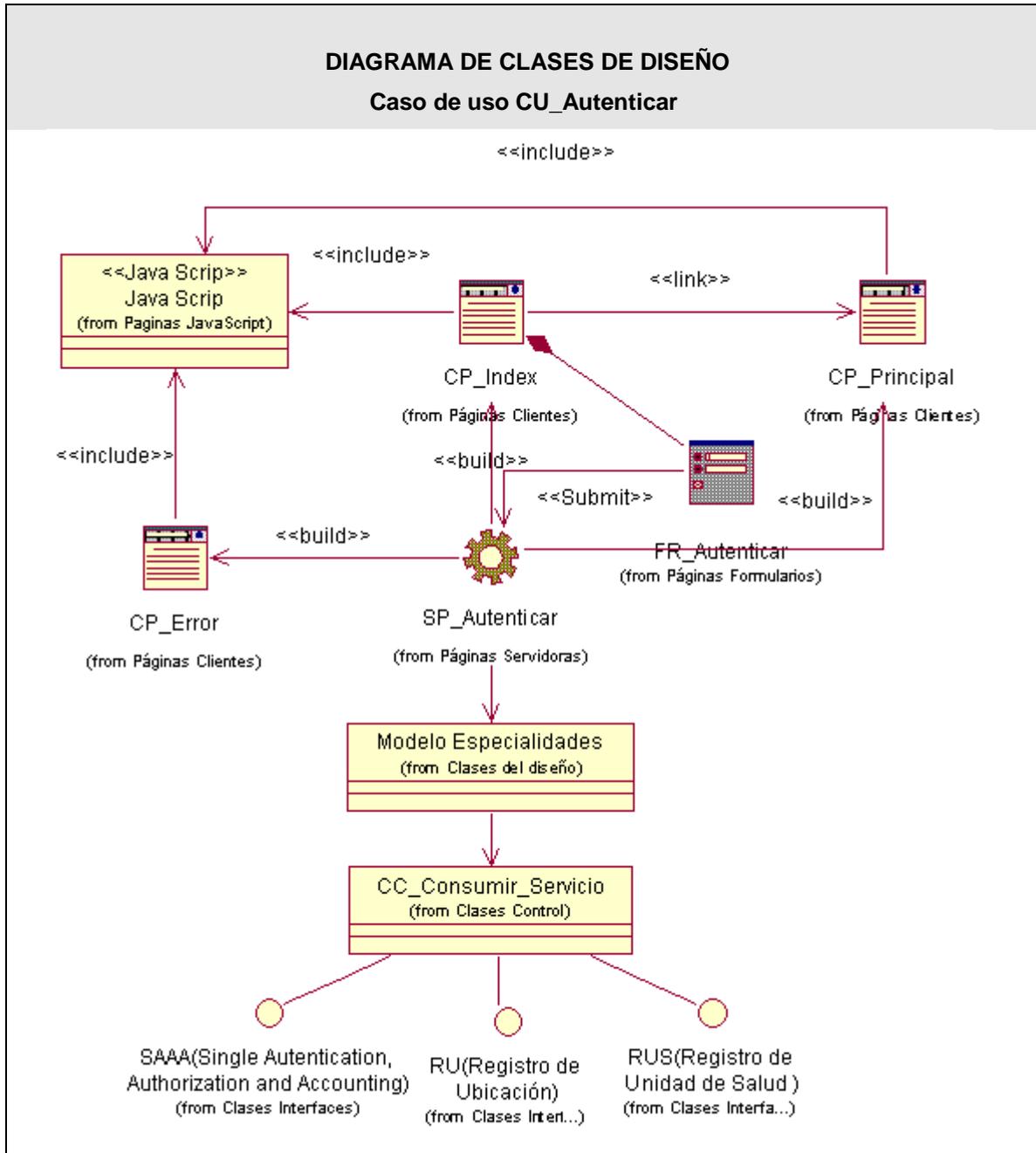


Fig. 3.7 Diagrama de clases del diseño CU_Autenticar

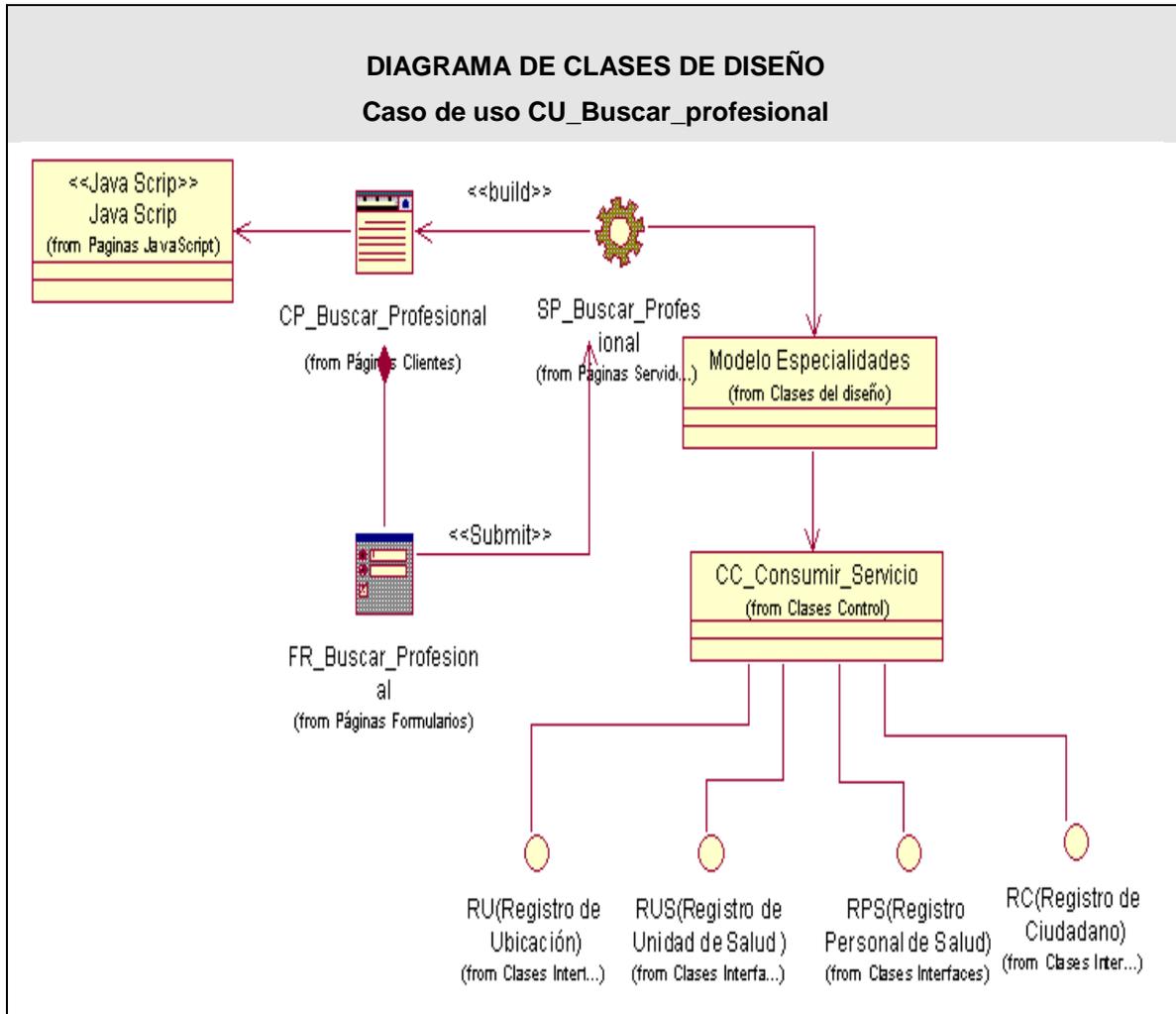


Fig. 3.8 Diagrama de clases del diseño CU_Buscar_profesional

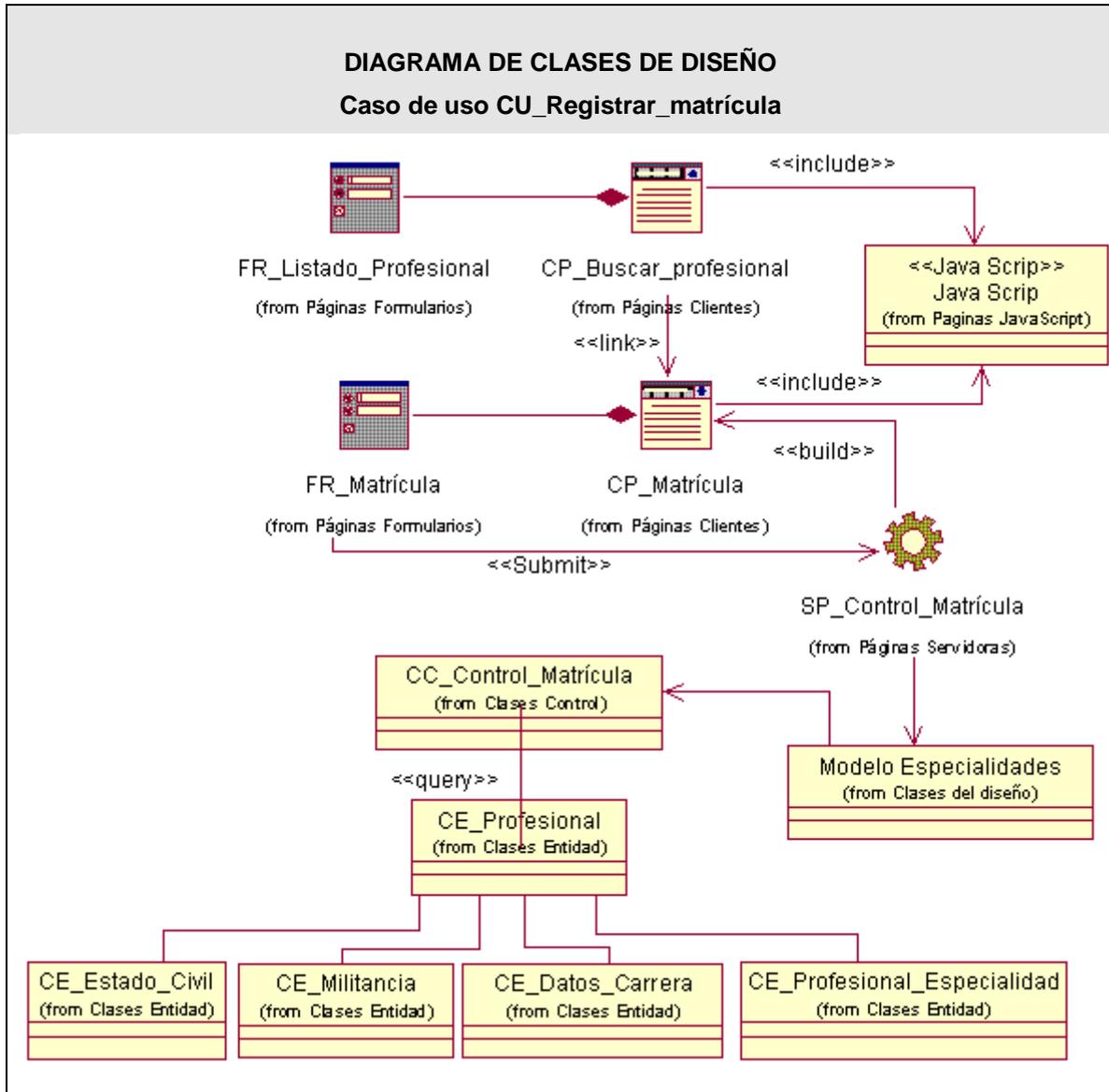


Fig. 3.9 Diagrama de clases del diseño CU_Registrar_matrícula

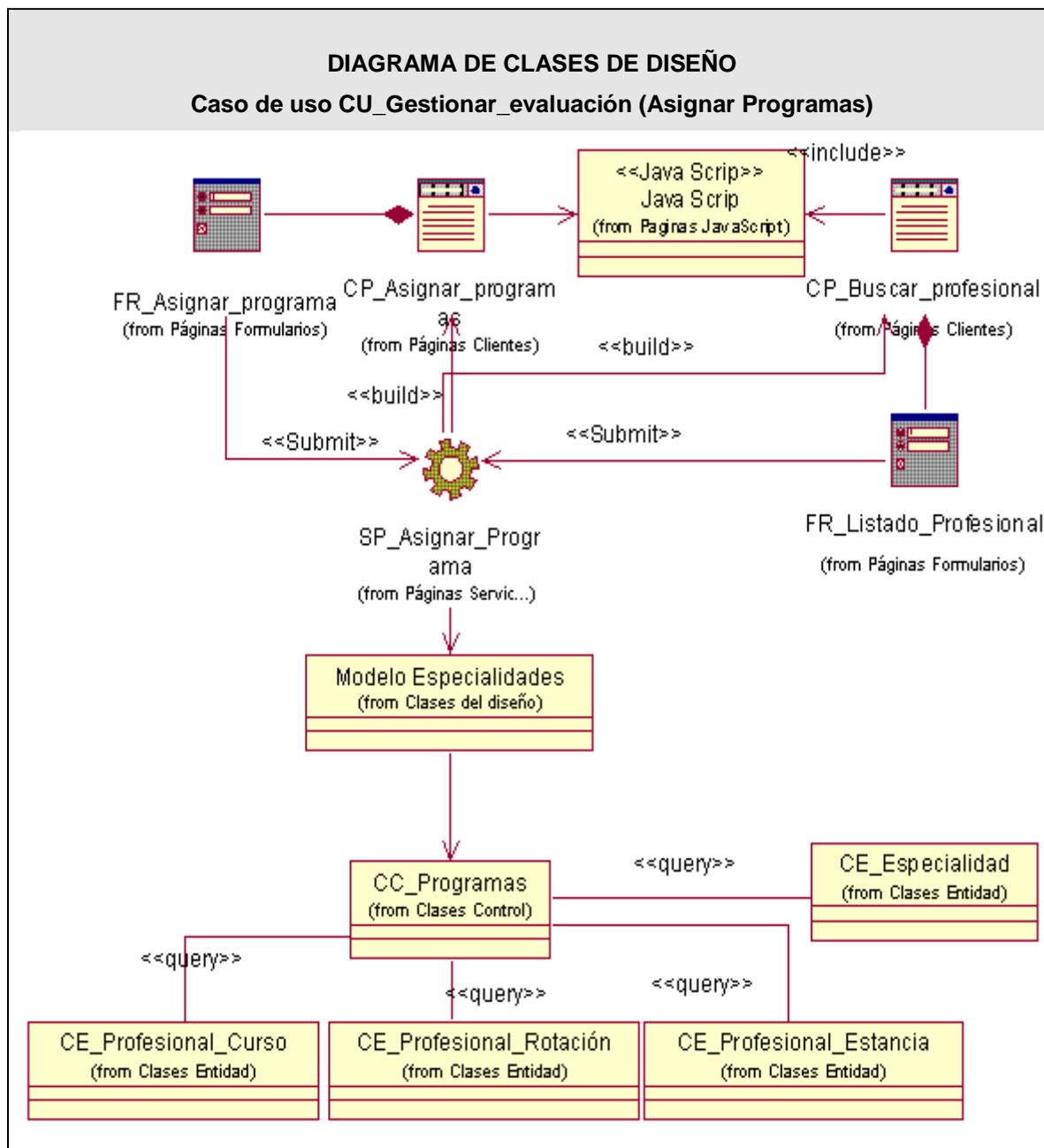


Fig. 3.10 Diagrama de clases del diseño CU_Gestionar_evaluación

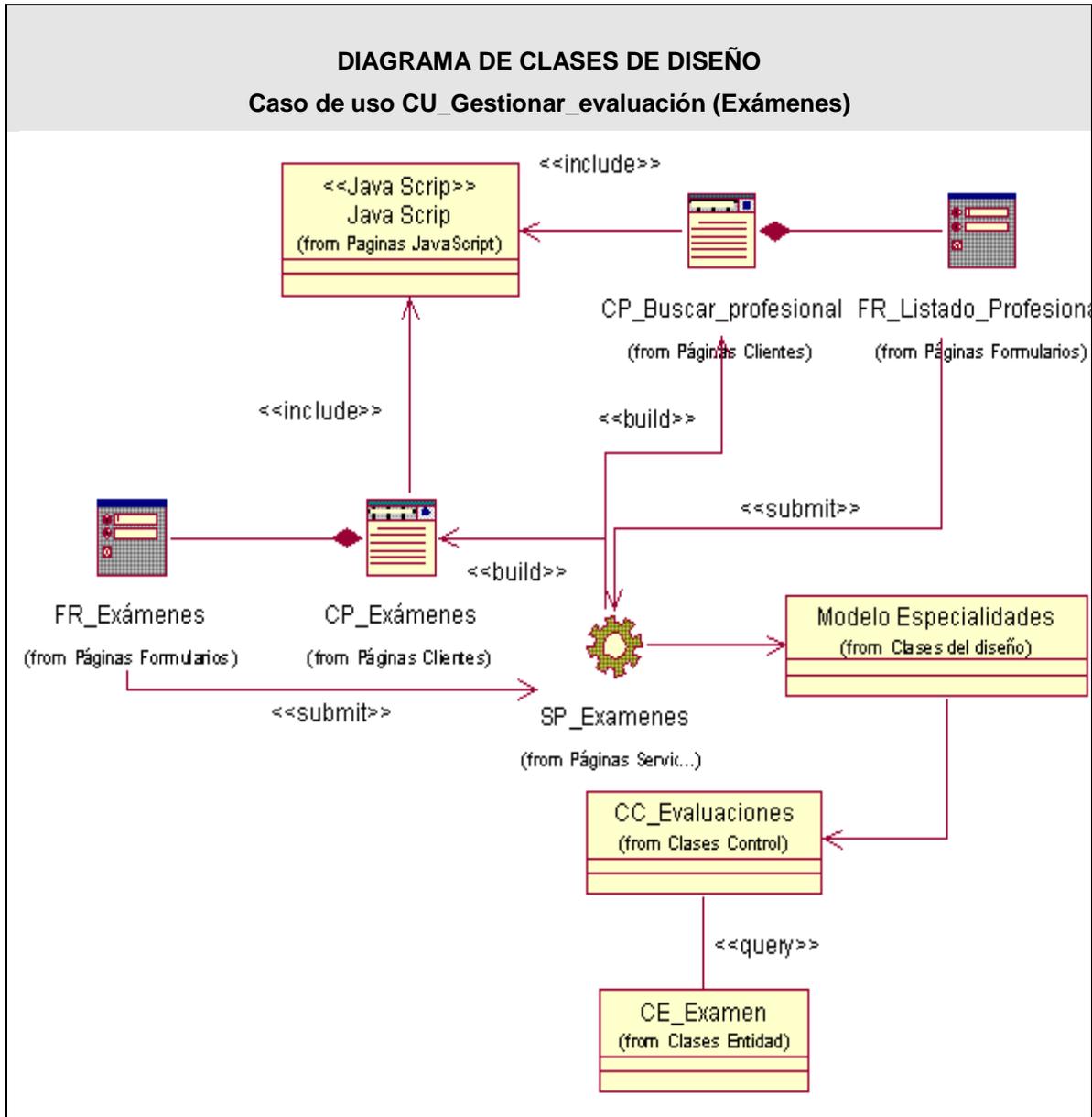


Fig. 3.11 Diagrama de clases del diseño CU_Gestionar_evaluación

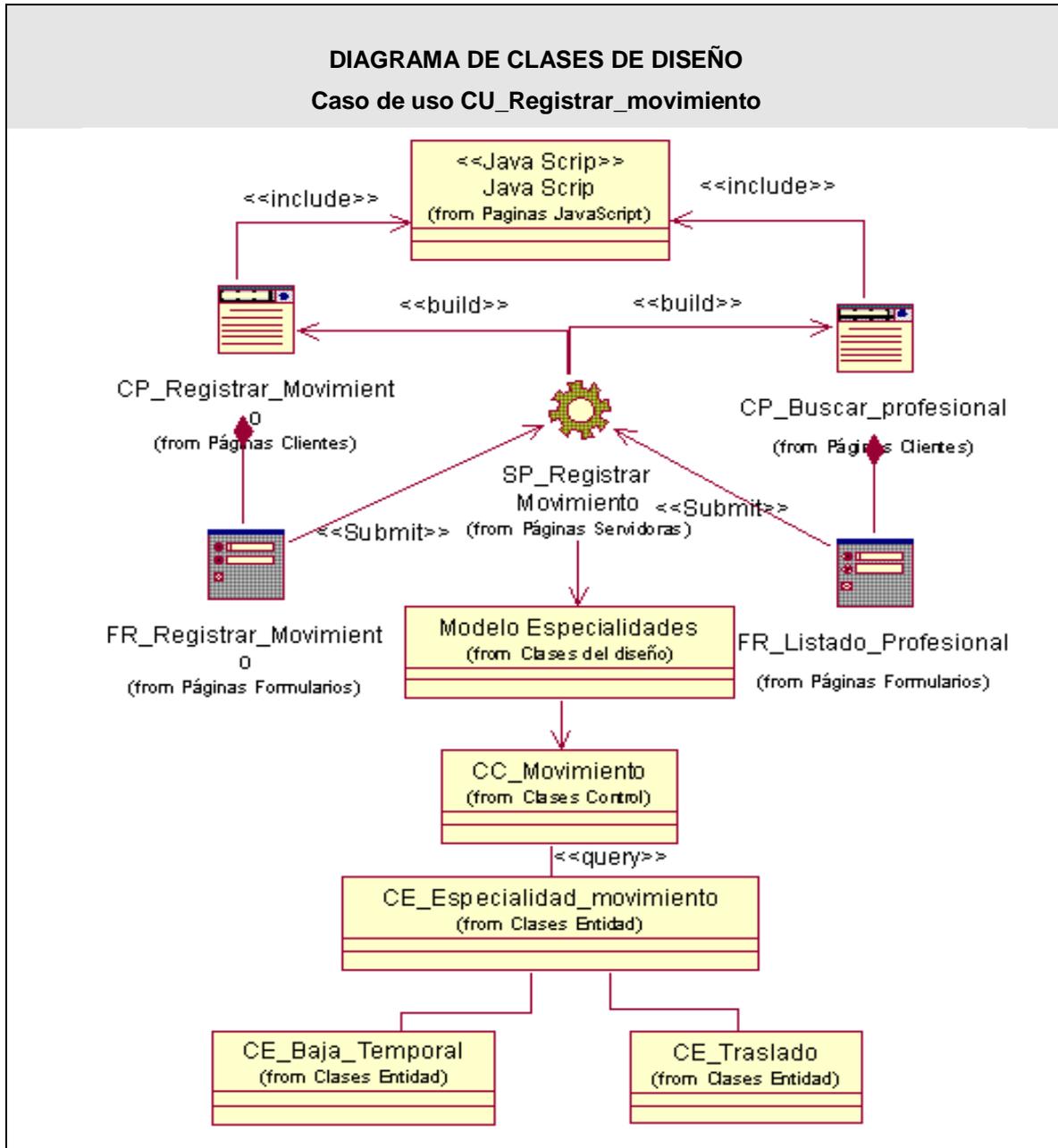


Fig. 3.12 Diagrama de clases del diseño CU_Registrar_movimiento

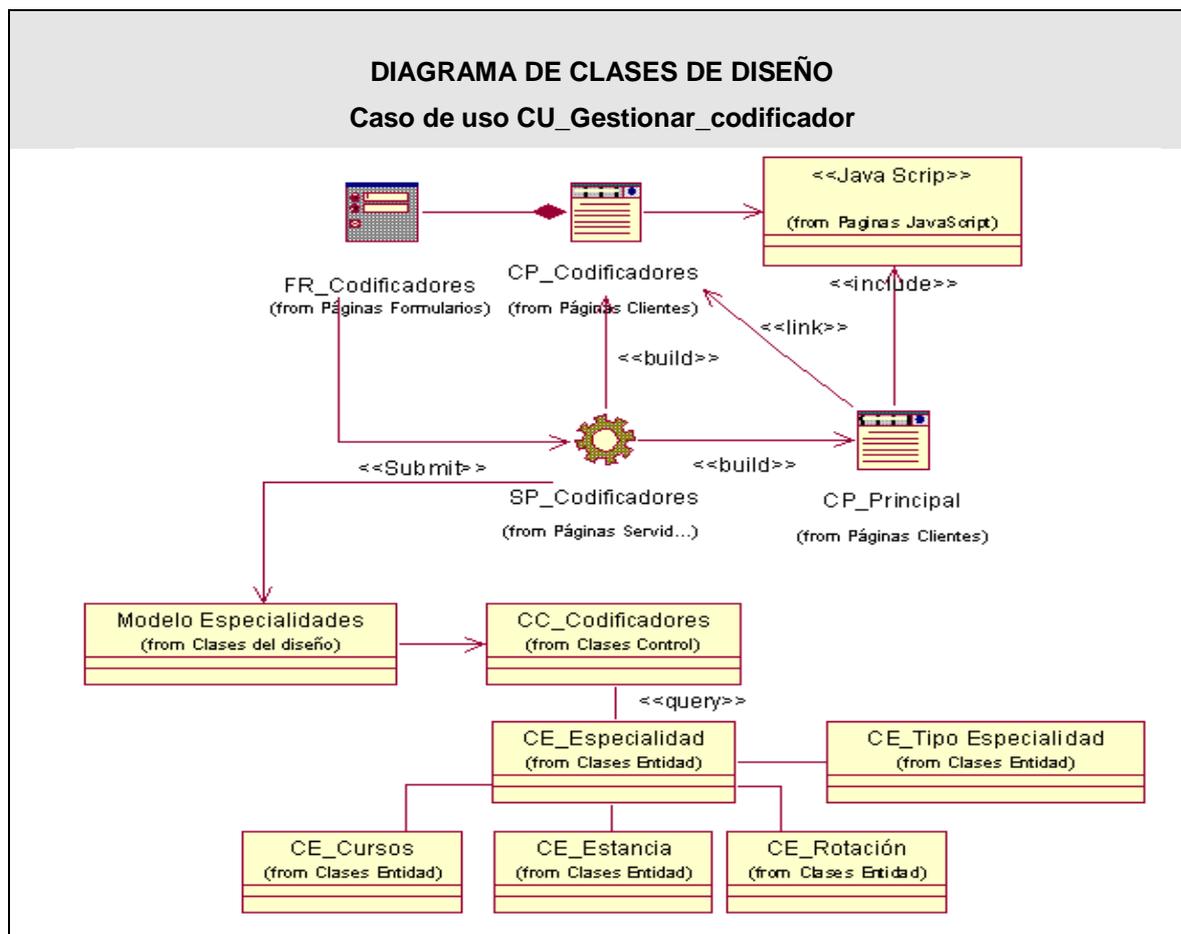


Fig. 3.13 Diagrama de clases del diseño CU_Gestionar_codificador

3.3 Descripción de las clases

Nombre: CC_Consumir_Servicio.	
Tipo de clase: Controladora.	
Atributo:	Tipo:
Para cada responsabilidad:	
Nombre:	Autentica.
Descripción:	Función para dar acceso al sistema según la responsabilidad del usuario.
Nombre:	Buscar_Ciudadano.
Descripción:	Función para buscar los datos necesarios de todo el personal que se encuentra en el RC.
Nombre:	Buscar_Personal_Salud.
Descripción:	Función para buscar los datos necesarios de todo el personal que se encuentra en el RPS.
Nombre:	Municipios_Provincias.

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

Descripción:	Función para obtener los municipios según la provincia.
Nombre:	Unidades_Salud.
Descripción:	Función para obtener las unidades de salud según el municipio y la provincia.
Nombre:	Grupos_Unidades_Salud.
Descripción:	Función para obtener los grupos de unidades de salud.
Nombre:	Sub_Grupos_Unidades_Salud.
Descripción:	Función para obtener los subgrupos de unidades de salud según los grupos unidades de salud.
Nombre:	Tipo_Unidades_Salud.
Descripción:	Función para obtener los tipos de unidad de salud.

Nombre: CC_Movimiento	
Tipo de clase: Controladora.	
Atributo:	Tipo:
Para cada responsabilidad:	
Nombre:	Mover.
Descripción:	Función para realizar los movimientos según el tipo (baja temporal, traslado, baja definitiva).

Nombre: CC_Control_Matricula	
Tipo de clase: Controladora.	
Atributo:	Tipo:
Para cada responsabilidad:	
Nombre:	registrar_medico.
Descripción:	Función para realizar la matrícula de un profesional en una especialidad determinada.
Nombre:	obtener_municipios.
Descripción:	Función para obtener los municipios según la provincia.
Nombre:	obtener_unidades_residencia.
Descripción:	Función para obtener las unidades residencia según el municipio.

Nombre: CC_Codificadores.	
Tipo de clase: Controladora.	
Atributo:	Tipo:
Para cada responsabilidad:	
Nombre:	agregar.
Descripción:	Función para adicionar una especialidad según el tipo.
Nombre:	listar_codificador.
Descripción:	Función para obtener el listado de todas las especialidades.

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

Nombre:	actualizar.
Descripción:	Función para modificar las especialidades.
Nombre:	eliminar_especialidad.
Descripción:	Función para eliminar las especialidades.

Nombre: CC_Programas.	
Tipo de clase: Controladora.	
Atributo:	Tipo:
Para cada responsabilidad:	
Nombre:	agregar_datos.
Descripción:	Función para adicionar un programa que pueden ser curso, estancia, rotaciones.
Nombre:	listado.
Descripción:	Función para mostrar un listado de todas las especialidades
Nombre:	Matricular_ciudadano
Descripción:	Función para asignarle un programa a un profesional.
Nombre:	eliminar_programa
Descripción:	Función para eliminar un programa seleccionado.

Nombre: CC_Evaluación.	
Tipo de clase: Controladora.	
Atributo:	Tipo:
Para cada responsabilidad:	
Nombre:	registrar_exámenes.
Descripción:	Función para adicionarles los tipos de exámenes y las notas a los profesionales.
Nombre:	registrar_programas.
