

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 de Recursos Humanos en Salud.

Módulo de Salidas

Autores:

Mildred Paneca Martínez

Yisel Reyes Cardoso

Alexis José Turruella Sánchez

Tutora: Ing. Yenisel Valido Pérez

Asesor: Lic. Julio Antonio Tejera Castillo

Ciudad de La Habana, Junio 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 20 días del mes de Junio del año 2008.

Autores:

Mildred Paneca Martínez

Yisel Reyes Cardoso

Alexis José Turruella Sánchez

Tutora:

Ing. Yenisel Valido Pérez

*Así como el hierro se oxida por falta de uso, también la
inactividad destruye el intelecto.*

Leonardo da Vinci

DATOS DE CONTACTO

Ing. Yenisel Valido Pérez. (yenisel@uci.cu). Graduada de Ingeniería Informática en el Instituto Superior Politécnico “José Antonio Echeverría” (ISPJAE) y la Universidad de Cienfuegos “Carlos Rafael Rodríguez”. Profesora en tercer año de adiestramiento en la Universidad de las Ciencias Informáticas, pertenece al departamento de Ingeniería y Gestión de Software, imparte las asignaturas de Ingeniería de Software y se desempeña como jefa de la misma en la facultad siete. Profesora del Área Temática Sistemas de Apoyo a la Salud, líder del proyecto Docencia Médica.

AGRADECIMIENTOS

Queremos agradecer:

A nuestros queridos Comandante Fidel y Raúl y a la UCI por hacernos partícipes de este proyecto de la Revolución.

A nuestros amigos y compañeros que vienen con nosotros desde primer año por brindarnos su apoyo incondicional, y por estimularnos para seguir creciendo intelectualmente.

A nuestra tutora de tesis Yenisel Valido y asesor Julio Antonio Tejera por su apoyo y dedicación.

A todas aquellas personas que de alguna manera confiaron en nosotros desde un principio y han sido parte de los éxitos alcanzados hasta el momento.

De Mildred:

A mi familia principalmente a mi madre querida Carmen Regla Martínez Padrón por darme el amor y el apoyo incondicional durante toda mi vida y ser además de madre mi amiga.

A mis abuelos, que a pesar de no tenerlos físicamente entre nosotros, fueron personas que llenaron de amor toda mi vida, para ellos todo mi amor.

A mi papá Enrique que se ha esforzado porque yo sea quien soy hoy, a mi papá Enrique Paneca y a mi abuela Rosa que a pesar de estar lejos se preocupan por mí, a mi tío Eloy y a mi hermano Enriquito.

A mis amigos y mis vecinos, a mis compañeros de grupo que han sido como mi segunda familia durante estos 5 años.

A mis compañeros de tesis Yisel y Alexis.

A mi novio por estar a mi lado, por darme su amor y su apoyo.

A todos los que de una forma u otra contribuyeron con mi formación personal y profesional muchas gracias.

De Yisel:

A mis queridos Padres Nancy y Rafael por estar conmigo en cada paso que di a pesar de la distancia, por darme aliento en los momentos más difíciles por tantos y tantos sacrificios que han hecho a lo largo de toda mi vida, por sus desvelos cuando tenía algún problema, por tanto apoyo, amor y dedicación.

A mi hermano Yoel que me quiere y me ha apoyado mucho.

A mi abuelo Reynaldo que aunque no está conmigo hoy se que se sentiría orgulloso de mí. A mi abuela Teresa, a mis tíos y tías especialmente a mi tía Mireya por su preocupación, por confiar en mí y darme cada día más ánimo a seguir y triunfar en mis estudios.

A mis primos y mi prima querida Yayi y a toda mi familia y vecinos por su colaboración y sobre todo por su valiosa amistad.

Agradecer hoy y siempre a aquellas personas que me han servido de soporte y compañía durante toda la carrera, a Maryis, Yaima, a mis compañeros de tesis Ale y Mildred, a todos mis compañeros de brigada y proyecto, a mis profesores.

A todos los que han hecho que este sueño se haga realidad.

De Alexis:

A mi novia por estar a mi lado en los buenos y malos momentos, por su paciencia y dedicación.

A mi piquete por ser el más unido y siempre estar donde uno lo necesita.

Al todos los miembros del grupo, a los presentes y no presentes actualmente porque de todos se aprende algo nuevo, en especial Pepe, Frank, Alfredo, Héctor, Ramos.

A William por sus valiosos consejos y su confianza en mi, confianza que me ha enseñado mucho.

Al tribunal de mi tesis por desde un inicio enseñarnos a perfeccionar el resultado final.

DEDICATORIA

*A mis queridísimos padres Rafael y Nancy, a mi hermano Yoel
y a toda mi familia por ser parte de mi corazón, por su
apoyo incondicional y su gran amor.
Con todo mi amor y cariño.*

Yisel

*Especialmente a la persona que más me ha apoyado en la vida y siempre soñó
con este día, que hoy es realidad, mi mamá.
A mi familia por ser la mejor del mundo.
A todos los involucrados en este trabajo,
al MINSAPE y a los especialistas que atienden
directamente este proyecto, que esperan con
anhelo la culminación de
este trabajo.*

Alexis

*Dedico este trabajo de diploma a mis queridísimos padres, por siempre guiarme
por el mejor camino, por su enorme sacrificio,
dedicación y por confiar en mí.
A toda mi familia y amigos por su apoyo y
cariño y a mis abuelos queridos.*

Mildred

RESUMEN

El presente trabajo tiene como objetivo facilitar el control y acceso de la información estadística de los estudiantes del perfil de salud durante su formación, mediante el desarrollo de un Módulo de Salidas para el Sistema de Gestión de Información en el Proceso de Formación de Recursos Humanos en Salud. Se ha realizado una investigación de los procesos de recolección de datos en la matrícula de estudiantes en los centros de enseñanza de salud; con la finalidad de brindar **resúmenes, listados y modelos oficiales** con información estadística a las secretarías docentes de las instituciones clínicas docentes cubanas y a los distintos niveles del Sistema Nacional de Salud.

Para el desarrollo del presente módulo se aplican las tecnologías y arquitectura definidas por el área temática Sistemas de Apoyo a la Salud, basada en tecnologías WEB, corriendo sobre un servidor LAMP (*Linux, Apache, MySQL, PHP*) y utilizando herramientas de software libre. La información estará almacenada en una base de datos a la que se accederá para realizar las consultas necesarias en dependencia de la solicitud del usuario.

El Módulo de Salidas facilitará la toma de decisiones con respecto a los recursos humanos que se encuentran formándose en los centros de enseñanza médica de la salud del país. Además, el proceso de obtención de la información se podrá realizar con más periodicidad, economizando el tiempo de trabajo en las secretarías docentes, brindando la información precisa de forma rápida y eficiente.

PALABRAS CLAVES

Pregrado, Sistema, Aplicación, Módulo, Recursos Humanos, Información estadística.

TABLA DE CONTENIDOS

TABLA DE CONTENIDOS	I
INTRODUCCIÓN	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA	5
1.1 Generación de reportes.....	5
1.2 Generalidades sobre los generadores de reportes.	5
1.3 Antecedentes de sistemas de gestión universitaria con generadores de informes.	6
1.4 Aspectos conceptuales.....	9
1.5 Tendencias y tecnologías actuales para dar solución al problema.	11
1.6 Lenguajes de programación para la Web.	13
1.7 Navegador recomendado: Mozilla Firefox.	16
1.8 Servidor WEB: Apache.....	17
1.9 Sistemas Gestores de base de datos.	17
1.10 Motor de plantillas utilizado: Smarty.	18
1.11 Metodologías de desarrollo del software.	20
1.12 Herramientas utilizadas.	21
CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA	24
2.1 Objeto de estudio.	24
2.2 Documentos que se procesan.....	27
2.3 Propuesta del sistema.....	28
2.4 Análisis comparativo de otras soluciones con la propuesta.	29
2.5 Modelo de negocio.....	29
2.6 Especificación de los requisitos de software.....	32
2.7 Definición de los Casos de Uso.....	38
CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA	41
3.1 Análisis.....	41
3.2 Diseño.....	45
3.3 Definiciones de diseño.	66
3.4 Tratamiento de errores.	67
3.5 Seguridad.....	68
3.6 Interfaz.....	69
3.7 Concepción de la ayuda.....	70
CAPÍTULO 4: IMPLEMENTACIÓN DEL SISTEMA	72
4.1 Diagramas de componentes.....	72
4.2 Diagrama de despliegue.....	76
CONCLUSIONES	78
RECOMENDACIONES	79

BIBLIOGRAFÍA.....	80
REFERENCIAS BIBILOGRÁFICAS.....	82
ANEXO # 1: MODELO DE NEGOCIO.....	85
ANEXO # 2: MODELO DE SISTEMA.....	92
ANEXO # 3: MODELO DE DISEÑO.....	98

INTRODUCCIÓN

La vertiginosa evolución de los medios informáticos en los últimos años, hace imprescindible dotar al futuro profesional y a todo el pueblo del manejo de herramientas de computación; no como un mero usuario, sino con formación suficiente, que le permita tomar decisiones sobre la conveniencia o no de su utilización en cada caso, es decir, buen uso y obtención de un mayor beneficio del capital que se invierte.

Por tales razones el Estado Cubano lleva a cabo el proceso de informatización de la sociedad, para elevar el nivel cultural y la calidad de vida del pueblo. Se ha incrementado el uso de las nuevas tecnologías de la información y la comunicación como parte de las tecnologías emergentes que habitualmente suelen identificarse con las siglas TIC. Estas hacen referencia a la utilización de medios informáticos para almacenar, procesar y difundir todo tipo de información, procesos de formación educativa y los datos necesarios para cualquier actividad humana.

La instrumentación tecnológica es una prioridad en la comunicación de hoy en día, se considera a la tecnología como una extensión de la innovación del ser humano para poder desenvolverse mejor en su medio. Éstas poseen la característica de ayudar a comunicarse porque se desaparecen las distancias geográficas y el tiempo. La tecnología no debe evaluarse, sino en términos de capacidad, de procesamiento de información y de creación de redes que beneficien a todos y que puedan contribuir a elevar las condiciones de vida. [1]

Para guiar este proceso en la Resolución Económica del V Congreso del PCC se elaboró la política para la Informatización de la Sociedad que incluye como objetivo principal: Apoyar el desarrollo e implementación de sistemas verticales de informatización, la Informática aplicada a la gestión económica y de recursos humanos; el fomento y establecimiento de redes de computadoras, sistemas de control automático (en los casos que proceda), que contribuyan a una mayor eficacia en la dirección y en la organización eficiente de la actividad productiva o de servicios. Por tal motivo se ha extendido la informatización no solo a la informática y las comunicaciones sino también a diversos sectores como la Educación y la Salud, para elevar la calidad de los servicios prestados a la sociedad. [2]

Actualmente existe una gama muy amplia de sistemas de información desarrollados para satisfacer las

necesidades de la mayoría de las empresas y organismos. El Ministerio de Salud Pública no está ajeno a estos cambios y se integra al proceso de informatización en el cual la Universidad de las Ciencias Informáticas juega un papel fundamental y especialmente la facultad siete que es la encargada de desarrollar numerosos proyectos para facilitar el trabajo en las instituciones de salud.

En los centros de enseñanza médica de salud, cada año la matrícula de estudiantes extranjeros y cubanos se incrementa en cifras difíciles de manejar, por la introducción de nuevos programas educacionales. Además del modelo tradicional que ya existía, donde se encuentran estudiantes cubanos y extranjeros de las carreras de estomatología y medicina, se introducen ahora los programas especiales. Dentro de estos se encuentran los policlínicos universitarios, la municipalización de la enseñanza, el nuevo modelo de formación de médicos en las carreras de psicología, tecnología de la salud y enfermería, así como curso para trabajadores.

Uno de los más recientes programas de la Revolución y del Sistema Nacional de Salud es el Nuevo Programa de Formación de Médicos Latinoamericanos. Este tiene como objetivo, formar masivamente profesionales de la salud de distintos países, para que puedan una vez graduados, aplicar los conocimientos adquiridos en sus países de origen. El proceso de gestión de la información docente en el sistema de salud surge a partir de la necesidad que tiene el Estado de lograr un control rápido y efectivo de la información real de los estudiantes que tiene en sus universidades médicas. Con la introducción de este nuevo programa, una sede universitaria puede ser desde una facultad hasta un hospital o policlínico docente; estos estudiantes residirán en instituciones escolares, viviendas familiares o residencias colectivas.

Tradicionalmente realizar reportes y gestionar los resultados académicos de un estudiante a lo largo de la carrera de medicina era de forma manual. Surgieron sistemas locales como un acercamiento a la solución del problema, pero se usaban de manera ineficiente y en muchos de los casos todo se procesaba manualmente. No se podía tener un dominio pleno de toda esta información a nivel nacional, debido a que existen numerosas sedes universitarias de ciencias médicas en todo el país y se hacía muy difícil conocer la información general o específica de un centro, así como obtener detalladamente los resultados cuantitativos y cualitativos para poder ofrecer un control de los datos estadísticos de cada institución y realizar resúmenes a distintos niveles.

Debido a esta situación existente surgió la idea de desarrollar un sistema informático que facilitara el proceso: Sistema de Gestión de Información en el Proceso de Formación de Recursos Humanos en Salud. Desde sus inicios, en este sistema se concibieron todos sus módulos, de los cuales algunos ya han sido liberados y permiten gestionar toda la información que antes parecía imposible de manipular. Aunque en una segunda iteración del producto, se requiere automatizar el proceso de control y acceso a la información estadística relacionada con los procesos de gestión docente durante la formación de pregrado para poder acceder a la información de cada institución desde cualquier nivel del sistema nacional de salud.

Dada la situación anterior el **Problema a resolver** radica en: ¿Cómo facilitar el control y acceso a la información estadística relacionada con los procesos de gestión docente durante pregrado?

Para dar solución al problema antes mencionado se propone como **objetivo general**: facilitar el control y acceso de la información estadística de los estudiantes de la salud durante su formación, mediante el desarrollo de un módulo de salidas para el Sistema de Gestión de Información en el Proceso de Formación de Recursos Humanos en Salud.

Para dar cumplimiento al objetivo general se proponen las siguientes **tareas de investigación**:

- Analizar los aspectos conceptuales relacionados con la información estadística de la docencia en pregrado del MINSAP para identificar los requerimientos del sistema.
- Asimilar las tecnologías y arquitecturas definidas para el desarrollo del módulo de Salidas para el Sistema de Gestión de Información en el Proceso de Formación de Recursos Humanos en Salud.
- Diseñar el módulo Salidas.
- Implementar el módulo de Salidas.
- Integrar el módulo de Salidas al Sistema de Gestión de Información en el Proceso de Formación de Recursos Humanos en Salud.

El **objeto de estudio** se centra en el proceso de gestión de información en los centros educacionales de salud durante pregrado teniendo como **campo de acción** el proceso de gestión de información estadística en los centros educacionales de salud durante el período de pregrado.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

Introducción

En este capítulo se realiza un estudio del estado del arte del proceso de gestión de información estadística en los centros educacionales de salud durante el período de pregrado. Se investiga acerca de las tecnologías, metodologías de software, herramientas y técnicas que por sus facilidades fueron escogidas para el desarrollo del módulo de Salidas. Teniendo en cuenta que el mismo va a ser integrado a un sistema y debe lograrse la compatibilidad a la hora de la integración.