Descripción:	Función para adicionarles los programas (cursos, estancias, rotaciones) a los profesionales.

Nombre: CE_Tipo_Especialidad.	
Tipo de clase: Entidad.	
Atributo:	Tipo:
nombre_tipo_especialidad.	STRING
idtipo_especialidad.	INTEGER
Para cada responsabilidad:	
Nombre:	
Descripción:	
Nombre: CE_especialidad_movimiento	
Tipo de clase: Entidad	
Atributo:	Tipo:
idespecialidad_movimiento	INTEGER

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

causas	STRING
fecha	DATETIME
idprofesional	INTEGER
idespecialidad	INTEGER
tipo_baja	INTEGER
Para cada responsabilidad:	
Nombre:	
Descripción:	

Nombre: CE_Baja_Temporal	
Tipo de clase: Entidad	
Atributo:	Tipo:
Idespecialidad_movimiento	INTEGER
tiempo_autorizado	DATETIME
fecha_debe_reincorporarse	DATETIME
Para cada responsabilidad:	
Nombre:	
Descripción:	

Nombre: CE_Profesionales	
Tipo de clase: Entidad	
Atributo:	Tipo:
idProfesional	INTEGER
noCarnet	CHAR
nombre	STRING
primerApellido	STRING
segundoApellido	STRING
provincia	STRING
municipio	STRING
telefono	INTEGER
idmilitancia	INTEGER
sexo	BOOLEAN
Idestado_civil	INTEGER
Id_gradoCientifica	STRING
edad	INTEGER
Id_categoriaDocente	STRING
Id_categoriaInvestigativa	STRING
idProvincia	INTEGER
idMunicipio	INTEGER
fechaNacimiento	DATETIME
unidad_salud	STRING
idunidad	INTEGER
nombre_especialidad_seg	STRING
Para cada responsabilidad:	
Nombre:	

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

Descripción:	
Nombre: CE_Profesional_Especialidad	
Tipo de clase: Entidad	
Atributo:	Tipo:
idProfesional	INTEGER
idespecialidad	INTEGER
fecha_otorgamiento	DATETIME
fecha_inicio	DATETIME
fecha_probable_terminacion	DATETIME
unidad_salud	VARCHAR
provincia	STRING
municipio	STRING
version_plan	INTEGER
duracion_plan	INTEGER
ldvia_ingreso	INTEGER
Para cada responsabilidad:	
Nombre:	
Descripción:	

Nombre: CE_Curso	
Tipo de clase: Entidad.	
Atributo:	Tipo:
idcurso	INTEGER
nombre	STRING
idespecialidad	INTEGER
Para cada responsabilidad:	
Nombre:	
Descripción:	

Nombre: CE_Datos_Carrera.	
Tipo de clase: Entidad.	
Atributo:	Tipo:
idprofesional	INTEGER
unidad_salud_graduado	STRING
anno_graduacion	INTEGER
carrera	STRING
indice_academico	FLOAT
alumno_ayudante	BOOLEAN
instructor_no_graduado	BOOLEAN
vanguardia_mario_munoz	BOOLEAN
Para cada responsabilidad:	
Nombre:	
Descripción:	

Nombre: CE_Especialidad.	
Tipo de clase: Entidad.	
Atributo:	Tipo:
idespecialidad	INTEGER
idtipo_especialidad	INTEGER
nombre_especialidad	STRING
Para cada responsabilidad:	
Nombre:	
Descripción:	

Nombre: CE_Estado_Civil.	
Tipo de clase: Entidad.	
Atributo:	Tipo:
Idestado_civil	INTEGER
nombre_estado_civil	STRING
Para cada responsabilidad:	
Nombre:	
Descripción:	

Nombre: CE_Estancia.	
Tipo de clase: Entidad.	
Atributo:	Tipo:
Idestancia	INTEGER
nombre	STRING
idespecialidad	INTEGER
Para cada responsabilidad:	
Nombre:	
Descripción:	

Nombre: CE_Militancia.	
Tipo de clase: Entidad.	
Atributo:	Tipo:
idmilitancia	INTEGER
nombre_militancia	STRING
sigla	STRING
Para cada responsabilidad:	
Nombre:	
Descripción:	

Nombre: CE_Rotacion.	
Tipo de clase: Entidad.	
Atributo:	Tipo:
idrotacion	INTEGER
nombre	STRING
idespecialidad	INTEGER
Para cada responsabilidad:	
Nombre:	
Descripción:	

Nombre: CE_Traslados.	
Tipo de clase: Entidad.	
Atributo:	Tipo:
idespecialidad_movimiento	INTEGER
provincia	STRING
Para cada responsabilidad:	
Nombre:	
Descripción:	

3.4 Diagramas de Interacción

Los diagramas de interacción son diagramas dinámicos que muestran la colaboración de los objetos que intervienen en la realización de los casos de uso. Existen dos tipos de estos diagramas: diagramas de secuencias y de colaboración. En el trabajo se desarrollaron los diagramas de secuencia que muestran la secuencia ordenada de las operaciones entre objetos.

Ver [ANEXO 3](#)

3.5.1 Descripción de las tablas

Nombre: tb_profesional.		
Descripción: Esta tabla contiene la información que se recoge de los profesionales.		
Atributo	Tipo	Descripción
idprofesional	TINYINT	Representa el identificador del profesional.
noCarnet	CHAR	Representa el número del carné de identidad del profesional.
nombre	VARCHAR	Representa el nombre del profesional.
primerApellido	VARCHAR	Representa el primer apellido del profesional.
segundoApellido	VARCHAR	Representa el centro de trabajo del profesional
edad	TINYINT	Representa la edad del profesional.
idmunicipio	TINYINT	Representa el identificador del municipio al que pertenece del profesional.
idprovincia	TINYINT	Representa el identificador de la provincia a la que pertenece del profesional.
telefono	INT	Representa el teléfono privado del profesional
provincia	VARCHAR	Representa la descripción de la provincia
sexo	BIT	Representa el sexo del profesional.
idestado_civil	INT	Representa el estado civil del profesional.
municipio	VARCHAR	Representa el municipio al que pertenece del profesional.
idmilitancia	INT	Representa el identificador de la militancia del profesional
id_categoriaCientifica	TINYINT	Representa la categoría científica que tiene el profesional
id_categoriaDocente	TINYINT	Representa la categoría docente que tiene el profesional
id_categoriaInvestigativa	TINYINT	Representa la categoría investigativa que tiene el profesional
nombre_especialidad_seg	VARCHAR	Representa el nombre de la segunda especialidad que tiene el profesional
fechaNacimiento	DATETIME	Representa la fecha nacimiento del profesional
idunidad	TINYINT	Representa el identificador de la unidad de salud a la que pertenece del profesional.
unidad_salud	VARCHAR	Representa la unidad de salud a la que pertenece del profesional.

Tabla 3.1 tb_profesional

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

Nombre: tb_militancia.		
Descripción: Esta tabla contiene las diferentes militancias que existen.		
Atributo	Tipo	Descripción
idmilitancia	INT	Representa el identificador de la militancia.
nombre_militancia	VARCHAR	Representa el nombre de la militancia.
sigla	VARCHAR	Representa la sigla de la militancia.

Tabla 3.2 tb_militancia

Nombre: tb_militancia.		
Descripción: Esta tabla es la relación que existe entre las tablas profesional y militancia.		
Atributo	Tipo	Descripción
idprofesional	TINYINT	Representa el identificador del profesional.
idmilitancia	TINYINT	Representa el identificador de la militancia.

Tabla 3.3 tr_profesional_militancia

Nombre: tb_estado_civil.		
Descripción: Esta tabla contiene el estado civil de los profesionales.		
Atributo	Tipo	Descripción
idestado_civil	INT	Representa el identificador del estado civil.
nombre_estado_civil	VARCHAR	Representa el nombre del estado civil.

Tabla 3.4 tb_estado_civil

Nombre: tb_datos_carrera.		
Descripción: Esta tabla contiene la información de los datos de la carrera.		
Atributo	Tipo	Descripción
idprofesional	TINYINT	Representa el identificador del profesional para recoger determinados datos de su carrera.
unidad_salud_graduado	VARCHAR	Representa la unidad de salud donde estudió la carrera.
anno_graduado	INT	Representa el año de graduación del profesional.
carrera	VARCHAR	Representa la carrera que estudió el profesional.
indice_academico	FLOAT	Representa el índice académico del profesional.
alumno_ayudante	BIT	Representa la condición de alumno ayudante en una especialidad del profesional.
instructor_no_graduado	BIT	Representa la condición de instructor no graduado en una especialidad del profesional
vanguardia_mario_munoz	BIT	Representa la condición de vanguardia de Mario Muños Monroy del profesional.

Tabla 3.5 tb_datos_carrera

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

Nombre: tr_profesional_especialidad.		
Descripción: Esta tabla contiene la información de las especialidades para un profesional determinado.		
Atributo	Tipo	Descripción
idprofesional	TINYINT	Representa el identificador del profesional.
idespecialidad	INT	Representa el identificador de la especialidad.
fecha_otorgamiento	DATETIME	Representa la fecha que se le otorgó la especialidad a un profesional.
fecha_inicio	DATETIME	Representa la fecha de inicio a la especialidad de un profesional.
fecha_probable_terminacion	DATETIME	Representa la fecha probable de terminación de la especialidad de un profesional.
unidad_salud	VARCHAR	Representa el nombre de la unidad donde realiza la residencia un profesional.
municipio	VARCHAR	Representa el municipio donde realiza la residencia el profesional.
provincia	VARCHAR	Representa la provincia donde realiza la residencia el profesional.
version_plan	INT	Representa la versión del plan de estudio por el cual se rige un profesional.
duracion_plan	INT	Representa la duración del plan de estudio de un profesional.
idvia_ingreso	INT	Representa el identificador de la vía de ingreso determinada para una especialidad de un profesional.

Tabla 3.6 tr_profesional_especialidad

Nombre: tn_especialidad.		
Descripción: Esta tabla contiene la información de las especialidades.		
Atributo	Tipo	Descripción
idespecialidad	INT	Representa el identificador de la especialidad.
idtipo_especialidad	INT	Representa el identificador de un tipo determinado de especialidad.
nombre_especialidad	VARCHAR	Representa el nombre de las especialidades.

Tabla 3.7 tn_especialidad

Nombre: tb_tipo_especialidad.		
Descripción: Esta tabla contiene la información de especialidades determinadas.		
Atributo	Tipo	Descripción
idtipo_especialidad	INT	Representa el identificador de un tipo determinado de especialidad.
nombre_tipo_especialidad	VARCHAR	Representa el nombre de un tipo determinado de especialidad.

Tabla 3.8 tb_tipo_especialidad

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

Nombre: tr_curso.		
Descripción: Esta tabla contiene la información de los cursos asociados a la especialidad del profesional.		
Atributo	Tipo	Descripción
idcurso	INT	Representa el identificador del curso.
curso	VARCHAR	Representa el nombre del curso.
idespecialidad	INT	Representa el identificador de la especialidad que tiene asociado determinados curso.

Tabla 3.9 tr_curso

Nombre: tr_profesional_curso.		
Descripción: Esta tabla contiene la información de los cursos asociados a un profesional.		
Atributo	Tipo	Descripción
idprofesional	TINYINT	Representa el identificador del profesional que tiene asociado una serie de cursos.
idcurso	INT	Representa el identificador del curso por profesional.
fecha	DATETIME	Representa la fecha de asignación del curso a un profesional.

Tabla 3.10 tr_profesional_curso

Nombre: tn_estancia.		
Descripción: Esta tabla contiene la información de las estancias asociadas a la especialidad del profesional.		
Atributo	Tipo	Descripción
idestancia	INT	Representa el identificador de la estancia.
nombre	VARCHAR	Representa el nombre de la estancia.
idespecialidad	INT	Representa el identificador de la especialidad que tiene asociada determinadas estancias.

Tabla 3.11 tn_estancia

Nombre: tr_profesional_estancia.		
Descripción: Esta tabla contiene la información de las estancias asociadas a un profesional.		
Atributo	Tipo	Descripción
idprofesional	TINYINT	Representa el identificador del profesional que tiene asociado una serie de estancias.
idestancia	INT	Representa el identificador de la estancia por profesional.
fecha	DATETIME	Representa la fecha de asignación de la estancia a un profesional.

Tabla 3.12 tr_profesional_estancia

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

Nombre: tn_rotacion.		
Descripción: Esta tabla contiene la información de las rotaciones asociadas a un profesional.		
Atributo	Tipo	Descripción
nombre	VARCHAR	Representa el nombre de la rotación.
idrotacion	INT	Representa el identificador de la rotación por profesional.
idespecialidad	INT	Representa el identificador de la especialidad que tiene asociada determinadas rotaciones.

Tabla 3.13 tn_rotacion

Nombre: tr_profesional_rotacion.		
Descripción: Esta tabla contiene la información de las rotaciones asociadas a la especialidad del profesional.		
Atributo	Tipo	Descripción
idrotacion	INT	Representa el identificador de la rotación.
nombre	VARCHAR	Representa el nombre de la rotación.
fecha_inicio	DATETIME	Representa la fecha de inicio de la rotación de un profesional.
lugar	VARCHAR	Representa el lugar donde realiza la rotación un profesional.
tiempo	DATETIME	Representa el tiempo en años de la rotación de un profesional.
idprofesional	TINYINT	Representa el identificador del profesional que tiene asociado una serie de estancias.

Tabla 3.14 tr_profesional_rotacion

Nombre: tb_via_ingreso.		
Descripción: Esta tabla contiene la información de las vías de ingreso a las especialidades.		
Atributo	Tipo	Descripción
idvia_ingreso	INT	Representa el identificador de las vías de ingreso a la especialidad.
nombre_via_ingreso	INT	Representa el nombre de las vías de ingreso.
idtipo_ingreso	INT	Representa el identificador de un tipo determinado de vía de ingreso a la especialidad.

Tabla 3.15 tb_via_ingreso

Nombre: tb_tipo_ingreso.		
Descripción: Esta tabla contiene la información de tipos determinados de vías de ingreso asociados a especialidades.		

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

Atributo	Tipo	Descripción
idtipo_ingreso	INT	Representa el identificador de un tipo determinado de vía de ingresos a la especialidad.
nombre_tipo_ingreso	VARCHAR	Representa el nombre del tipo de ingreso determinado asociado a una especialidad.

Tabla 3.16 tb_tipo_ingreso

Nombre: tb_examen.		
Descripción: Esta tabla contiene la información de los exámenes.		
Atributo	Tipo	Descripción
idexamen	INT	Representa el identificador de los exámenes.
idprofesional	TINYINT	Representa el identificador del profesional que tiene asociado una serie de estancias.
idespecialidad	INT	Representa el identificador de la especialidad que tiene asociado una serie de exámenes.

Tabla 3.17 tb_examen

Nombre: tb_examen_estatal.		
Descripción: Esta tabla contiene la información de los exámenes estatales.		
Atributo	Tipo	Descripción
idexamen	INT	Representa el identificador del examen.
tributa_grado	INT	Representa la condición de grado científico una buena defensa del examen estatal un profesional determinado.
fecha	DATETIME	Representa la fecha de realización del examen estatal.
calificacion	INT	Representa la nota del examen realizado.
idtipo_examen	INT	Representa el identificador del tipo de examen.

Tabla 3.18 tb_examen_estatal

Nombre: tb_examen_tte.		
Descripción: Esta tabla contiene la información de los TTE (Trabajo de Terminación de la Especialidad).		
Atributo	Tipo	Descripción
idexamen	INT	Representa el identificador del examen.
fecha_entrega	DATETIME	Representa la fecha de entrega del protocolo al tribunal.
fecha_aprobación	DATETIME	Representa la fecha de aprobación por el Consejo Científico para la defensa del trabajo.

Tabla 3.19 tb_examen_tte

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

Nombre: tb_examen_promocion.		
Descripción: Esta tabla contiene la información de los exámenes de promoción.		
Atributo	Tipo	Descripción
idexamen	INT	Representa el identificador del examen.
idconvocatoria	INT	Representa el identificador de la convocatoria a la que se presenta el profesional.
convocatoria	BIT	Representa convocatoria en que realizó el examen.
fecha	DATETIME	Representa la fecha de realización del examen.
calificacion	INT	Representa la nota del examen realizado.
anno	INT	Representa el año en que se encuentra el profesional.
idtipo_examen	INT	Representa el identificador del tipo de examen.

Tabla 3.20 tb_examen_promocion

Nombre: tb_tipo_examen.		
Descripción: Esta tabla contiene la información de las convocatorias de los exámenes.		
Atributo	Tipo	Descripción
idtipo_examen	INT	Representa el identificador del tipo de examen determinado.
nombre_examen	VARCHAR	Representa el nombre del examen.

Tabla 3.21 tb_tipo_examen

Nombre: tr_especialidad_movimiento.		
Descripción: Esta tabla contiene la información de los movimientos por especialidades.		
Atributo	Tipo	Descripción
idespecialidad_movimiento	INT	Representa el identificador del movimiento por especialidades.
causa	TEXT	Representa las causas que dan origen a un movimiento de la especialidad.
fecha	DATETIME	Representa la fecha de realización del movimiento.
idprofesional	TINYINT	Representa el identificador del profesional que tiene asociado una serie de estancias.
idespecialidad	INT	Representa el identificador de la especialidad que tiene asociado movimientos.
tipo_baja	TINYINT	Representa el identificador de la baja que se va a realizar.

Tabla 3.22 tr_especialidad_movimiento

Nombre: tb_baja_temporal.		
Descripción: Esta tabla contiene la información de las bajas temporales.		
Atributo	Tipo	Descripción
idespecialidad_movimiento	INT	Representa el identificador para las bajas temporales.
tiempo_autorizado	VARCHAR	Representa el tiempo autorizado de la baja temporal.
fecha_debe_reincorporarse	DATETIME	Representa la fecha en la cual debe reincorporarse el profesional de la baja temporal.

Tabla 3.23 tb_baja_temporal

Nombre: tb_traslado.		
Descripción: Esta tabla contiene la información de los traslados.		
Atributo	Tipo	Descripción
idespecialidad_movimiento	INT	Representa el identificador para los traslados.
Provincial	VARCHAR	Representa la provincia a la que se traslada el profesional.

Tabla 3.24 tb_traslado

3.6 Definiciones del diseño

La interfaz es la apariencia del sistema y está definida por el diseño visual que se realice. Considerando que es la capa de presentación al usuario, se hace necesario mostrar una aplicación con interfaz clara, amigable y fácil de usar. Esto propicia comodidad al usuario al intercambiar con el sistema. Esta aplicación va dirigida al personal del MINSAP. En ocasiones no todos tienen la capacitación necesaria para trabajar con la misma. Por esta razón es necesario que sea lo más flexible posible para garantizar claridad y fácil manejo del flujo de datos.

3.6.1 Diseño de página principal

La página principal del sistema debe llevar: una cabecera que lo identificará, una imagen no muy grande representativa del sitio, barra de menú con sus opciones, área de trabajo, área de pie o barra de estado. Además debe ofrecer información sobre el usuario que está trabajando en la aplicación, tal como: nombre de usuario y tipo de acceso.

A continuación se detallan algunos elementos a tener en cuenta:

- ✓ Uniformidad, se requiere que todos los patrones de diseño adopten las reglas de estandarización. Por ejemplo, que la fuente y campos sean del mismo color y tamaño, utilizando para ello hojas de estilos.
- ✓ Para gestionar el cúmulo de datos se necesita de un área de trabajo, y en ocasiones requiere de la optimización de los elementos. Lo que contribuye a que las páginas no se encuentren tan cargadas y puedan mostrar cierta claridad en los datos.
- ✓ Mostrar un orden de paginado, posibilita que la información quede estructurada para todas las pantallas de la misma forma.

3.7 Tratamiento de errores

En el sistema propuesto se evitan, minimizan y tratan los posibles errores, con el fin de garantizar la integridad y confiabilidad de la información que en este se registra y muestra. Los errores se tratan en una página especial que incluye el fichero de configuración general, y está preparada para recoger el número del error y presentar la pantalla con el error que le corresponde a ese código.

Los mensajes de error que emite el sistema se muestran en un lenguaje de fácil comprensión para los usuarios. Cuando se introduce información en un formulario y faltan datos, sale un cuadro de alerta indicando el campo o dato que falta. Similar ocurre cuando se introduce información errónea en un campo numérico que lleva una cantidad exacta.

3.8 Seguridad

La seguridad del sistema es un parámetro esencial a tener en cuenta, pues la información que se digitalizará es clasificada para el Sistema Nacional de Salud y requiere de protección. El sistema desarrollado cuenta con un módulo de administración. Brinda un estado seguro para cada usuario según el rol que desempeña y nivel de acceso. Logra un alto nivel de confiabilidad, integridad y disponibilidad de información.

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

Gracias a la seguridad puesta en práctica, el sistema tiene la capacidad de prevenir la entrada de código malicioso, pues no permite a usuarios indebidos, ver información de alta confidencialidad. Este tipo de ataque se ha evitado utilizando funciones propias de PHP.

En el presente capítulo se desarrollaron los diagramas de clases de la aplicación y el diseño de la base de datos del sistema. Se describieron además, los principios de diseño, específicamente, los temas de estándares de la interfaz y la concepción del tratamiento de errores.