1.1 Generación de reportes.

En la actualidad existe una gran cantidad de proyectos enfrascados en el desarrollo de generadores de reportes estadísticos. El generador es la herramienta que permite emitir reportes en un formato personalizado de tal manera que le sea útil al personal al cual va dirigido; igual a cualquier sistema de gestión, aunque con algunas peculiaridades propias de este tipo de aplicación. Concretamente, sirve de puente entre la base de datos y los diseños de reportes que el usuario previamente configura de acuerdo a sus necesidades para filtrar y analizar datos.

1.2 Generalidades sobre los generadores de reportes.

De forma general un generador de reportes se compone de dos elementos básicos, un diseñador de reportes y un motor de generación como se muestra en la figura 1.1.

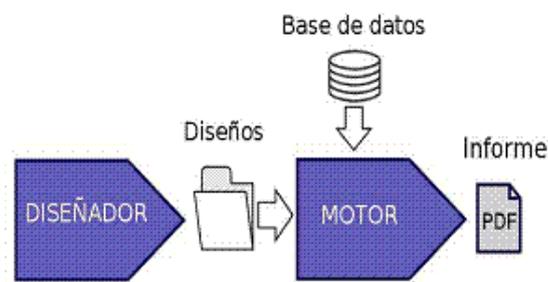


Fig. 1.1. Estructura general de un generador de reportes.

De aquí se puede deducir que las principales tareas que debe realizar un generador de reportes son las siguientes:

- Diseñar la apariencia que van a tener los reportes.
- Extraer la información y mostrarla de forma ordenada con la apariencia diseñada.

Haciendo uso del diseñador se obtienen plantillas de diseños que posteriormente son empleadas por el motor de generación para filtrar datos provenientes de almacenes de datos y obtener el informe final con la apariencia preestablecida en el diseño.

Existen algunos proyectos que sólo desarrollan el motor de generación. En este caso también se considera la aplicación como un generador de reportes, pero se debe tener presente la necesidad de brindar el diseño al motor en algún formato entendible por este. Pues de no contarse con un diseñador, puede resultar un proceso muy engorroso. De aquí que la presencia de un diseñador que brinde flexibilidades para crear los reportes que posteriormente se generarán, sea muy deseable en una aplicación como esta. [3]

1.3 Antecedentes de sistemas de gestión universitaria con generadores de informes.

1.3.1 GestAcad. Sistema para la Gestión Académica.

Dada la importancia que posee el manejo de la información en la toma de decisiones en cualquier proceso de la vida, se han realizado varios intentos de automatizar el procesamiento de la información en las instituciones universitarias del país, ejemplo de ello lo es el Sistema Estudiante y otros sistemas que se han utilizado en los Centros de Educación Superior (CES) cubanos, los que han perdido actualidad con las nuevas tecnologías de la información y las comunicaciones.

Es de destacar que el sistema ha pasado por varias etapas desde su concepción hasta la actualidad lo que ha dado lugar a la aparición ya de dos versiones. La primera fue la creación de módulos Stand Alone para la actualización de los datos (lo que garantizaba un primer nivel de seguridad al no estar

disponibles las vías de actualización de la información de forma pública) y un sitio Web destinado a mostrar reportes de la información almacenada.

La segunda versión, ya en explotación en la Universidad de Matanzas, está concebida para desprenderse de aquellas dependencias de los módulos programados en un lenguaje que no es de código abierto, posibilitando la migración total hacia plataformas de Software Libre. Esta versión está completamente basada en plataforma Web y totalmente programada en PHP. Ya incluye los módulos correspondientes de administración, secretarías y matrícula.

En la actualidad el sistema es capaz de realizar diversas acciones y brindar numerosos reportes los cuales son fruto de los requisitos funcionales del sistema recogidos en la fase de análisis como son:

- Búsqueda de un alumno. (Brinda la ubicación según el horario docente).
- Listado de estudiantes por grupo.
- Reportes dinámicos de la información existente. (Se le da la posibilidad al usuario de seleccionar los campos de datos que desea obtener en el reporte así como el título de este y las condiciones que debe cumplir la información a mostrar).
- Reporte de notas por asignatura y grupo. (Examen final, extraordinario, especial, premio).
- Tabla con los resultados docentes de un grupo en un semestre.
- Reporte de los resultados académicos de un estudiante en toda su carrera. (Hoja de Rendimiento).
- Actas de exámenes de las diferentes asignaturas.
- Registro de características de un grupo de estudiantes.
- Dar baja a un estudiante.

Como se ha explicado, el sistema GestAcad es un acercamiento a la solución de un problema que atañe directamente a la educación superior cubana y que necesita una rápida solución debido a la

calidad, competitividad, prestigio y eficiencia de la educación cubana. La influencia de las nuevas tecnologías de la informática y las comunicaciones en las sociedades actuales es innegable y nosotros no podemos estar ajenos a ello. Si bien se piensa que la ciencia y la tecnología se han convertido en recursos estratégicos, políticos y económicos tanto para los Estados como para las Industrias, no se puede desconocer que el desarrollo tecnocientífico aporta ventajas al bienestar de la sociedad.

1.3.2 Sistemas de Gestión Universitaria desarrollados por el Centro de Cibernética Aplicada a la Medicina (CECAM).

Uno de los primeros Sistemas diseñados e implantados por el CECAM fue el Sistema Automatizado de Información Docente (SAIDO). A partir de su explotación se han tenido que incorporar nuevos módulos y subsistemas para elaborar otros sistemas relacionados, de acuerdo a las necesidades de los usuarios.

SAIDO-F: Sistema automatizado de información docente, facultades. Es un sistema de base de datos que contiene y procesa información sobre los estudiantes de ciencias médicas. Incluye módulos de matrícula, actualización y notas. Salidas para información al Instituto y el MINSAP y permite obtener por selección listados, tablas y totales sobre los datos almacenados.

SAIDO-I: Sistema Automatizado de Gestión docente para Institutos. El sistema es actualizado con los datos de las facultades, las cuales mensualmente envían su información al Instituto, allí es consolidada y se envían al ministerio los modelos requeridos así como se brinda informaciones a los distintos niveles que la solicitan. Entre sus principales salidas fijas están los 13 modelos para el Sistema de Información Estadístico Nacional (SIEN) y un boletín de 42 tablas. [4]

Algunos de estos sistemas se están utilizando en las facultades de ciencias médicas de todo el país, otros por diversas causas han dejado de utilizarse o no se utilizan eficientemente.

1.4 Aspectos conceptuales.

1.4.1 Definición de Pregrado

En el Sistema Nacional de Salud se entiende como pregrado, al proceso de formación de recursos humanos, es decir, todo el período desde que un estudiante matricula en una determinada carrera de Ciencias Médicas, hasta su graduación. Comprende todos los resultados en la formación del personal que abarcan los principales elementos conceptuales sobre el proceso educativo. En esta etapa se construyen las bases del futuro desempeño profesional y es el espacio donde deberá confluir el aprendizaje de conocimientos y habilidades con la formación de las actitudes apropiadas para ejercer como profesional de la salud dentro de la sociedad.

En este aspecto es importante el concepto de “competencias”, entendidas no solamente como el conjunto de conocimientos y habilidades propios del desempeño técnico científico de una profesión u oficio, sino adicionalmente, de todas aquellas dimensiones humanas, cognitivas, de comunicación y de relación con el entorno que permitan que el futuro profesional se desempeñe idóneamente de cara a sus pacientes y a la sociedad.

Igualmente la formación del personal de la salud, pero especialmente de médicos y enfermeras, es preferentemente una formación en servicio y este hecho es evidente tanto para la formación de pregrado como para la de postgrado. Esto supone que el “ambiente docente” deberá incluir además de la clásica institución educativa, la intervención de nuevos escenarios o centros de formación debidamente habilitados y acreditados como tales, donde se integre verdaderamente la formación y el trabajo dentro de un clima organizativo favorable que facilite el aprendizaje del estudiante.