CAPÍTULO 4: IMPLEMENTACIÓN

En este capítulo se documenta la importancia de utilizar otros sistemas para un buen funcionamiento. Se profundiza en la realización de la integración, por las necesidades planteadas por los clientes. Se describe el flujo de trabajo de implementación y se representa la organización del sistema de acuerdo a los nodos específicos en el modelo de despliegue.

4.1 Dependencias y Relaciones con otros Sistemas

El sistema se relaciona con otros sistemas como: RPS (Registro de Personal de la Salud), el RUS (Registro de Unidades de Salud), el RU (Registro de Ubicación), el RC (Registro de Ciudadano) y el SAAA (Single Authentication, Authorization and Accounting) para controlar el acceso desde los diferentes niveles del Sistema Nacional de Salud.

Con la utilización de estos servicios, se hace posible el funcionamiento del Módulo de Posgrado para el MINSAP en sus diferentes áreas, ya sea de Capacitación, Superación, Maestrías, Doctorados y Especialidades, y hace posible que estas se interrelacionen. La posibilidad de utilizar los Servicios Web brindados por otros componentes, garantiza la reutilización de los datos y evita la duplicidad de la información.

4.4.1 Componente SAAA (Single Authentication, Authorization and Accounting)

SAAA (Single Authentication, Authorization and Accounting). Este componente tiene implementado un mecanismo de seguridad basado en el modelo de Autenticación, Autorización y Auditoría. La autenticación es la primera acción del usuario en el sistema. Consiste en suministrar un nombre de usuario único y una contraseña. Si el usuario no se encuentra registrado se reporta un error de acceso. Si el usuario autenticado se encuentra registrado se autoriza su acceso y se crea un certificado digital que contiene un identificador único (token) de 32 caracteres, que tiene como información: el identificador del usuario, el nivel de acceso (Nacional, Provincial, Municipal, Unidad de Salud), el identificador de nivel de acceso, un listado de los módulos a los que el usuario tiene acceso y el tipo de acceso en cada uno de ellos (Editor o Visualizador).

La autenticación, autorización y seguridad en el proyecto Sistema de Gestión de Información para la Formación Posgraduada en Salud, se realiza a través del SAAA. Esto garantiza el acceso de los diferentes usuarios que se identifican con la aplicación una vez en funcionamiento.

4.1.2 RUS (Registro de Unidades de Salud)

La comunicación con este servicio se establece para conocer la unidad de salud a la cual pertenecen los usuarios de este nivel de dirección que se autentican en el sistema. De él se obtienen datos adicionales de las unidades de salud como son: el tipo de unidad y la subordinación.

Este registro tiene almacenada la información correspondiente a las unidades de salud de todo el país. Brinda un conjunto de funcionalidades que se utilizan en el Sistema de Gestión de Información para la Formación Posgraduada en Salud. Facilita la búsqueda interna para saber desde qué unidad de salud se está llevando a cabo una actividad de posgrado.

4.1.3 RPS (Registro de Personal de Salud)

Este componente tiene la funcionalidad de gestionar la información de todo el personal de salud del país, ya sea profesional o no. El Sistema de Gestión de Información para la Formación Posgraduada en Salud utiliza el RPS para obtener información de las personas que forman parte de la matrícula del programa de las Especialidades.

4.1.4 RC (Registro de Ciudadano).

Este es un componente que contiene la información referente a cada ciudadano del país, guarda sus datos personales como: nombre, apellidos, carné de identidad, edad, dirección particular y datos de nacimiento. Se realiza una integración con este registro para homologar los datos que maneja el sistema con los reales, que son los que contiene el Registro de Ciudadano, responsable absoluto de garantizar la integridad de esta información.

4.2 Integración del módulo Especialidades al Sistema

La concepción de la integración para el sistema de la docencia de posgrado esta concebida por 3 módulos. Estos comprenden: Especialidades, Maestría y Doctorado, Capacitación y Superación Profesional dentro del Departamento de Posgrado del MINSAP, organismo rector de esta actividad conjuntamente con el MES. El sistema dispone de un grupo de funcionalidades propias de cada módulo que se implementó de acuerdo a los requerimientos plateados por el cliente.

El sistema que se confeccionó, propone un trabajo íntegro entre los 3 módulos. La labor de algunos depende de otros, por eso la necesidad de integrar la información. Se dispone además de una base de datos única que contiene la información concentrada para la utilización y obtención de información. No se necesita de algún sistema aislado que intervenga. De ahí que el nivel de respuesta sea mucho mayor. Así mismo toda la información se replica a una misma base de datos, lo que evita la redundancia de valores de los profesionales, los cursos y actividades que se complementen.

El sistema para la docencia de posgrado posibilita un mejor funcionamiento y automatización de los procesos de registro y actualización de las actividades de los profesionales. Como resultado se muestra, en todo momento, el quehacer de cada uno de ellos y por ende un resultado significativo para el país.

4.3 Modelo de Implementación

El modelo de implementación describe cómo los elementos del modelo de diseño y las clases se implementan en términos de componentes, ficheros de código fuente, ejecutables, entre otros. Los diagramas de despliegue y componentes conforman el modelo de implementación al describir los componentes y construir su organización y dependencia entre los nodos físicos en que funciona la aplicación. Aquí se muestra como quedó la lógica de la implementación para el módulo de las Especialidades.

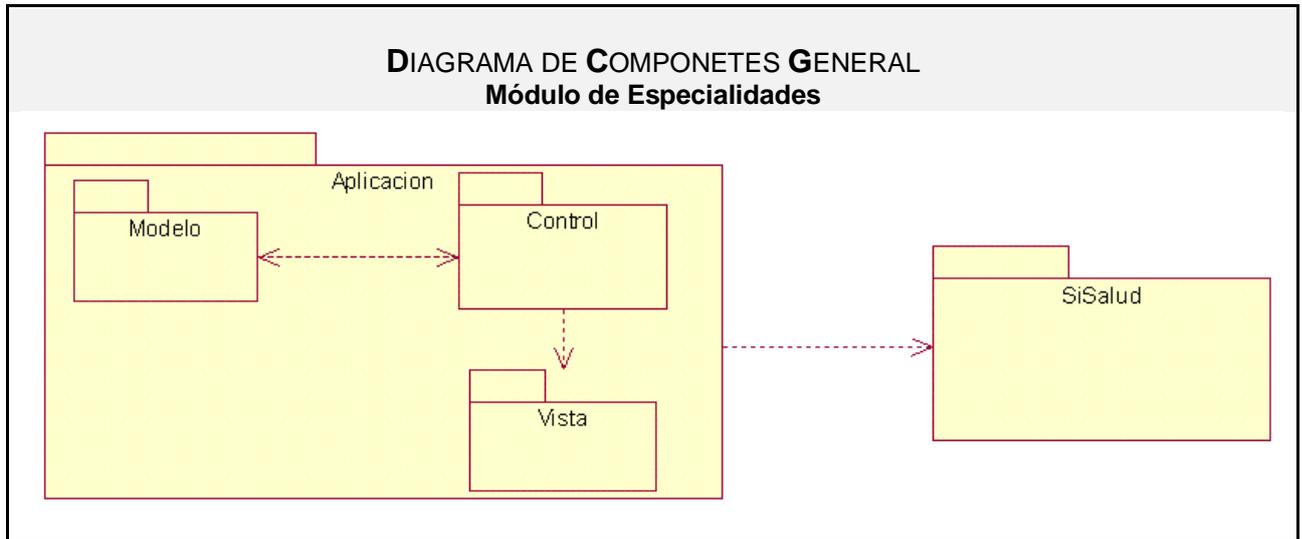


Fig.4.1 Diagrama de Componentes General

4.3.1 Diagrama de componentes

Los diagramas de componentes describen los elementos físicos del sistema y sus relaciones. Muestran las opciones de realización incluyendo código fuente, binario y ejecutable. Los componentes, como su nombre lo indica, representan todos los tipos de elementos software que entran en la fabricación de aplicaciones informáticas. Por esta razón el diagrama de componentes muestra las organizaciones y dependencias lógicas entre componentes software.

Un diagrama de componente es un segmento de reutilización, que cuenta con interfaces y sus relaciones. Se puede tomar como un diagrama de clases de alto nivel. A continuación se muestra la estructura del modelo de implementación para el módulo de Especialidades, respecto al diseño realizado anteriormente.