1.4.2 Concepto de Estadísticas de Salud.

Es la información numérica, imprescindible y cuantificable para conocer el estado de salud de la población así como para planificar, organizar, evaluar y controlar programas y acciones de salud. La información estadística de salud debe reflejar lo más fiel posible la realidad objetiva que mide. Las estadísticas de salud tienen cada vez mayor alcance y complejidad en la medida que se amplía y desarrolla el quehacer en salud. [5]

Importancia y uso de las Estadísticas de Salud.

Las estadísticas de salud tienen uso individual y estadístico. El uso individual se refiere al uso de los registros médicos de cada persona que accede a los servicios de salud donde quedan registradas ciertas características del individuo y la historia de su enfermedad, muerte, tratamientos u otros servicios recibidos. Los registros médicos deben poseer los atributos de confidencialidad y custodia lo cual se regula por leyes y reglamentaciones con amparo legal. El uso estadístico se refiere al manejo de datos agregados donde se suman los datos relativos a cada individuo en modelos que compilan la información individual o “de caso a caso” con las periodicidades establecidas para los diferentes niveles del sistema nacional de salud.

Las estadísticas de salud son imprescindibles para movilizar recursos humanos y materiales, intervienen de manera determinante en el proceso de toma de decisiones en los diferentes niveles de mando, estratégico y operativo. Se utilizan en todas las etapas de la administración o la gerencia del sector de la salud, son de utilidad para otros sectores que se ocupan de las demás actividades socioeconómicas del país. De tal manera, no resulta difícil entender el valor que en estos tiempos tiene el uso de las estadísticas de salud para desarrollar la gerencia sobre una base científica. Se utilizan en todas las etapas de la misma con especial interés en el proceso de toma de decisiones. [6]

1.4.3 Comunicación Estadística.

Se entiende por Comunicación Estadística, el reflejo o proyección del proceso y los resultados de la actividad estadística. Sin duda el cuidado con que se realice la misma favorecerá el mejor uso de los datos en sus diferentes funciones, fundamentalmente, en el proceso de toma de decisiones.

La comunicación estadística puede hacerse de diferentes formas que varían en función del contenido, la extensión y en aspectos formales propiamente dichos.

Antes de decidir la forma de la comunicación estadística se debe establecer lo siguiente: [7]

- ¿A quién va dirigida?
- ¿Qué nos proponemos comunicar o para qué la necesitan los usuarios o receptores?

- ¿Cómo recibirán los resultados?
- ¿Cuándo o cada qué tiempo se les comunicará la información?

1.5 Tendencias y tecnologías actuales para dar solución al problema.

En la actualidad con el avance de las nuevas tecnologías, las aplicaciones de gestión han ganado espacio por las facilidades que brindan. Las tecnologías y herramientas más usadas para desarrollar este tipo de aplicaciones giran alrededor de diferentes elementos como: Internet, aplicaciones Web, así como los lenguajes orientados a la arquitectura cliente servidor y los sistemas de gestión de bases de datos.

1.5.1 Página Web

Una página de Internet o página Web es un documento electrónico que contiene información específica de un tema en particular y que es almacenado en algún sistema de cómputo que se encuentre conectado a la red mundial de información denominada Internet. Este documento puede ser consultado por cualquier persona que se conecte a esta red mundial de comunicaciones y que cuente con los permisos apropiados para hacerlo. Una página Web es la unidad básica del World Wide Web.

1.5.2 Sitio Web

Es un conjunto de archivos electrónicos y páginas Web referentes a un tema en particular, que incluye una página inicial de bienvenida, generalmente denominada home page, con un nombre de dominio y dirección en Internet específicos. Empleados por las instituciones públicas y privadas, organizaciones e individuos para comunicarse con el mundo entero. [8]

1.5.3 Aplicación Web

En ingeniería de software una aplicación Web es aquella que los usuarios usan accediendo a un servidor Web a través de Internet o de una intranet. Las aplicaciones Web son populares debido a la practicidad del navegador Web como cliente ligero. La habilidad para actualizar y mantener

aplicaciones Web sin distribuir e instalar software en miles de potenciales clientes es otra razón de su popularidad.

Ventajas de las aplicaciones Web: [9]

1. Extrapolación y sindicación absoluta. El hecho de que todas las aplicaciones se realicen sobre Web, va a permitir que entre ellas se pueda compartir toda la información.
2. Ubicuidad. La Web ya se ha consagrado como el canal de interoperabilidad por excelencia. Es decir, las aplicaciones basadas en Web pueden desarrollarse en cualquier terminal (y no necesariamente en los PC): ordenadores, móviles, PDAs, TV digital. Esto permite tener la información en todo momento y desde cualquier terminal con conexión a Internet.
3. Seguridad. La capacidad de seguridad y de protección de datos de servidores de empresas profesionales será siempre mucho mayor que la mantenida en servidores compartidos o en los mismos ordenadores de gestión diaria. Pérdidas de datos por fallos del sistema, virus, ataques, son constantes en los ordenadores personales sin que se mantengan copias de seguridad adecuadas y siendo el coste de restauración muy elevado para estas empresas.

1.5.4 Arquitectura Cliente / Servidor

La arquitectura Cliente / Servidor representa la combinación de sistemas que pueden colaborar entre si para dar a los usuarios toda la información que ellos necesiten sin que tengan que saber donde esta ubicada. Este tipo de organización se basa en que entre todos los ordenadores que están en la red, unos ofrecen servicios (los llamados servidores) y otros usan esos servicios (los denominados clientes).

Las aplicaciones de gestión empresariales basadas en la arquitectura cliente/servidor permiten mayor y más rápido acceso a la información. El usuario tiene un ordenador en su mesa, al que llegan los datos que necesita y que posteriormente puede analizar en su propio ordenador con su propio criterio de análisis.

Ventajas: [10]

- **Aumento de la productividad:** Los usuarios pueden utilizar herramientas que le son familiares, como hojas de cálculo y herramientas de acceso a bases de datos. Mediante la integración de las aplicaciones cliente/servidor con las aplicaciones personales de uso habitual, los usuarios pueden construir soluciones particularizadas que se ajusten a sus necesidades cambiantes. Una interfaz gráfica de usuario consistente reduce el tiempo de aprendizaje de las aplicaciones.
- **Menores costes de operación:** Permiten un mejor aprovechamiento de los sistemas existentes, protegiendo la inversión. Por ejemplo, el compartir los servidores (habitualmente caros) y dispositivos periféricos (como impresoras) entre máquinas clientes permite un mejor rendimiento del conjunto. Proporcionan un mejor acceso a los datos. La interfaz de usuario ofrece una forma homogénea de ver el sistema, independientemente de los cambios o actualizaciones que se produzcan en él y de la ubicación de la información. El movimiento de funciones desde un ordenador central hacia servidores o clientes locales origina el desplazamiento de los costes de ese proceso hacia máquinas más pequeñas y por tanto, más baratas.
- **Mejora en el rendimiento de la red:** Las arquitecturas cliente/servidor eliminan la necesidad de mover grandes bloques de información por la red hacia los ordenadores personales o estaciones de trabajo para su proceso. Los servidores controlan los datos, procesan peticiones y después transfieren sólo los datos requeridos a la máquina cliente. Entonces, la máquina cliente presenta los datos al usuario mediante interfaces amigables. Todo esto reduce el tráfico de la red, lo que facilita que pueda soportar un mayor número de usuarios. Tanto el cliente como el servidor pueden escalarse para ajustarse a las necesidades de las aplicaciones.

1.6 Lenguajes de programación para la Web.

En la actualidad los lenguajes de programación para la Web se clasifican en dos grupos teniendo en cuenta donde se implementan respecto a la arquitectura Cliente/Servidor, nombrándose lenguajes del lado del cliente y del lado del servidor.