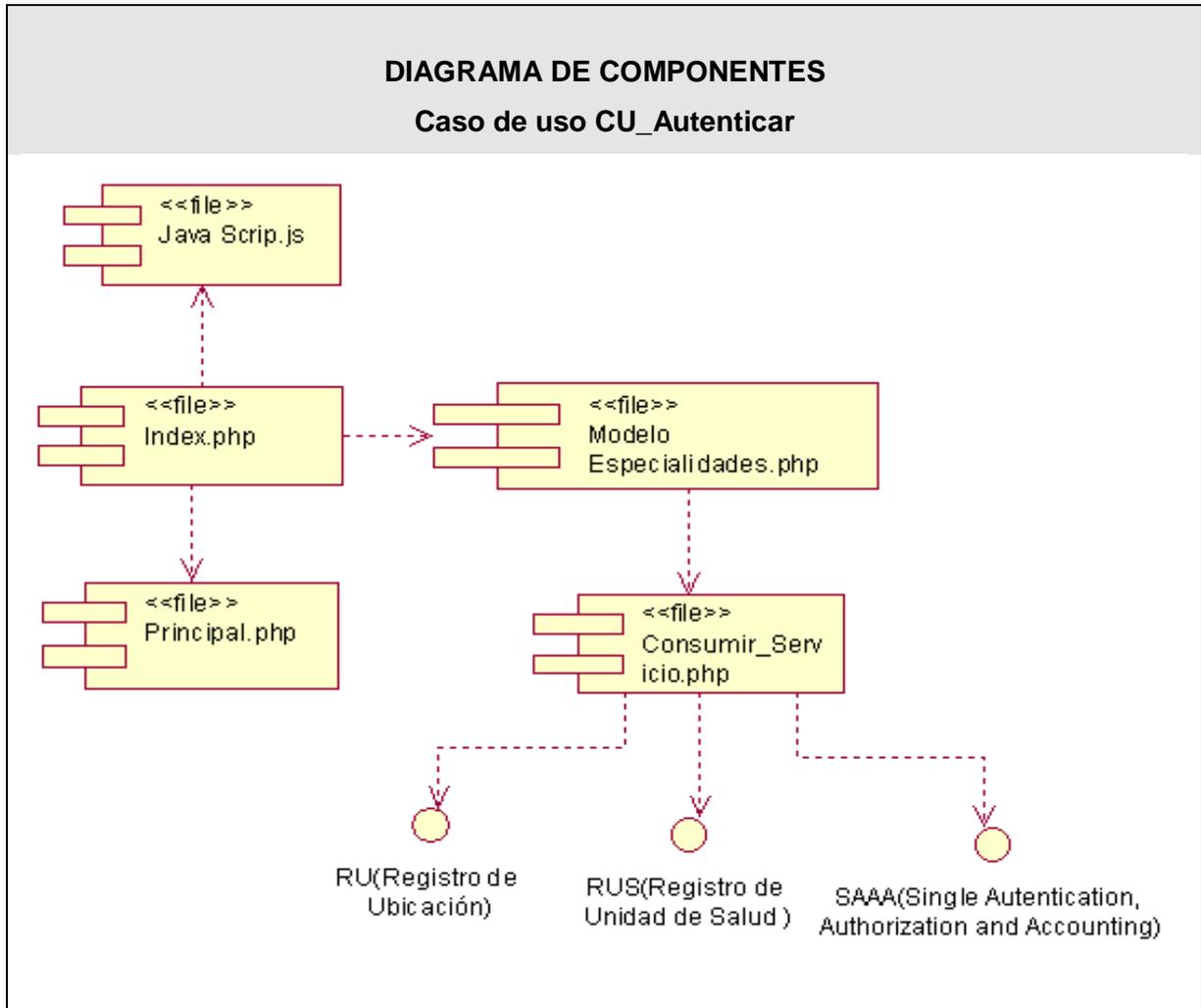


Fig. 4.1 Diagrama de componente del Caso de uso CU_Autenticar

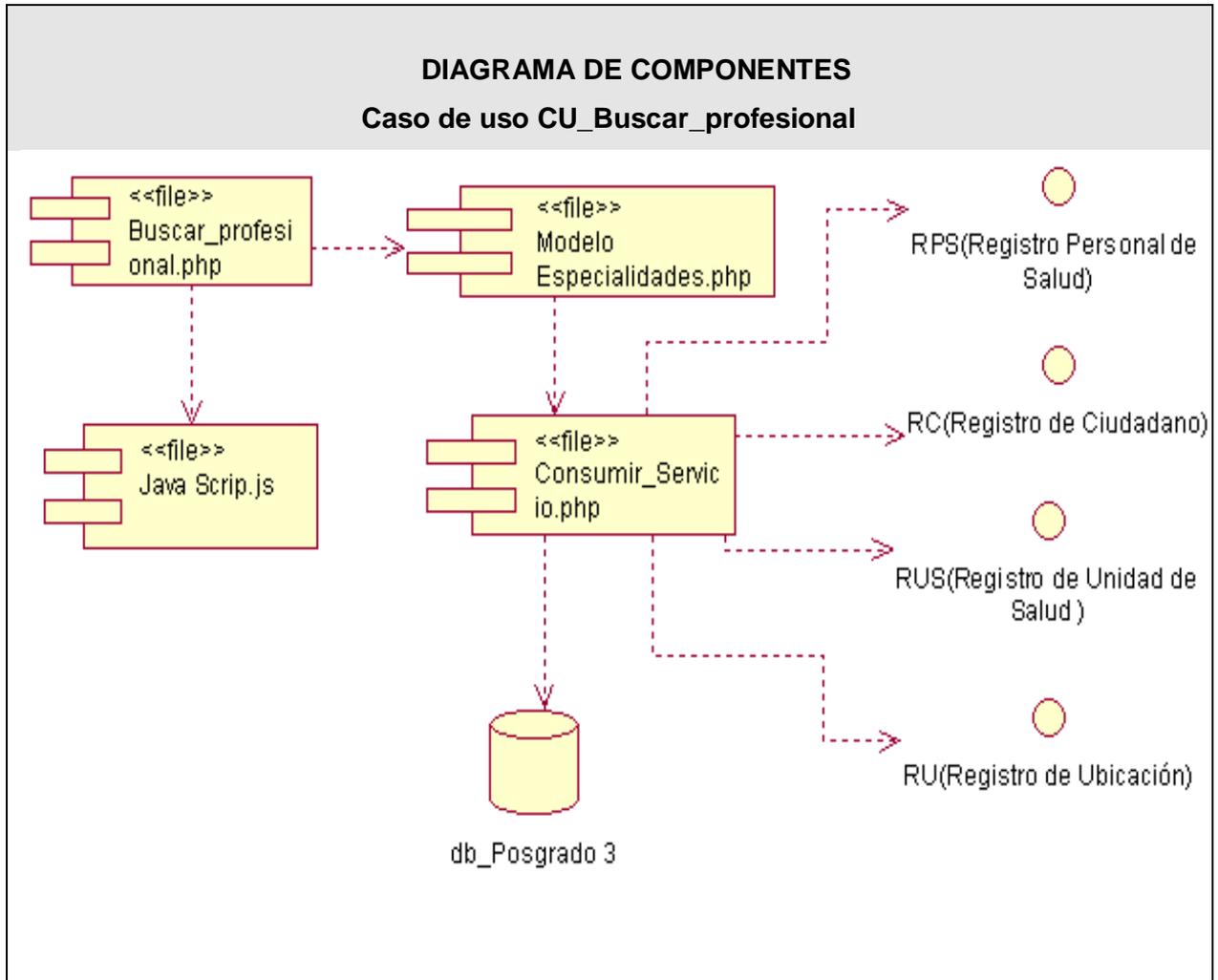


Fig. 4.2 Diagrama de componente del Caso de uso CU_Buscar_profesional

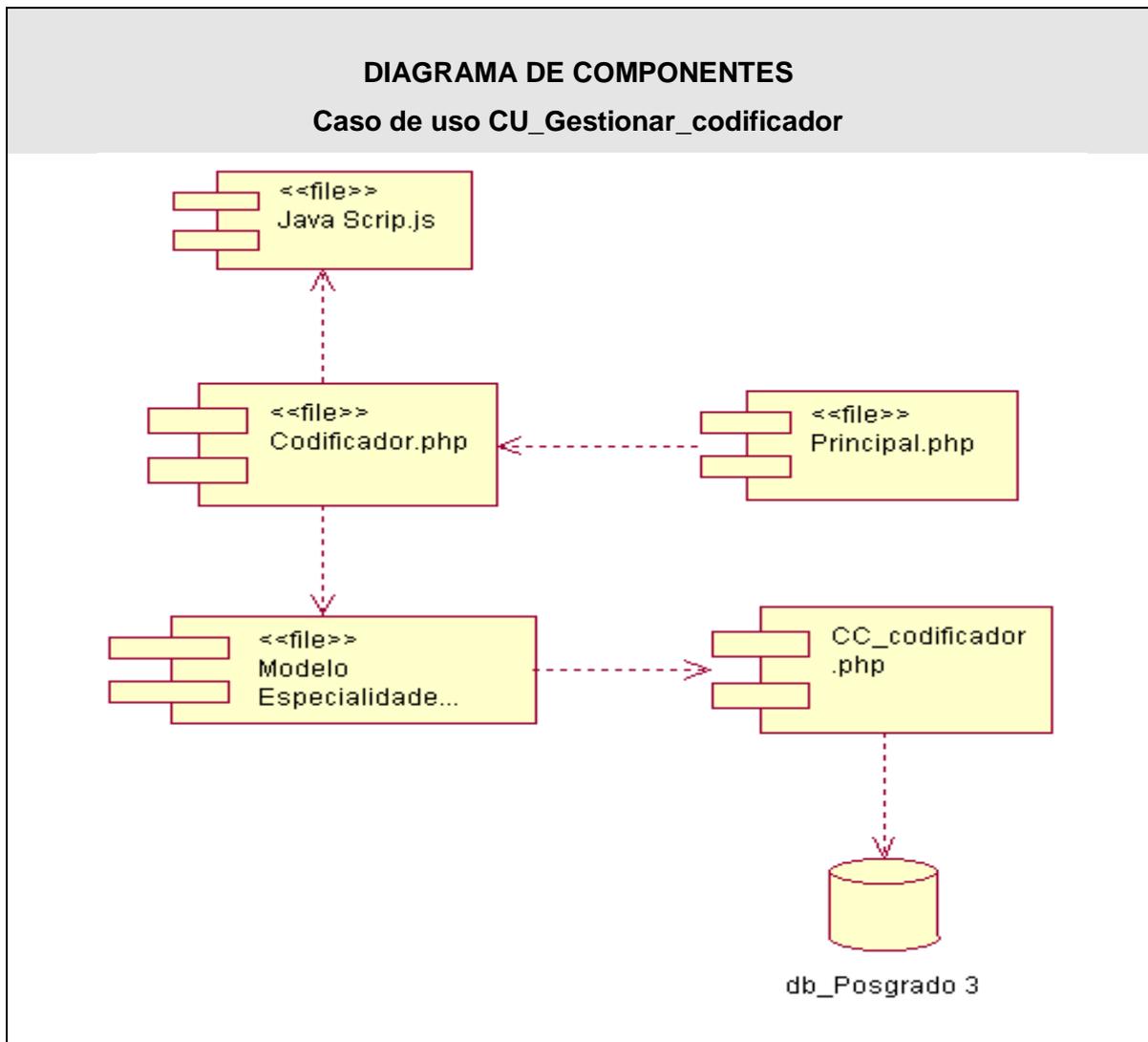


Fig. 4.3 Diagrama de componente del Caso de uso CU_Gestionar_Codificador

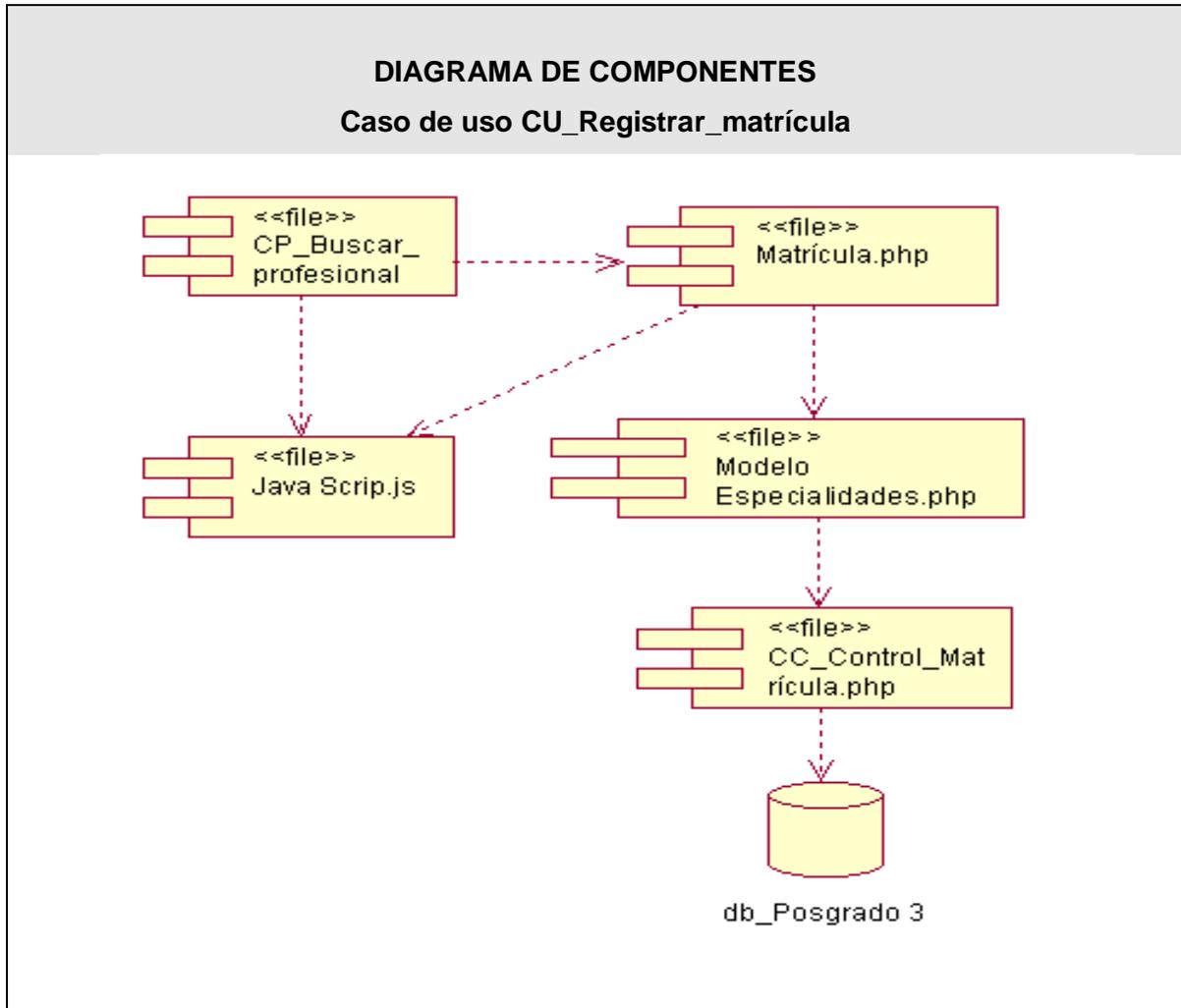


Fig. 4.3 Diagrama de componente del Caso de uso CU_Registrar_matrícula

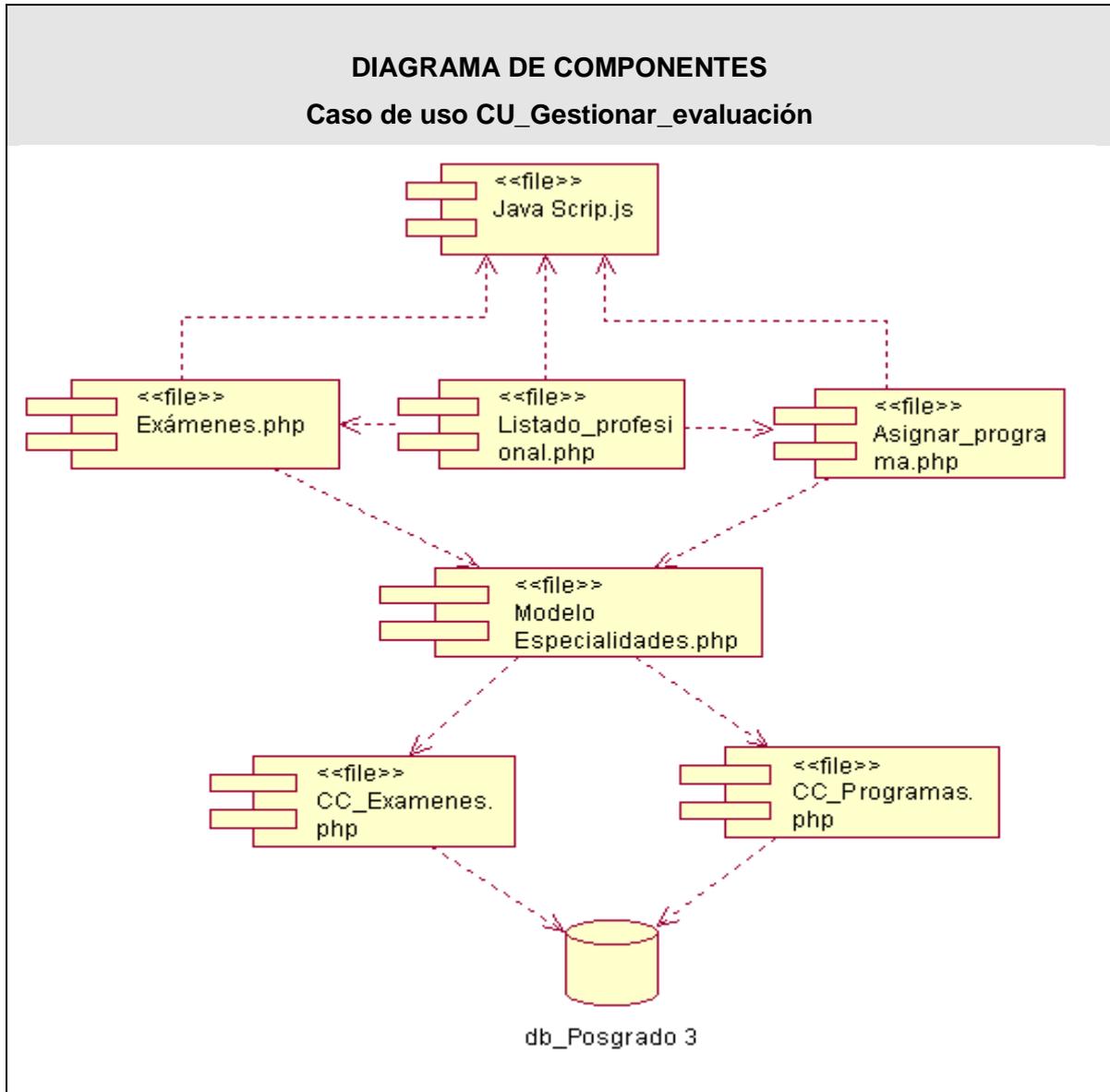


Fig. 4.4 Diagrama de componente del Caso de uso CU_Gestionar_evaluación

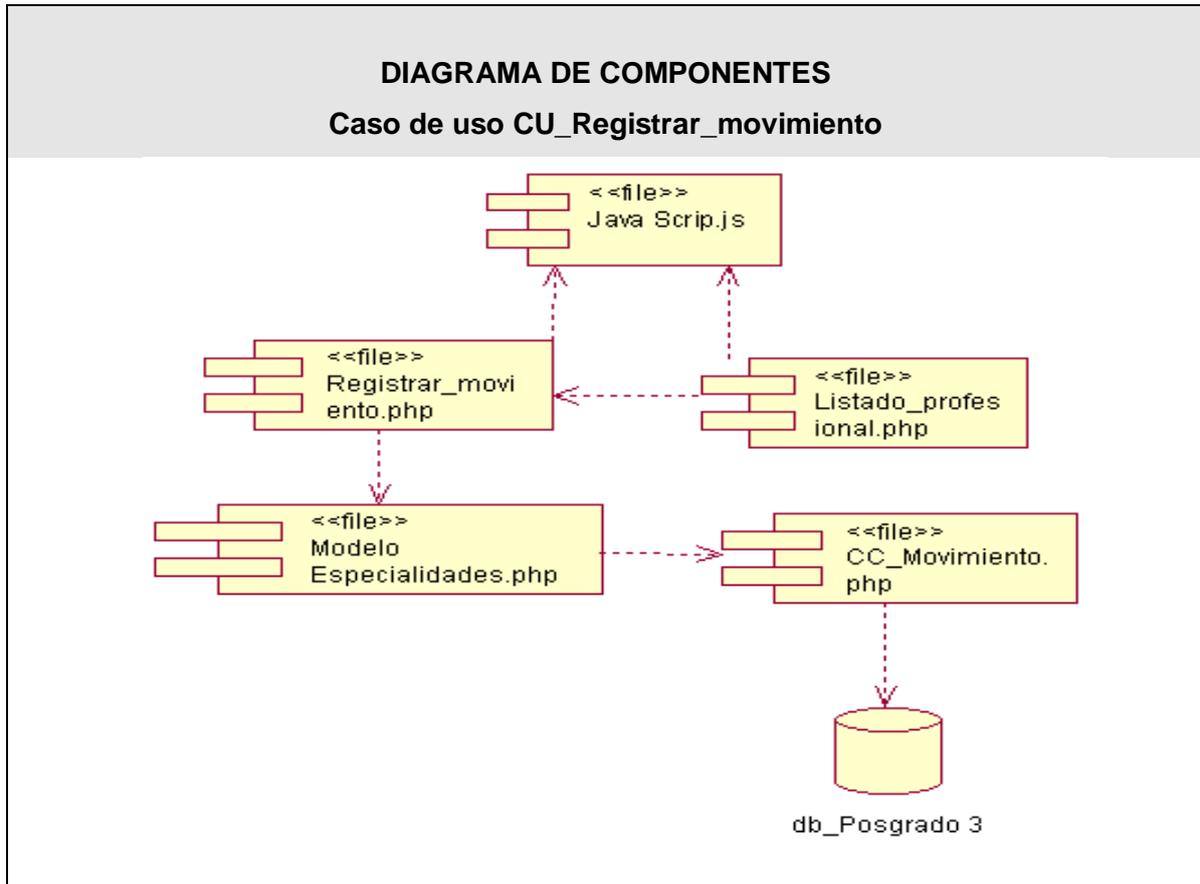


Fig. 4.5 Diagrama de componente del Caso de uso CU_Registrar_movimiento

4.3.2 Diagrama de Despliegue

Los diagramas de despliegue muestran la configuración en funcionamiento del sistema, incluyendo su hardware y su software. Estos permiten describir la plataforma sobre la que se ejecuta el sistema al representar la topología de procesadores y dispositivos, lo que facilita la definición y manejo de la frontera entre el hardware y el software.

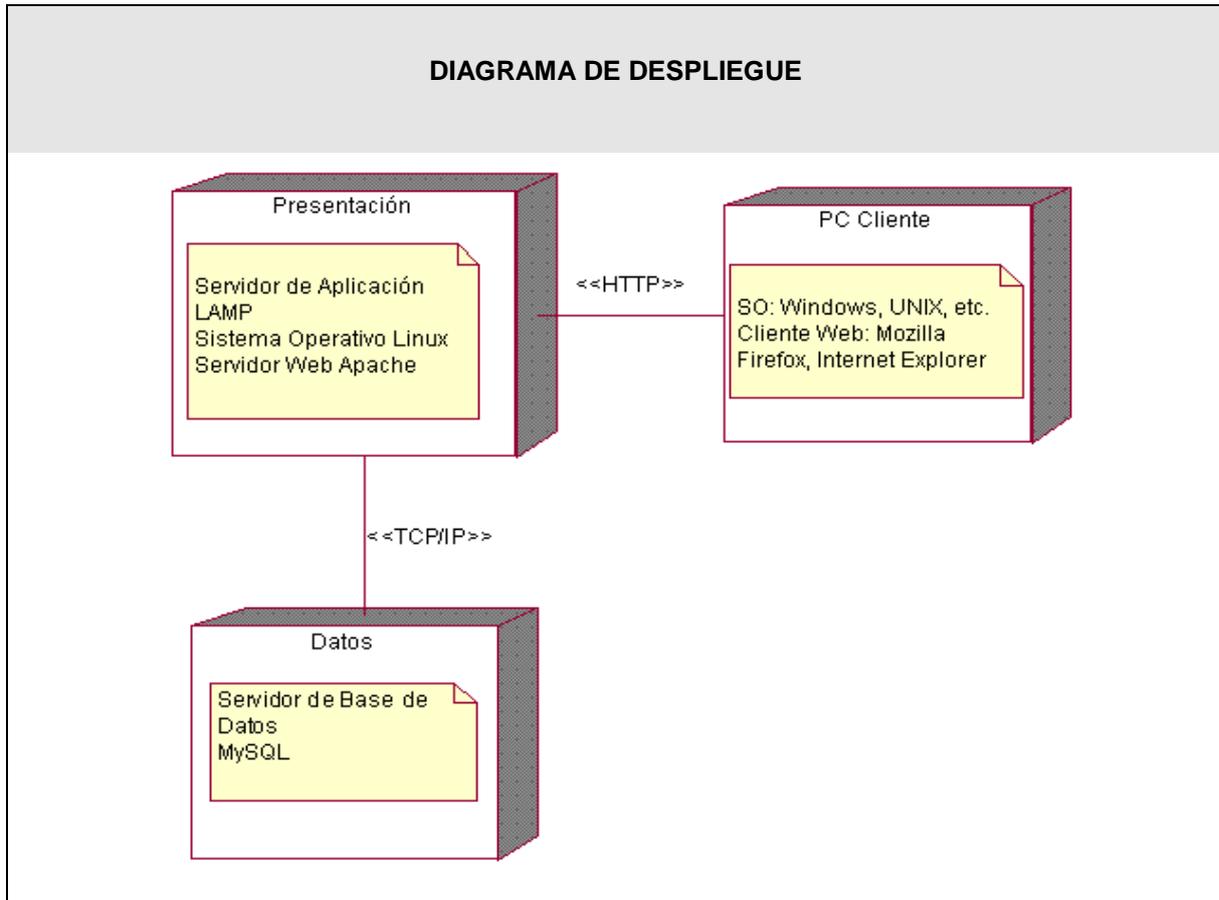


Fig. 4.7 Diagrama de Despliegue

Se realizó una investigación de la utilización de servicios externos. Se explicaron las causas de la integración de los programas del Departamento de Posgrado. Se realizaron los diagramas de implementación correspondientes al presente capítulo, importantes para las fases de despliegue y puesta en práctica de la aplicación.

CONCLUSIONES

Tras realizar la investigación para el tratamiento del Módulo de Especialidades del Sistema de Gestión de Información en el Proceso de Formación Posgraduada en Salud, se puede decir que su desarrollo estuvo dirigido hacia las tareas de la investigación planteadas, llegando así a las siguientes conclusiones:

- ✓ Se analizaron los aspectos conceptuales relacionados con la formación de posgrado en el MINSAP y específicamente con el programa de especialidades, para determinar la importancia de la informatización de esta área.
- ✓ La aplicación de las tecnologías y arquitectura definidas por el MINSAP y el proyecto, contribuyeron al desarrollo de la solución.
- ✓ Se documentaron los artefactos para el Modelo del negocio, Modelo del sistema, Modelo de análisis y Modelo del diseño, para el desarrollo del módulo de Especialidades.
- ✓ Se automatizó el Módulo de Especialidades para la Gestión de Información de la Formación Posgraduada en Salud mediante una aplicación web.
- ✓ Se integró el Módulo de Especialidades al “Sistema de Gestión de Información en el Proceso de Formación Posgraduada en Salud”.

RECOMENDACIONES

Tras realizar la investigación en cuestión y dar cumplimiento a los objetivos trazados, se les recomienda a todos los interesados:

- ✓ Realizar el manual de usuarios como documentación adjunta a la aplicación desarrollada para un mejor entendimiento de sus funcionalidades.
- ✓ Realizar futuras versiones, incorporándole nuevas funcionalidades, adecuándolo a la demanda creciente de exigencias de los usuarios.
- ✓ Desarrollar una funcionalidad que permita actualizar la tabla caché de los datos generales de los profesionales que se almacena en la base de datos con respecto al RPS.
- ✓ Implementar un servicio web que muestre las especialidades y programas que tiene asociados para servir a otros Sistemas ya funcionales.
- ✓ Realizar la implantación del producto en la Facultades o Institutos Médicos como prueba piloto.

REFERENCIAS BIBLIOGRÁFICAS

1. Vidal Ledo M, de Armas Y. Estrategias de informatización del sector de la salud (I). Rev Infor Méd. 2002;4(11):24-7.
2. Informática en la salud pública cubana. [En línea] [Citado el: 20 de Febrero de 2008.] http://bvs.sld.cu/revistas/spu/vol32_3_06/spu15306.htm.
3. Marín Díaz. Miguel E. Fundamentos del Sistema de salud Pública en Cuba para estudiantes de Informática. La Habana. Cuba 2006. p 3.
4. Delgado Ramos, Ariel. *Presentación Informatización del Sistema Nacional de Salud*. Dirección Nacional de Registros Médicos y Estadísticas de Salud. Ministerio de Salud Pública. La Habana. Cuba. 2006
5. Reyes Miranda D, Hatim Ricardo A. El Diplomado en sistema nacional de salud. Educ Med Super.2002; 16(2):85-93.
6. Gestión de información en salud. [En línea] 20 de febrero de 2008. http://bvs.sld.cu/revistas/aci/vol14_4_06/aci01406.htm.
7. Acerca de las especialidades. [En línea] [Citado el: 10 de Junio de 2008.] <http://www.odontologia-online.com/casos/part/GRM/GRM01/grm01.html>.
8. Modelo Cliente-Servidor disponible en: <http://www.csi.map.es/csi/silice/Global71.html>
9. Arquitectura orientada a servicios (SOA) disponible en: http://es.wikipedia.org/wiki/Arquitectura_orientada_a_servicios
10. Servicio Web disponible en: <http://sistemas3.wordpress.com/2007/06/11/web-service.htm>
11. Arquitectura en tres capas disponible en: http://es.wikipedia.org/wiki/Arquitectura_de_tres_niveles
12. Patrones Grasp disponible en: <http://jorgesaavedra.wordpress.com>

- 13 AJAX disponible en: <http://es.wikipedia.org/wiki/AJAX>
- 14 Plantilla de motor Smarty disponible en: <http://es.wikipedia.org/wiki/Smarty>
- 15 Framework de diseño(YUI) disponible en: http://es.wikipedia.org/wiki/Yahoo!_UI_Library.
- 16 Lenguaje de programación web PHP disponible en: <http://es.wikipedia.org/wiki/PHP>
- 17 Lenguaje JavaScrip disponible en: <http://es.wikipedia.org/wiki/JavaScript>
- 18 Lenguaje HTML disponible en: <http://es.wikipedia.org/wiki/HTML>
- 19 Sistema Gestor de BD disponible en:
http://es.wikipedia.org/wiki/Sistemas_gestores_de_bases_de_datos#Objetivos
- 20 MySQL disponible en: <http://www.esepestudio.com/articulo/desarrollo-web/bases-de-datos-mysql/Que-es-MySQL.htm>
- 21 Servidor Apache disponible en:
<http://www.linuxparatodos.net/portal/staticpages/index.php?page=servidor-web>
- 22 Lenguaje Unificado de Modelado disponible en:
http://es.wikipedia.org/wiki/Lenguaje_Unificado_de_Modelado
- 23 Herramienta Rational Rose Enterprise Edition disponible en:
<http://www.vico.org/TallerRationalRose.pdf>
- 24 Herramienta Dreamweaver 8 disponible en:
http://es.wikipedia.org/wiki/Adobe_Dreamweaver
- 25 Zend estudio disponible en: <http://www.desarrolloweb.com/articulos/1178.php>

REFERENCIAS BIBLIOGRÁFICAS

26 Control de versiones disponible en: http://es.wikipedia.org/wiki/Control_de_versiones

27 TortoiseSVN disponible en: <http://es.wikipedia.org/wiki/TortoiseSVN>

BIBLIOGRAFÍA

Acerca de las especialidades. [En línea] [Citado el: 10 de Junio de 2008.] <http://www.odontologia-online.com/casos/part/GRM/GRM01/grm01.html>.

Arquitectura en tres capas disponible en: http://es.wikipedia.org/wiki/Arquitectura_de_tres_niveles

Arquitectura orientada a servicios disponible en: http://es.wikipedia.org/wiki/Arquitectura_orientada_a_servicios.

Control de versiones disponible en: http://es.wikipedia.org/wiki/Control_de_versiones.

Delgado Ramos, Ariel. *Presentación Informatización del Sistema Nacional de Salud*. Dirección Nacional de Registros Médicos y Estadísticas de Salud. Ministerio de Salud Pública. La Habana. Cuba. 2006.

FRANCO NAVARRO, J. A. 2005. *UML en acción. Modelando Aplicaciones Web*. 2005.

GALLEGO VÁZQUEZ, J. A. 2003. *Desarrollo Web con PHP y MySQL*. 2003.

"Gestión de información en salud". [En línea] 20 de febrero de 2008. http://bvs.sld.cu/revistas/aci/vol14_4_06/aci01406.htm.

HANSEN, G. W. H., JAMES V. *Diseño y Administración de Bases de Datos*.

Herramienta Dreamweaver 8 disponible en: http://es.wikipedia.org/wiki/Adobe_Dreamweaver.

Hernández, Pedro V. El Proceso Unificado de Racional (RUP) y su relación con las técnicas y métodos de la ingeniería y usabilidad del software. [En línea] [Citado el: 14 de 02 de 2008.] <http://is.ls.fi.upm.es/doctorado/Trabajos20042005/Hernandez.pdf>.

Herramienta Rational Rose disponible en: <http://www.vico.org/TallerRationalRose.pdf>.

Herramienta TortoiseSVN disponible en: <http://es.wikipedia.org/wiki/TortoiseSVN>.

Herramienta Zend estudio disponible en: <http://www.desarrolloweb.com/articulos/1178.php>.

Informática en la salud pública cubana. [En línea] [Citado el: 20 de Febrero de 2008.] http://bvs.sld.cu/revistas/spu/vol32_3_06/spu15306.htm.

Lenguaje de programación web disponible en: <http://es.wikipedia.org/wiki/PHP>.

Lenguaje HTML disponible en: <http://es.wikipedia.org/wiki/HTML>.

Lenguaje JavaScript disponible en: <http://es.wikipedia.org/wiki/JavaScript>.

Lenguaje Unificado de Modelado disponible en: http://es.wikipedia.org/wiki/Lenguaje_Unificado_de_Modelado.

Marín Díaz. Miguel E. Fundamentos del Sistema de salud Pública en Cuba para estudiantes de Informática. La Habana. Cuba 2006. p 3.

Modelo Cliente-Servidor disponible en: <http://www.csi.map.es/csi/silice/Global71.html>.

MySQL disponible en: <http://www.esepestudio.com/articulo/desarrollo-web/bases-de-datos-mysql/Que-es-MySQL.htm>.

Patrones Grasp disponible en: <http://jorgesaavedra.wordpress.com>.

Plantilla de motor Smarty disponible en: <http://es.wikipedia.org/wiki/Smarty>.

PRESSMAN, R. 2002. *Ingeniería del Software. Un enfoque práctico*. 2002.

Reyes Miranda D, Hatim Ricardo A. El Diplomado en sistema nacional de salud. *Educ Med Super*.2002; 16(2):85-93.

RUMBAUGH, J. J., IVAR. 2000. *El lenguaje unificado de modelado*. 2000.

Servicio Web disponible en: <http://sistemas3.wordpress.com/2007/06/11/web-service.htm>.

Servidor Apache disponible en:
<http://www.linuxparatodos.net/portal/staticpages/index.php?page=servidor-web>.

Sistemas gestores de BD disponible en:
http://es.wikipedia.org/wiki/Sistemas_gestores_de_bases_de_datos#Objetivos.

Técnica de desarrollo web AJAX disponible en: <http://es.wikipedia.org/wiki/AJAX>.

Vidal Ledo M, de Armas Y. Estrategias de informatización del sector de la salud (I). *Rev Infor Méd.* 2002;4(11):24-7.

ANEXO 1: MODELO DE NEGOCIO

Especificación de los Casos de Uso del Negocio.

Caso de uso del negocio	Solicitud de matrícula de la especialidad
Actores	Profesional de Salud.
Resumen	El caso de uso se inicia cuando el profesional de salud solicita la matrícula de la especialidad al metodólogo de la provincia. Verifica una serie de datos y si cumple con todos los requisitos se le hace la matrícula y se le orienta donde le corresponde hacer su formación docente.
Casos de uso asociados	
Acción del actor	Respuesta del proceso de negocio
1. El profesional solicita la matrícula de la especialidad.	2. Solicita los datos.
3. Entrega los datos.	4. Verifica si es graduado universitario.
	5. Busca datos de los profesionales.
	6. Verifica si esta autorizado y avalado por el centro de trabajo.
	7. Verifica exigencia de ingreso.
	8. Realiza la matrícula en la especialidad.
	9. Comunica al profesional que esta matriculado.
10. Se retira.	
Otras secciones	-
Mejoras propuestas	Se encuentran las matrículas de todos los profesionales de forma automatizada.

Caso de uso del negocio	Gestión de formación docente.
Actores	Profesional de Salud.
Resumen	El caso de uso se inicia cuando el profesional realiza sus exámenes y el secretario procede a buscar en los archivos de información los datos concernientes a él. Una vez encontrados registra las notas y actualiza su perfil.

ANEXO 1 MODELO DEL NEGOCIO

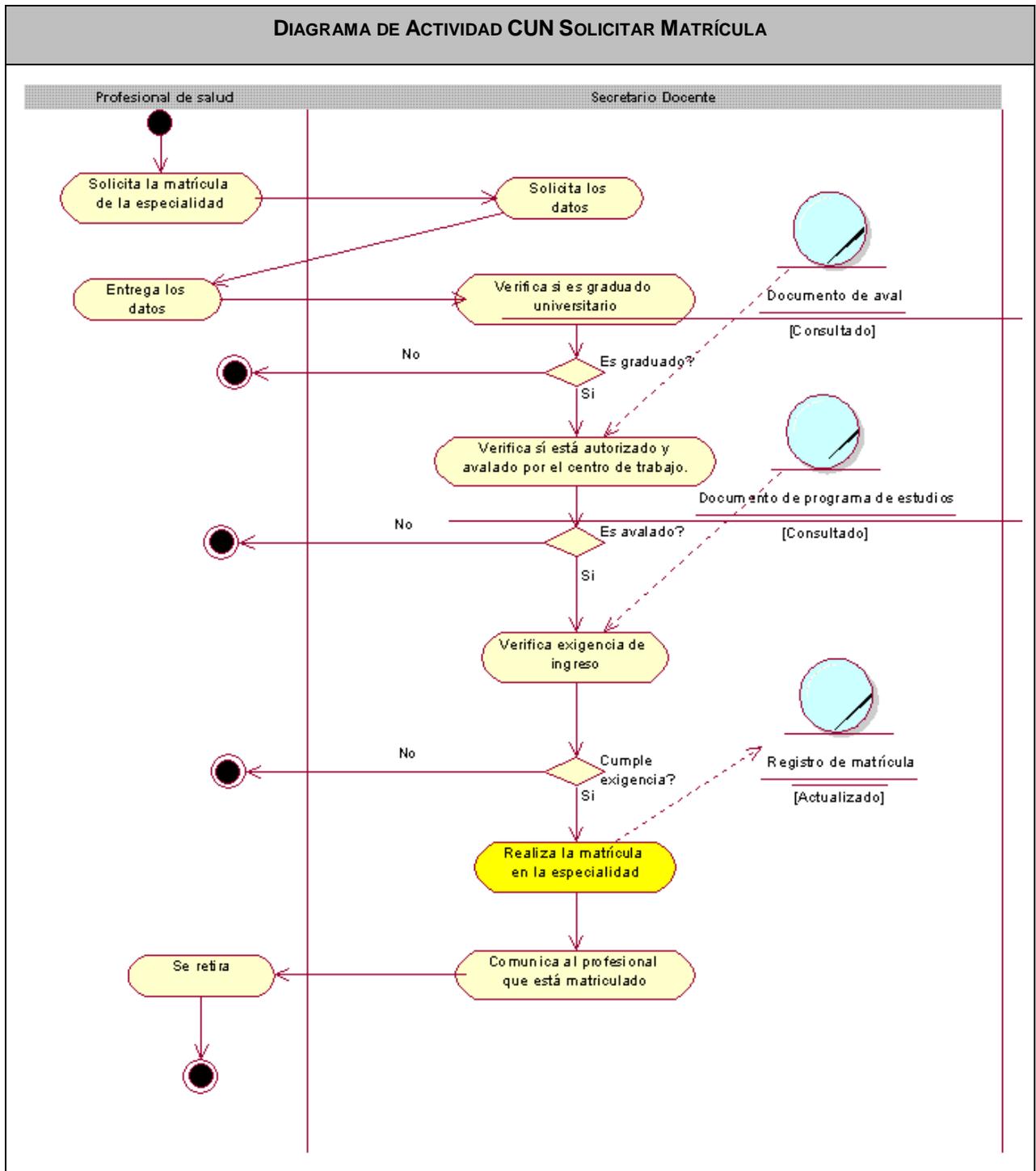
Casos de uso asociados	
Acción del actor	Respuesta del proceso de negocio
1. Realiza examen el profesional.	2. Busca el profesional.
	3. Controla la formación docente.
	4. Archiva las notas. 5. Envía un mensaje de registro de las notas.
6. Recibe mensaje.	
Otras secciones	-
Mejoras propuestas	Se archivan las notas en un sistema automatizado.