1.6.1 Lenguajes utilizados del lado del cliente:

HTML: HyperText Markup Language, es un lenguaje simple utilizado para crear documentos de hipertexto. La presentación de la página es muy dependiente del browser (o programa navegador) utilizado: el mismo documento no produce el mismo resultado en la pantalla si se visualiza con un browser en modo línea, Mozilla o Netscape, o sea, HTML se limita a describir la estructura y el contenido de un documento, y no el formato de la página y su apariencia. Por tanto, HTML es un lenguaje muy sencillo que permite preparar documentos Web insertando en el texto de los mismos una serie de marcas que controlan los diferentes aspectos de la presentación y comportamiento de sus elementos. [11]

Java Script: Lenguaje de programación que ha permitido el gran desarrollo de la animación de las páginas Web. En realidad es el lenguaje más fácil de aprender para un diseñador de páginas Web. Ha sido la base para lograr una animación en la Web sin que se tengan que estudiar lenguajes difíciles de programar.

El advenimiento de Java Script ha resuelto de manera fácil y elegante la mayoría de los problemas con que se enfrenta el diseñador de páginas Web referidos a la programación. En primer lugar porque sus requerimientos son relativamente sencillos, y lo que quizás sea más importante, es un lenguaje cuyos códigos se resuelven en el navegador del cliente, sin tener que ir y venir del cliente al servidor actualizando la información. [12]

PHP. AJAX: Es una técnica de desarrollo Web que usa una combinación de varias tecnologías:

- (HTML (o XHTML) y Hojas de Estilo en Cascada (CSS) para presentar la información.
- Document Object Model (DOM) y Java Script, para interactuar dinámicamente con los datos.
- XML y XSLT, para intercambiar y manipular datos de manera sincronizada con un servidor Web.

Es una de las tecnologías desarrolladas para coordinar los lenguajes del lado del cliente con tecnologías del lado del servidor, uno de los principales objetivos es añadir más funcionalidad, especialmente para crear una experiencia interactiva que no requiera recargar la página cada vez (cosa que suele molestar a los usuarios).

AJAX no constituye una tecnología en sí, sino que es un término que engloba a un grupo de estas que trabajan conjuntamente. 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]

1.6.2 Lenguajes utilizados del lado del servidor:

En este grupo encontramos entre los más utilizados: PERL, ASP, JSP, PHP. Estos lenguajes permiten desarrollar lógica del negocio dentro del servidor, y posibilitan el acceso a las bases de datos y el procesamiento de la información.

PHP: Este lenguaje tiene gran popularidad a la hora de desarrollar aplicaciones de gestión, respecto a los demás lenguajes del lado del servidor, por sus características y su facilidad de aprendizaje. Es un lenguaje interpretado de alto nivel, embebido en páginas HTML y ejecutado en el servidor. Es originario del nombre PHP Tools, o Personal Home Page Tools, sirve principalmente para proporcionar características dinámicas a una página Web. Se interpreta y ejecuta directamente en el servidor en el que está albergada la página Web, con lo que el visitante a la misma únicamente recibe el resultado buscado por el código en el que está escrito. [14]

Características: Velocidad, estabilidad, seguridad y simplicidad.

1. Velocidad: No solo la velocidad de ejecución, la cual es importante, sino además no crea demoras en la máquina. Por esta razón no debe requerir demasiados recursos de sistema.
2. Estabilidad: La velocidad no sirve de mucho si el sistema se cae cada cierta cantidad de ejecuciones. Ninguna aplicación es 100% libre de bugs, pero teniendo de respaldo una increíble comunidad de programadores y usuarios es mucho más difícil para estos sobrevivir. PHP utiliza su

propio sistema de administración de recursos y dispone de un sofisticado método de manejo de variables, conformando un sistema robusto y estable.

3. Seguridad: El sistema debe poseer protecciones contra ataques. PHP provee diferentes niveles de seguridad, estos pueden ser configurados desde el archivo .ini.

4. Simplicidad: Se les debe permitir a los programadores generar código productivamente en el menor tiempo posible. Usuarios con experiencia en C y C++ podrán utilizar PHP rápidamente. [15]

1.6.3 ¿Por qué usar PHP? [16]

1. Corre en casi cualquier plataforma utilizando el mismo código fuente. Como en todos los sistemas se utiliza el mismo código base, los scripts pueden ser ejecutados de manera independiente al sistema operativo.

2. Es completamente expandible. Está compuesto de un sistema principal, un conjunto de módulos y una variedad de extensiones de código.

3. Tiene muchas interfaces distintas para cada tipo de servidor. Actualmente se puede ejecutar bajo diferentes servidores Web por ejemplo: Apache, IIS, AOLServer, Roxen yTHTTPD.

4. Puede interactuar con muchos motores de bases de datos tales como MySQL, MS SQL, Oracle, Informix, PostgreSQL, y otros muchos.

5. Brinda una gran variedad de módulos, cuando un programador PHP necesita una interface para una librería en particular, fácilmente podrá crear una API para esta. Algunas de las que ya vienen implementadas permiten manejo de gráficos, archivos PDF, Flash, Cybercash, calendarios, XML, IMAP, POP, etc.

6. Es software libre. Se puede obtener en la Web y su código esta disponible bajo la licencia GPL.

1.7 Navegador recomendado: Mozilla Firefox.

Un Navegador o "Browser" está compuesto por una serie de estructuras programáticas que le permiten desplegar y ejecutar documentos e instrucciones. Este tipo de instrucciones pueden variar desde

documentos estáticos (HTML), contenido dinámico (JavaScript), contenido gráfico (Flash), hasta otro ambiente de navegación por separado al "Browser". [17]

Uno de los navegadores más destacados es Mozilla Firefox, es de código abierto, multiplataforma, con interfaz gráfica de usuario desarrollado por la Mozilla Corporation y cientos de voluntarios. Se ha convertido en el principal foco de desarrollo de la Fundación Mozilla, junto con el cliente de correo electrónico y el lector de noticias.

1.8 Servidor WEB: Apache.

Apache fue hecho para proveer un alto grado de calidad y fortaleza para las implementaciones que utilizan el protocolo HTTP. Está ligado a las plataformas Linux, Windows y UNIX. Es un software libre, porque sus desarrolladores defienden la teoría de que las transmisiones usando la red deben estar en las manos de todos, y que las compañías de software deben hacer el dinero ofertando servicios con valor añadido tales como módulos especializados, soportes, entre otros, y no siendo dueñas de un protocolo. Así, el proyecto de crear una implementación robusta con referencia absolutamente libre para quien lo quiera usar es un buen paso para evitar la propiedad sobre los protocolos. [18]

1.9 Sistemas Gestores de base de datos.

Los gestores de base de datos son sistema formados por un conjunto de datos y un paquete de software para la gestión del mismo, de modo que se controla el almacenamiento de datos redundantes. Los datos resultan independientes de los programas que los usan, se almacenan las relaciones entre los datos junto con éstos y se puede acceder a los datos de diversas formas. [19]

Para la construcción de aplicaciones de gestión se destacan por su eficiencia gestores como: Oracle, que es considerado uno de los más potentes, MySQL, SQL Server y PostgreSQL. Este último es considerado el Sistema de Gestión de Bases de Datos de código abierto (gratuito y con código fuente disponible) más avanzado del mundo.

1.9.1 Gestor de Base de Datos utilizado: MySQL.

Su principal objetivo de diseño fue la velocidad, por ello se suprimieron algunas características de los demás Sistema de Gestión de Base de Datos (SGBDs), como las transacciones y los subselects. Consume pocos recursos y se distribuye bajo licencia GPL (a partir de la versión 3.22).

Ventajas de MySQL: [20]

- Mayor rendimiento.
- Mejores utilidades de administración.
- Integración perfecta con PHP.
- Sin límites en los tamaños de los registros.
- Mejor control de acceso de usuarios.
- Escalabilidad: Es posible manipular bases de datos enormes, del orden de seis mil tablas y alrededor de cincuenta millones de registros, y hasta 32 índices por tabla. MySQL está escrito en C y C++ y probado con multitud de compiladores y dispone de APIs para muchas plataformas diferentes.
- Conectividad: Es decir, permite conexiones entre diferentes máquinas con distintos sistemas operativos. Es corriente que servidores Linux o Unix, usando MySQL, sirvan datos para ordenadores con Windows, Linux, Solaris, etc. Para ello se usa TCP/IP, tuberías, o sockets Unix. Es multihilo, con lo que puede beneficiarse de sistemas multiprocesadores. Permite manejar multitud de tipos para columnas. Permite manejar registros de longitud fija o variable.

1.10 Motor de plantillas utilizado: 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.

El motor de plantillas utilizado en el desarrollo fue Smarty, se encuentra bajo la licencia GPL por lo que puede ser usado libremente. [21]

Algunas de las características de Smarty:

- Es extremadamente rápido.
- Es eficiente ya que puede interpretar el trabajo más sucio.
- No analiza gramaticalmente desde arriba el template, sólo compila una vez.
- Está atento para sólo recompilar los archivos de plantilla que fueron cambiados.
- Se pueden crear funciones habituales y modificadoras de variables personalizadas, de modo que el lenguaje de la plantilla es altamente extensible.
- Los constructores if/elseif/else/endif son pasados por el interpretador de PHP, así la sintaxis de la expresión {if...} puede ser compleja o simple de la forma que usted quiera.
- Permite un anidamiento ilimitado de sections, ifs, etc.
- Es posible incrustar directamente código PHP en los archivos de plantilla.
- Funciones habituales de manipulación de cache.
- Arquitectura de Plugin.

1.11 Metodologías de desarrollo del software.

Todo desarrollo de software es riesgoso y difícil de controlar, pero si no se utiliza una metodología, lo que se obtienen son clientes insatisfechos con el resultado y desarrolladores aún más insatisfechos. Dentro de las metodologías más utilizadas e importantes se destacan: Rational Unified Process (RUP), Extreme Programming (XP) y Microsoft Solution Framework (MSF).

Lo más importante antes de elegir la metodología que se usará para todo el ciclo de desarrollo de un software, es determinar el alcance que tendrá y luego de ahí ver cuál es la que más se acomoda a la aplicación.

1.11.1 ¿Por qué usar RUP?

RUP es un marco de desarrollo de software iterativo e incremental, que está compuesto por cuatro fases denominadas Inicio, Elaboración, Construcción y Transición. Estas iteraciones ofrecen como resultado un incremento del producto desarrollado que añade o mejora las funcionalidades del sistema en desarrollo. Es un marco de trabajo extensible que puede ser adaptado a organizaciones o proyectos específicos. Está basado en componentes e interfaces bien definidas y utiliza el Lenguaje Unificado de Modelado (UML) para visualizar, especificar, construir y documentar un sistema de software.

1.11.2 Características de RUP. [22]

- Dirigido por los casos de uso: En el Proceso Unificado los casos de uso se utilizan para capturar los requisitos funcionales y para definir los contenidos de las iteraciones. La idea es que cada iteración coja un conjunto de casos de uso o escenarios y desarrolle todo el camino a través de las distintas disciplinas: diseño, implementación, prueba, etc.
- Centrado en la arquitectura: El Proceso Unificado asume que no existe un modelo único que cubra todos los aspectos del sistema. Por dicho motivo existen múltiples modelos y vistas que definen la arquitectura software de un sistema. La analogía con la construcción es clara, cuando construyes un edificio existen diversos planos que incluyen los distintos servicios del mismo: electricidad, fontanería, etc.

- Iterativo e incremental: RUP divide el proceso en cuatro fases, dentro de las cuales se realizan varias iteraciones en número variable según el proyecto y las cuales se definen según el nivel de madurez que alcanzan los productos que se van obteniendo con cada actividad ejecutada. La terminación de cada fase ocurre en el hito correspondiente a cada una, donde se evalúa que se hayan cumplido los objetivos de la fase en cuestión.

1.11.3 Características del Lenguaje de Modelado Unificado (UML).

UML es un lenguaje estándar para escribir planos de software. Puede utilizarse para visualizar, especificar, construir y documentar los artefactos de un sistema que involucra gran cantidad de software. Ofrece un estándar para describir un "plano" del sistema (modelo), incluyendo aspectos conceptuales tales como procesos de negocios y funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y componentes de software reutilizables.

Es importante resaltar que UML es un "lenguaje" para especificar y no para describir métodos o procesos, es un lenguaje de modelado orientado a objetos estándar. 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 aplica en una gran variedad de formas para dar soporte a una metodología de desarrollo de software, tal como el Proceso Unificado Racional. [23]

1.12 Herramientas utilizadas.

Una buena selección de las herramientas, ayuda a todo el equipo de desarrollo a realizar un buen trabajo en la construcción del software y a lograr que el producto salga con la mayor calidad y en el menor tiempo posible. Las herramientas usadas para el desarrollo de esta aplicación fueron:

1.12.1 Dreamweaver 8.

Dreamweaver 8 es la herramienta de desarrollo Web líder del mercado, permite diseñar, desarrollar y mantener de forma eficaz sitios y aplicaciones Web basados en normas, lo abarca todo, desde la creación y el mantenimiento de sitios Web básicos hasta aplicaciones avanzadas compatibles con las mejores prácticas y las tecnologías más recientes.

1.12.2 Rational Suite 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 y entregables.

Rose 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 de Rose es que utiliza la notación estándar en la arquitectura de software (UML), la cual 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. [24]

Rational Rose permite completar una gran parte de las disciplinas (flujos fundamentales) del proceso unificado de Rational (RUP), como Modelado del negocio, Captura de requisitos, Análisis y diseño, Implementación y Control de cambios y gestión de configuración.

1.12.3 EMS 2005 y MySQL Administrator.

Estas son herramientas para el trabajo con el sistema gestor de base de datos, que facilita toda la gestión de usuarios y configuración, además para la realización de consultas, vistas y procedimientos almacenados.

1.12.4 ZendStudio 5.2.0

Se trata de un programa de la casa Zend, impulsores de la tecnología de servidor PHP, orientada a desarrollar aplicaciones web en lenguaje PHP. El programa, además de servir de editor de texto para páginas PHP, proporciona una serie de ayudas que pasan desde la creación y gestión de proyectos hasta la depuración de código. [25]

Conclusiones

En este capítulo se realizó un estudio detallado de todas las tecnologías, metodologías de software, herramientas y técnicas usadas en el desarrollo del software así como una breve justificación de su uso. Además se abordaron conceptos importantes y algunos antecedentes que están muy relacionados con el objetivo de este trabajo y el dominio del problema. A partir de estos puntos se comenzará el desarrollo de la propuesta de sistema.

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

Introducción

En este capítulo se realiza un estudio de todos los aspectos relacionados con el dominio y el entorno del problema para caracterizar el sistema de acuerdo al objeto de estudio y los procesos a automatizar. Se hace una descripción general de la propuesta de sistema y cómo debe funcionar. Además se describe el modelado del negocio, la especificación de los requisitos funcionales y no funcionales así como la definición de los casos de uso.

2.1 Objeto de estudio.

2.1.1 Situación problemática.

Tradicionalmente realizar reportes y gestionar los resultados académicos de un estudiante a lo largo de la carrera de medicina era de forma manual. Surgieron sistemas locales como un acercamiento a la solución del problema, pero se usaban de manera ineficiente y en muchos de los casos todo se procesaba manualmente. No se podía tener un dominio pleno de toda esta información a nivel nacional, debido a que existen numerosas sedes universitarias de ciencias médicas en todo el país y se hacía muy difícil conocer la información general o específica de un centro, así como obtener detalladamente los resultados cuantitativos y cualitativos para poder ofrecer un control de los datos estadísticos de cada institución y realizar resúmenes a distintos niveles.