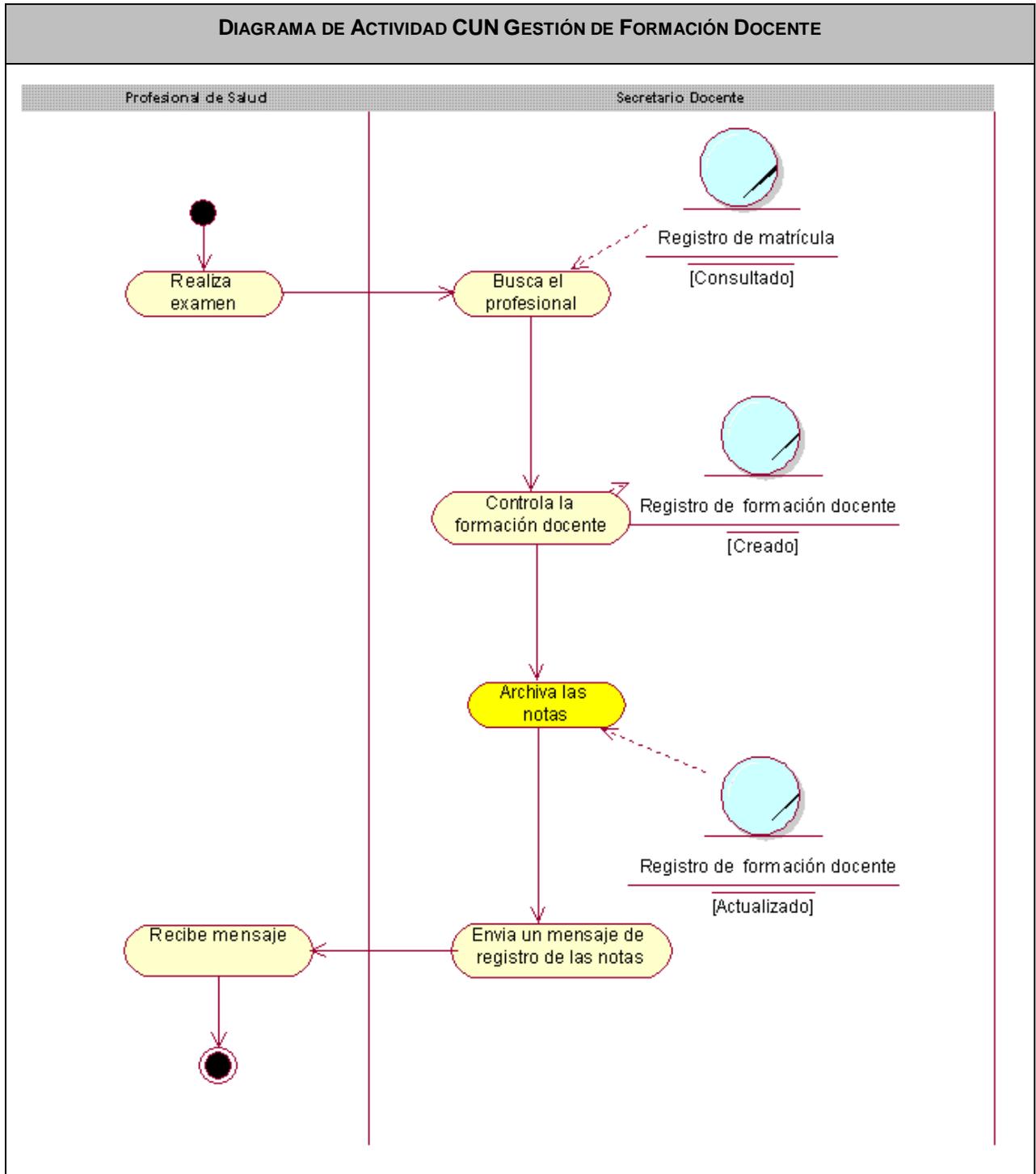
Caso de uso del negocio	Solicitud de movimiento.
Actores	Profesional de Salud
Resumen	El caso de uso se inicia cuando el solicitante pide un trámite, ya sea una baja temporal o definitiva. El secretario procede a informar la solicitud a la facultad analizando las causas. Este registra el movimiento en el archivo de trámites y comunica al profesional si fue tramitado.
Casos de uso asociados	
Acción del actor	Respuesta del proceso de negocio
1. El profesional pide un trámite.	2. Informa solicitud a la facultad.
	3. Analiza las causas.
	4. Registra trámites según las causas.
	5. Comunica el trámite al profesional.
6. Recibe la información del trámite.	
Otras secciones	-
Mejoras propuestas	Se archivan los trámites en un sistema automatizado.

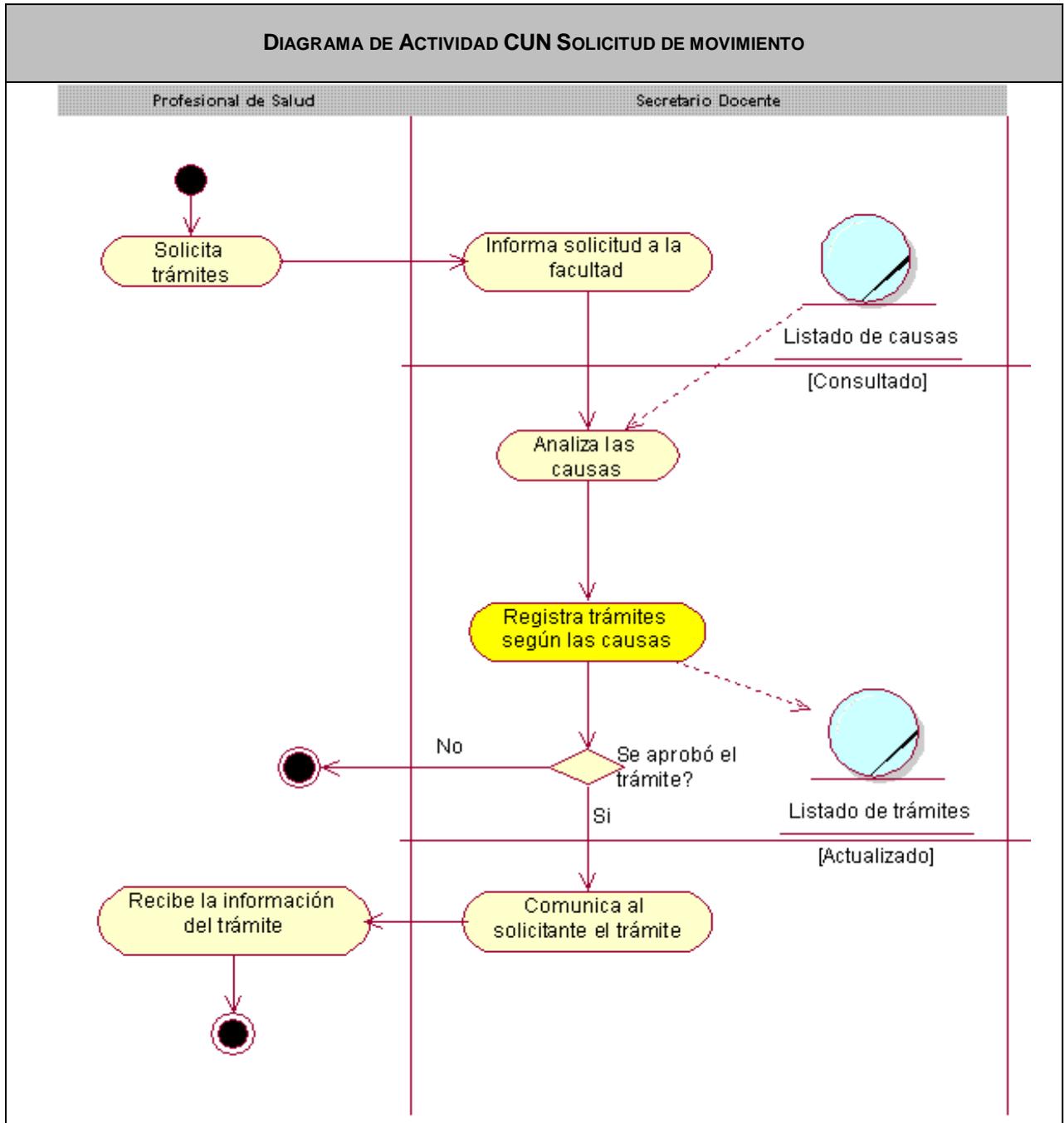
ANEXO 1 MODELO DEL NEGOCIO

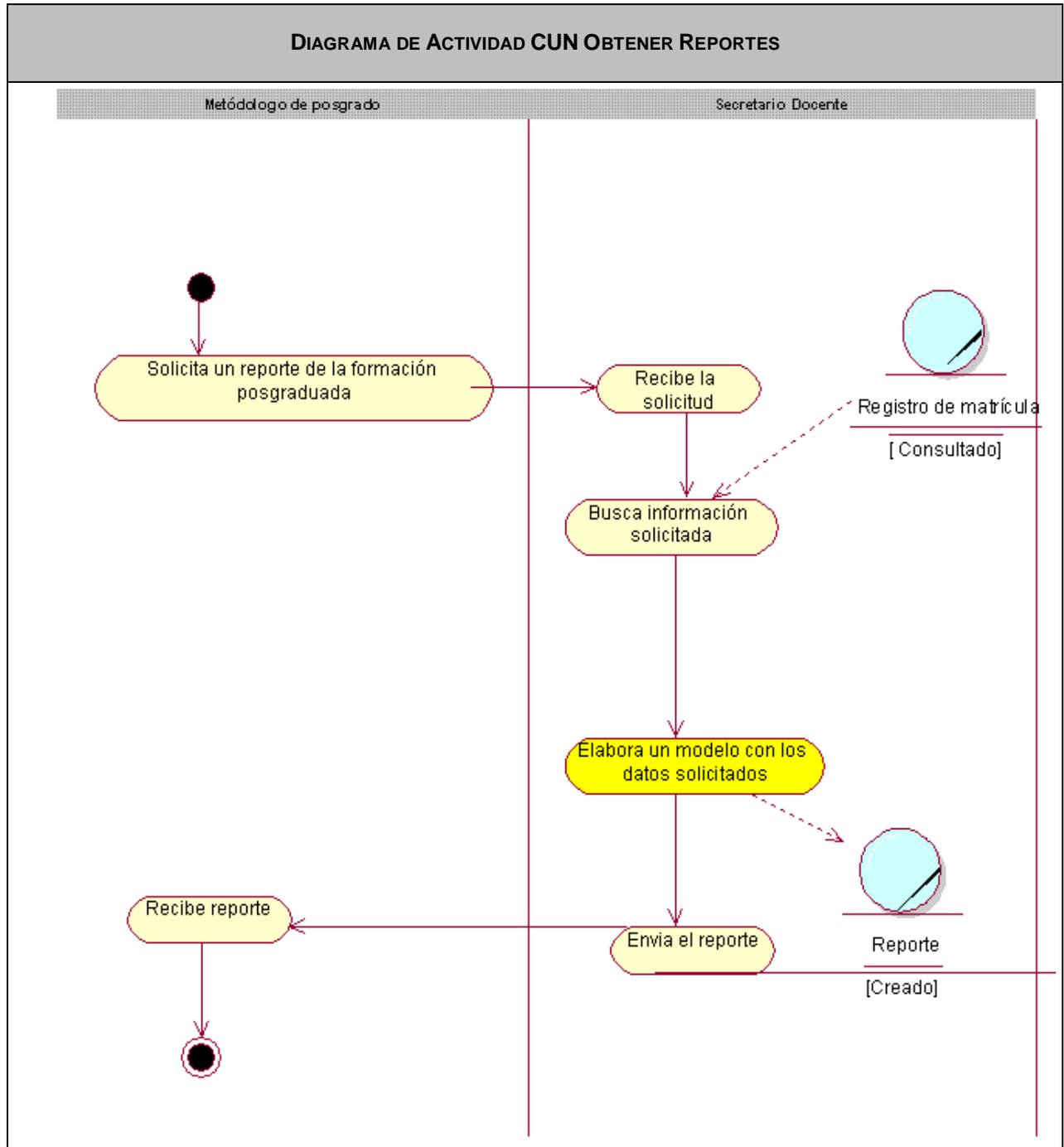
Caso de uso del negocio	Obtener reportes.
Actores	Metodólogo de posgrado.
Resumen	El caso de uso se inicia cuando el Metodólogo pide un reporte de la información estadística de la formación posgraduada, el secretario docente procede a buscar la información en el registro de matrícula. Elabora un modelo con los datos solicitados y envía el reporte al metodólogo.
Casos de uso asociados	
Acción del actor	Respuesta del proceso de negocio
1. Solicita un reporte de la formación posgraduada.	2. Recibe la solicitud del reporte.
	3. Busca la información solicitada.
	4. Elabora un modelo con los datos solicitados.
	5. Comunica el trámite al profesional.
	6. Envía el reporte.
7 Recibe el reporte.	
Otras secciones	-
Mejoras propuestas	Se generan los reportes automáticamente.

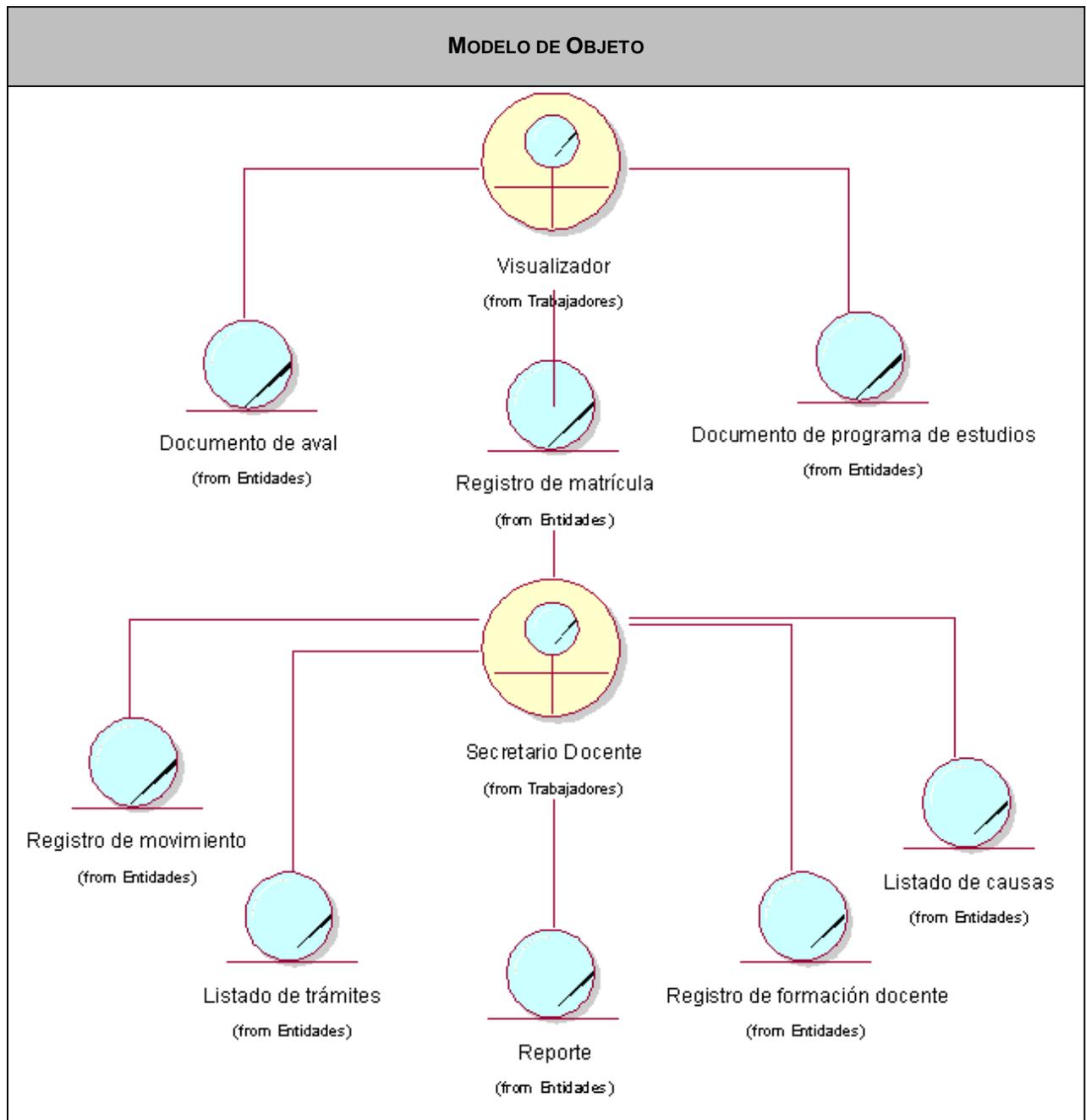
Diagrama de Actividades





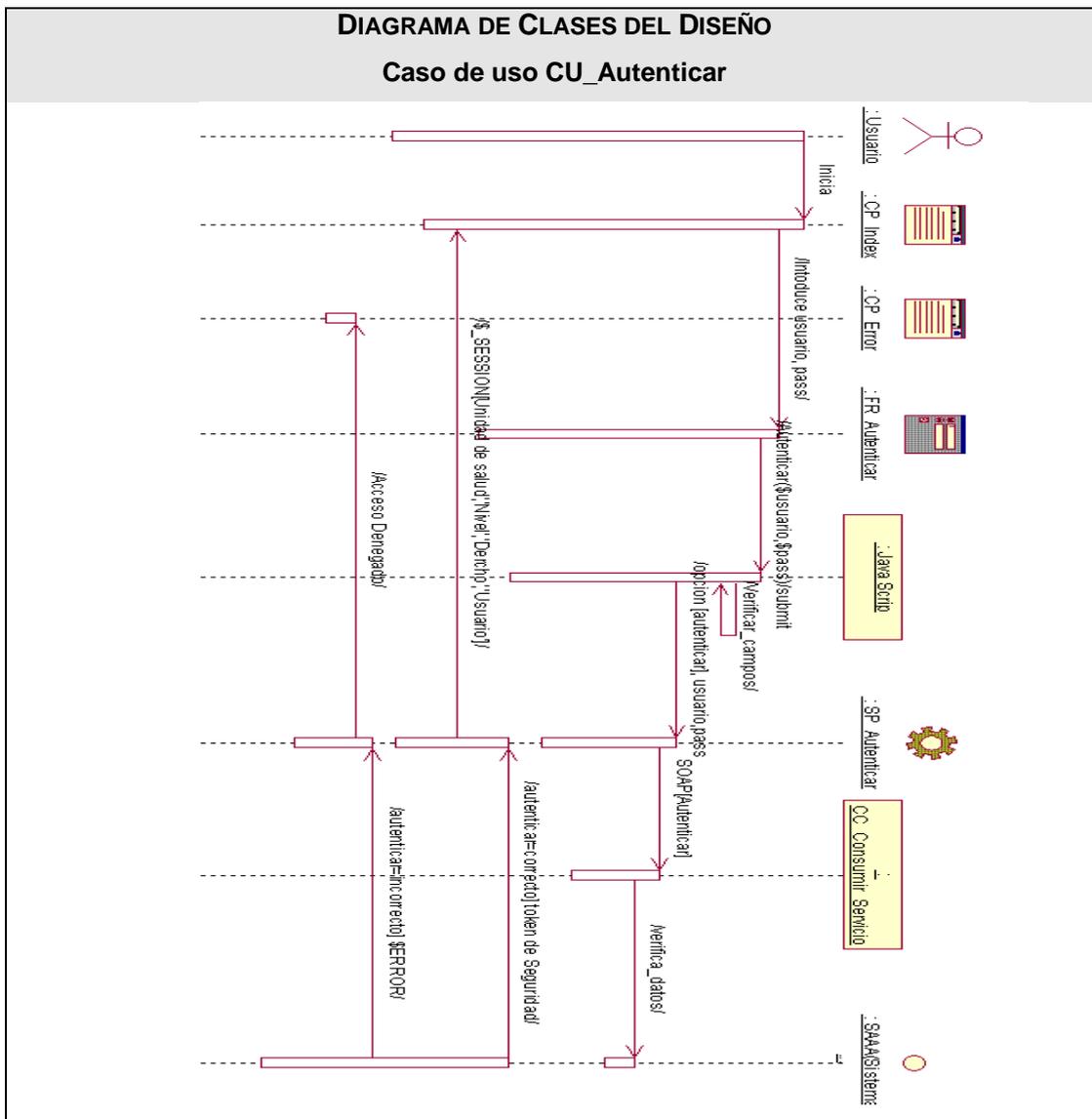


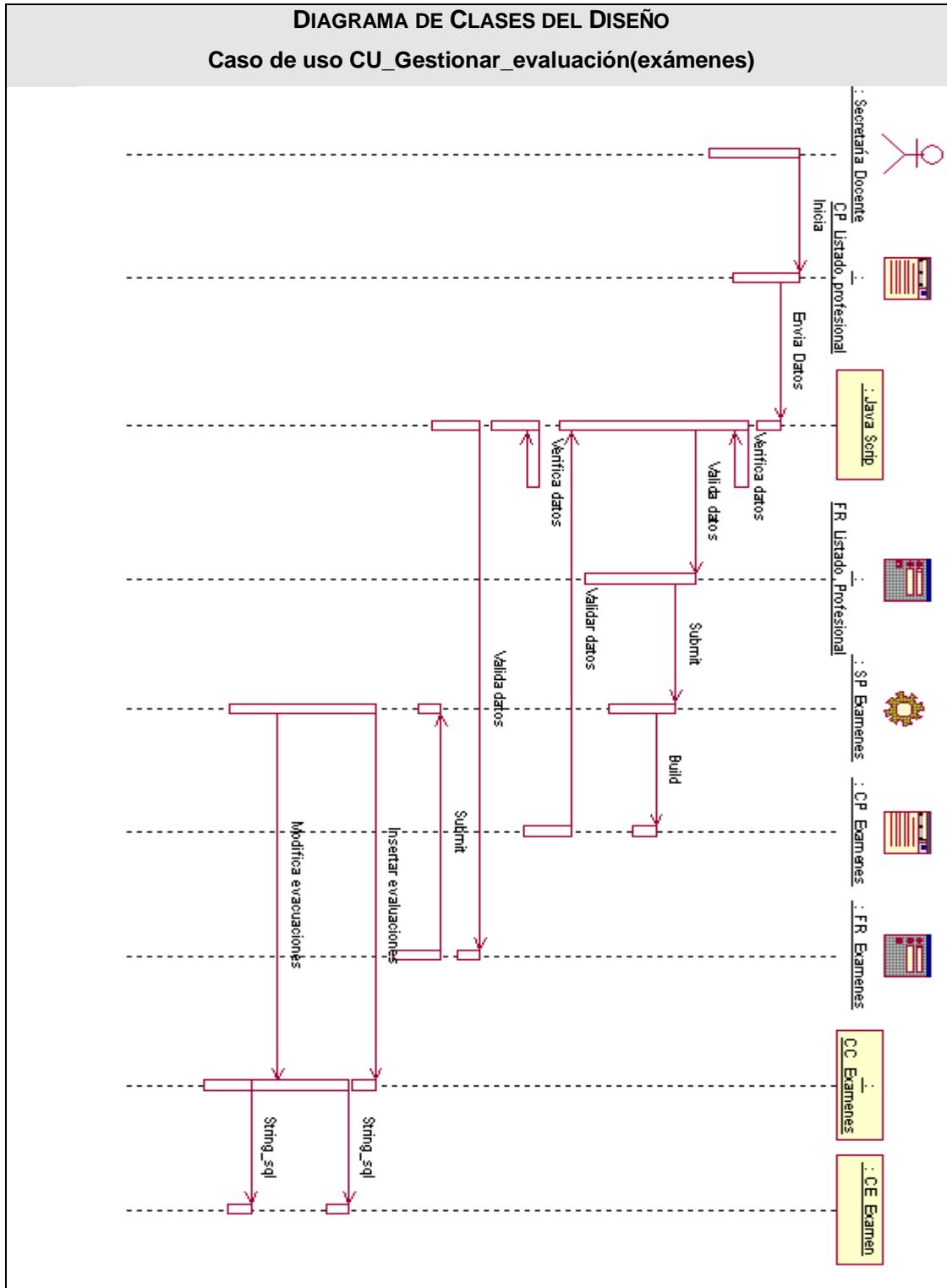


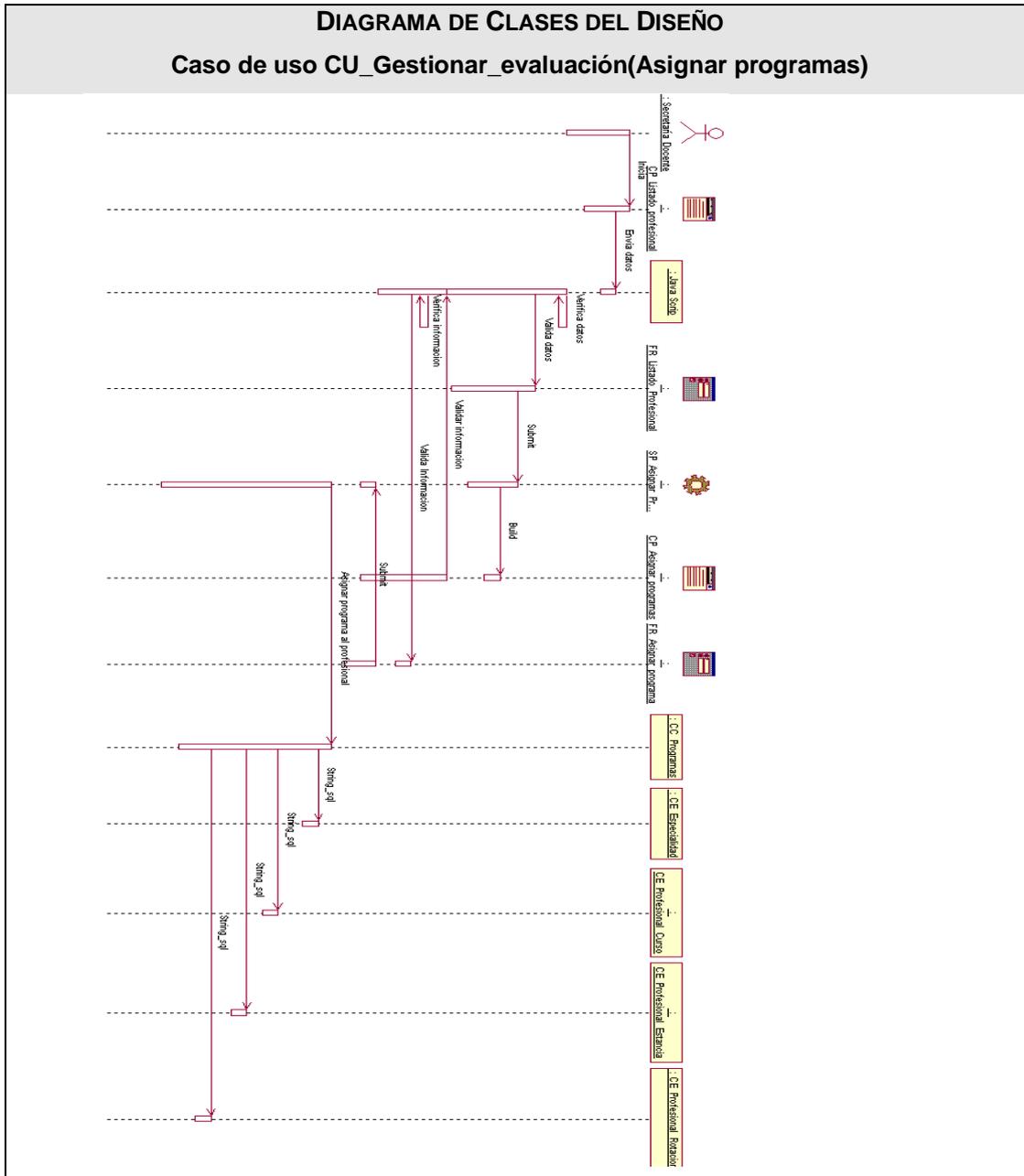


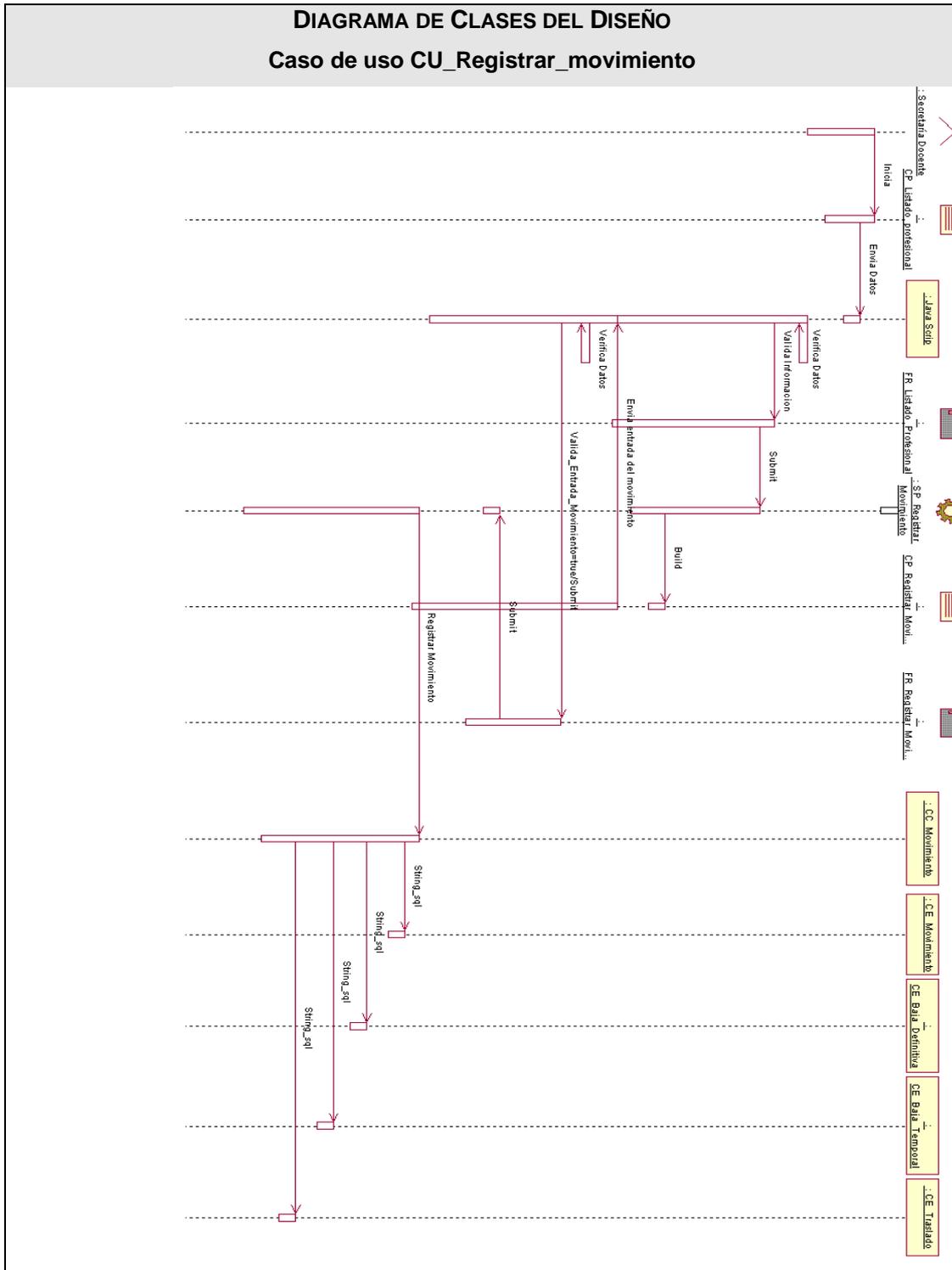
ANEXO 2: MODELO DE DISEÑO

Para reflejar como fluye el intercambio de mensajes entre las clases que tienen ya definidas las operaciones que en la parte de análisis eran sólo 'frases', se realizan los diagramas de interacción. A continuación se presentan los diagramas de los casos de uso correspondientes.









Informática: Disciplina que estudia el tratamiento automático de la información utilizando dispositivos electrónicos y sistemas computacionales.

Instituto Superior de Ciencias Médicas: Es un centro universitario de las ciencias médicas localizado en un territorio, que tiene como función la orientación rectora de la formación de pregrado y posgrado y desarrolla sus actividades a través de las facultades que tienen subordinadas.

Facultad de Ciencias Médicas: Es un centro universitario de las Ciencias Médicas que tiene la misión de formar integralmente los profesionales de la salud en el pregrado y posgrado, para lograr que se brinden servicios con la calidad requerida en el ámbito nacional e internacional.

Facultad Dependiente: Es la facultad de Ciencias Médicas que está subordinada a un Instituto Superior de Ciencias Médicas.

Facultad Independiente: Es la facultad de Ciencias Médicas que tiene las funciones inherentes a un Instituto Superior de Ciencias Médicas, exceptuándose las relacionadas con los procesos de graduación.

Ministerio de Salud Pública (MINSAP): Organismo rector del Sistema Nacional de Salud. Encargado de dirigir, ejecutar y controlar la aplicación de la política del Estado y del Gobierno en cuanto a la Salud Pública, el desarrollo de las Ciencias Médicas y la Industria Médico Farmacéutica.

Niveles de dirección: Se divide en unidades de salud que es lo más reducido, luego a nivel municipal que incluye todas las unidades asistenciales que le pertenecen a un determinado municipio, después a nivel provincial que incluye todos los municipios de una determinada provincia y lo superior que es el MINSAP que es el órgano rector de la salud en Cuba que sería el nivel de atención nacional.

Posgrado: Estudios de especialización posterior al grado o licenciatura. El posgrado o postgrado, es un nivel educativo que forma parte del tipo superior o de tercer ciclo; es la última fase de la educación formal, tiene como antecedente obligatorio la licenciatura o pregrado.

Servidor: Son ordenadores que actúan como "almacenes" de información. Esta información es solicitada por los ordenadores cliente y el servidor responde a tales peticiones devolviendo los datos solicitados.

Servicio Web: Es una colección de protocolos y estándares que sirven para intercambiar datos entre aplicaciones.

Sistema Nacional de Salud (SNS): Sistema universal, gratuito, accesible, regionalizado e integral, al alcance de todos los ciudadanos en el campo y en la ciudad, independientemente de su filiación política, raza, sexo o creencia religiosa; con una amplia participación social e intersectorial y una profunda concepción internacionalista.

Software: Conjunto de programas y procedimientos necesarios para hacer posible la realización de una tarea específica, en contraposición a los componentes físicos del sistema.

Software Libre: Es el software que, una vez obtenido, puede ser usado, copiado, estudiado, modificado y redistribuido libremente.