Debido a esta situación existente surgió la idea de desarrollar un sistema informático que facilitara el proceso: Sistema de Gestión de Información en el Proceso de Formación de Recursos Humanos en Salud. Desde sus inicios, en este sistema se concibieron todos sus módulos, de los cuales algunos ya han sido liberados y permiten gestionar toda la información que antes parecía imposible de manipular. Aunque en una segunda iteración del producto, se requiere automatizar el proceso de control y acceso a la información estadística relacionada con los procesos de gestión docente durante la formación de pregrado para poder acceder a la información de cada institución desde cualquier nivel del sistema nacional de salud.

2.1.2 Descripción de los procesos del negocio.

Proceso de confección de los informes estadísticos.

En cada Centro de Enseñanza Médica Superior el área de Estadística de la secretaría docente tiene que realizar una serie de reportes estadísticos de forma periódica, sobre la docencia y la matrícula real que tienen en sus instalaciones. Estos reportes miden una serie de parámetros con un formato estándar que resulta muy engorroso por la cantidad de información manejada. Una vez terminado el informe o reporte este se entrega al organismo superior que lo atiende, que se encarga de verificar la información hasta que esté correcta, está en dependencia del tipo de centro que sea.

Por la dinámica que tiene el proceso de formación de recursos humanos en salud, se ha convertido en un proyecto donde estudian jóvenes de diferentes países de Latinoamérica y de otros continentes. Por las características de la Revolución, se hace necesaria obtener de forma inmediata información que no tienen parámetros de búsqueda fijos. Esto se hace mucho más engorroso debido a la cantidad de documentación que hay que manejar para poder darle cumplimiento a la entrega de la información solicitada.

Clasificación de los Centros de Enseñanza Médica Superior: [26]

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 postgrado 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 postgrado, 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.

Filial de Ciencias Médicas:

Es un centro universitario que puede tener una o varias carreras de Ciencias Médicas que está subordinada a una Facultad de Ciencias Médicas.

Sede Universitaria:

Es un centro universitario de ciencias médicas enclavado en un Municipio, que utiliza para el desarrollo del proceso docente educativo las unidades asistenciales existentes en las comunidades del mismo.

Flujo de información

El siguiente diagrama muestra cómo se realiza el proceso de entrega de información a los distintos niveles del Sistema Nacional de Salud, para entender el funcionamiento del Área de Estadística en las Secretarías de los Centros de Enseñanza Médica Superior de la salud identificando en cada caso quién se subordina a quién y cómo fluye la información estadística a cada uno de los niveles que la solicitan.



Fig. 2.1 Flujo de Información Estadística.

2.1.3 Sistemas automatizados vinculados al módulo de salidas.

Para manipular y obtener alguna información almacenada en los centros de enseñanza médica de salud antes mencionados a través del sistema es necesario hacer uso varios servicios Web. Uno de ellos es el Registro de Ubicación, que brinda toda la información referente a la distribución de provincias y municipios del país. Además, el Registro de Unidades de Salud posibilita obtener la información de todos los Centros de Enseñanza Médica Superior que se encuentran en cada territorio.

2.2 Documentos que se procesan.

Los documentos que se procesan para el funcionamiento del sistema son los Modelos Oficiales Estadísticos del MINSAP para los centros de enseñanza médica superior, entre ellos están:

Modelo 1163-01: Fuentes de ingreso en la educación superior para cursos diurnos.

Modelo 1163-01: Fuentes de ingreso en la educación superior para programas especiales.

- Modelo 1167-02: Informe de graduados de la educación superior para cursos diurnos.
- Modelo 1167-02: Informe de graduados de la educación superior para programas especiales.
- Modelo 223.001: Matrícula por provincia o municipio de residencia, carrera y año de estudio.
- Modelo 223.005: Composición de la matrícula inicial por color de la piel, sexo y por tipo de curso.
- Modelo 223.008: Matrícula y graduados de estudiantes extranjeros por países.

2.3 Propuesta del sistema.

El módulo de Salidas es capaz de generar listados, resúmenes y modelos oficiales estadísticos de toda la información que se almacena en los CEMS relacionada con la formación de los recursos humanos durante pregrado. En este caso salida se refiere a los resultados e informaciones generadas por el Sistema. Este módulo se integrará al Sistema de Gestión de Información en el Proceso de Formación de Recursos Humanos en Salud y los usuarios sólo tendrán la posibilidad de visualizar la información en la aplicación y exportarla en formato PDF.

El sistema que se propone se encontrará brindando servicio sobre la red privada de salud cubana INFOMED, a través de ella todos los nodos (secretarías docentes de todos los centros de enseñanza médica superior y los puestos de trabajos relacionados con el sistema) accederán a este, desde cualquier lugar del país.

Existirán diferentes privilegios para el acceso a la información estadística almacenada en dependencia de las características del usuario de acuerdo a su cargo y a su nivel en el Sistema Nacional de Salud. Podrán acceder a la aplicación los estadísticos de todas las secretarías del país sólo con la posibilidad de obtener la información de su centro, y además los encargados de esta función a nivel municipal, provincial y nacional, cada uno de ellos con acceso limitado a los niveles inferiores.

Para el sistema se propone en una futura versión separar las bases de datos por especialidades, para lograr un mayor rendimiento de las mismas, esto se haría para mejorar la velocidad del trabajo en las secretarías docentes y para no provocar el almacenamiento de grandes volúmenes de datos de estudiantes ingresados en la enseñanza universitaria de salud en un mismo lugar.

2.4 Análisis comparativo de otras soluciones con la propuesta.

Después de haber realizado un estudio previo de algunas de las soluciones que actualmente se encuentran funcionando, se concluye que las soluciones que han surgido para facilitar el proceso de gestión de información estadística en las secretarías docentes del país, hoy no son capaces de brindar información rápida y precisa a todos los niveles del SNS. Por este motivo es necesario implementar un Módulo de Salidas centralizado, para garantizar el control y acceso a la información estadística relacionada con la formación de los recursos humanos de salud en el país de una manera más eficiente.

Con este objetivo la solución propuesta cuenta con tecnologías actuales desarrolladas sobre la web, lo que posibilita a los diferentes niveles de dirección, el control y consulta de la información registrada en cualquier centro clínico docente del país. Posibilita la obtención de los indicadores estadísticos establecidos en tiempo real, que en la actualidad los sistemas existentes no brindan debido a que son soluciones locales que carecen de una integración a nivel nacional. Además de que los sistemas de este tipo no contribuirían a la gran plataforma informatizada de salud que se propone desarrollar en el país.

El Módulo de Salidas se integrará al Sistema de Gestión de Información en el Proceso de Formación de Recursos Humanos en Salud, que forma parte de la plataforma informatizada que se desarrolla en el Ministerio de Salud Pública cubano, utilizando arquitecturas orientadas a servicios contando con tecnologías novedosas.

2.5 Modelo de negocio.

El proceso de modelado de negocio permite obtener una visión de la organización para definir los procesos, roles y responsabilidades de esta en los modelos de casos de uso del negocio y de objetos. El modelado del negocio brinda una vía natural para determinar los requerimientos del sistema de información, de aquí que estos procesos estén estrechamente relacionados, así como con el análisis y diseño.

Actores del Negocio.

Actor	Descripción
Solicitante	Es el que solicita información en las secretarías, ya sean listados, resúmenes o modelos oficiales estadísticos de los datos de docencia de pregrado. Si es una secretaría general puede ser un Secretario General, la Dirección del Centro Docente correspondiente, así como las dependencias del Nivel Provincial y Central del MINSAP, que soliciten. Si es una secretaría docente puede ser el Secretario Docente, así como las dependencias correspondientes.

Tabla 2.1: Actores del negocio.

Trabajadores del Negocio.

Trabajador	Descripción
Estadístico o Gestor Económico	Encargado de registrar, analizar, procesar, evaluar y controlar la información que emana de los procesos docentes que se producen en los centros de enseñanza médica superior. Puede ser a nivel Nacional, Provincial, Municipal, de Centro o Facultad.

Tabla 2.2: Trabajadores del negocio.

Diagrama de Casos de Uso del Negocio.

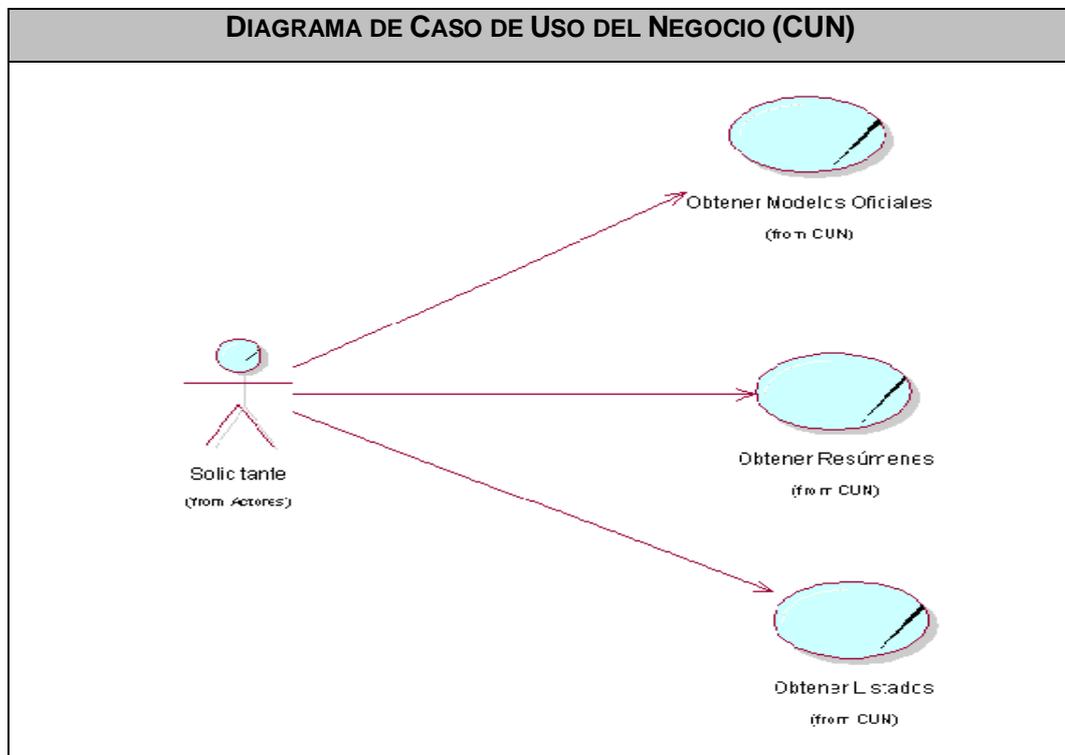


Fig. 2.2 Diagrama de Caso de Uso del Negocio.

La descripción correspondiente a los casos de uso del negocio puede verse en el **Anexo #1**.

2.5.1 Objetivos del Modelo de negocio.

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

2.6 Especificación de los requisitos de software.

2.6.1 Requisitos Funcionales.

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

Listados

El usuario debe indicar los siguientes criterios de búsqueda de acuerdo a su nivel de acceso: provincia, municipio, facultad, o grupo en la carrera, luego debe indicar los atributos que desea mostrar para conformar el listado de los estudiantes.

R1. Generar listado de estudiantes configurado por el usuario.

- 1.1. Mostrar listado de estudiantes con las características indicadas según los criterios seleccionados.
- 1.2. Imprimir el listado en el formato PDF.

Resúmenes

El usuario debe indicar los siguientes criterios de búsqueda de acuerdo a su nivel de acceso: provincia, municipio, facultad, o grupo en la carrera para obtener la ubicación y posteriormente seleccionar el tipo de resumen que desea.

R2. Generar resumen de estudiantes por país, año académico y ubicación en Cuba.

- 2.1 Mostrar total de estudiantes por países en cada año de la carrera.
- 2.2 Imprimir el resumen en el formato PDF.

R3. Generar resumen de estudiantes por vía de ingreso y país, provincia y centro de estudios.

- 3.1 .Mostrar total de estudiantes de cada vía de ingreso por país.
- 3.2 Imprimir el resumen en el formato PDF.

R4. Generar resumen de estudiantes por sus nacionalidades, provincia y centro de estudios.

4.1 .Mostrar total de estudiantes de cada nación.

4.2 Imprimir el resumen en el formato PDF.

R5. Generar resumen de estudiantes por edad, sexo y provincia.

5.1 Mostrar total de estudiantes por sexo y grupos etarios (menores de 17, 17, 18, 19, 20, 21-25, 26-30, 31-35, mayores de 35).

5.2 Imprimir el resumen en el formato PDF.

R6. Generar resumen de estudiantes de cada centro por provincia, municipio y centro de estudios.

6.1 Mostrar total de estudiantes por centro de estudios.

6.2 Imprimir el resumen en el formato PDF.

R7. Generar resumen de estudiantes por fecha de matrícula, provincia y centro de estudios.

7.1 Mostrar total de estudiantes por mes de matrícula y centro de estudios.

7.2 Imprimir el resumen en el formato PDF.

R8. Generar resumen de estudiantes por estado civil, provincia y centro de estudios.

8.1 Mostrar total de estudiantes por estado civil y centro de estudios.

8.2 Imprimir el resumen en el formato PDF.

R9. Generar resumen de estudiantes por color de la piel, provincia y centro de estudios.

9.1 Mostrar total de estudiantes por color de piel y centro de estudios.

9.2 Imprimir el resumen en el formato PDF.

R10. Generar resumen de estudiantes por la cantidad de hijos.

10.1 Mostrar total de estudiantes que poseen hijos por cantidad de hijos y centro de estudios.

10.2 Imprimir el resumen en el formato PDF.

R11. Generar resumen de estudiantes por etnias, provincia y centro de estudios.

11.1 Mostrar total de estudiantes por etnia y centro de estudios.

11.2 Imprimir el resumen en el formato PDF.

R12. Generar resumen de estudiantes indígenas por provincia y centro de estudios.

12.1 Mostrar total de estudiantes indígenas de cada centro.

12.2 Imprimir el resumen en el formato PDF.

R13. Generar resumen de estudiantes creyentes y no creyentes, por provincia y centro de estudios.

13.1 Mostrar total de estudiantes creyentes y no creyentes por centro de estudio.

13.2 Imprimir el resumen en el formato PDF.

R14. Generar resumen de estudiantes por religión, provincia y centro de estudios.

14.1 Mostrar total de estudiantes por religión.

14.2 Imprimir el resumen en el formato PDF.

R15. Generar resumen de estudiantes por procedencia social, país, provincia y centro de estudios.

15.1 Mostrar total de estudiantes por países y procedencia social.

15.2 Imprimir el resumen en el formato PDF.

R16. Generar un nuevo resumen configurable, donde el usuario selecciona el atributo a mostrar en la fila y en la columna.

En el caso de este resumen además de la ubicación el usuario debe seleccionar los tributos deseados.

16.1 El sistema debe mostrar un resumen en forma de tabla de acuerdo a los atributos seleccionados por el usuario por las filas y columnas.

16.2 Imprimir el resumen en el formato PDF.

Modelos

El usuario debe indicar los siguientes criterios de búsqueda de acuerdo a su nivel de acceso: provincia, municipio, facultad, o grupo en la carrera para obtener la ubicación y posteriormente seleccionar el modelo.

R17. Generar modelo 1163-01: Fuentes de ingreso en la educación superior para curso diurno.

17.1 Mostrar en formato PDF la matrícula y el nuevo ingreso en los centros de educación superior por fuente de ingreso, sexo y carreras.

R18. Generar modelo 1163-01: Fuentes de ingreso en la educación superior para programas especiales.

18.1 Mostrar en formato PDF la matrícula y el nuevo ingreso en los centros de educación superior por fuente de ingreso, sexo y carreras.

R19. Generar modelo 1167-02: Informe de graduados de la educación superior para curso diurno.

19.1 Mostrar en formato PDF los graduados, según provincia de procedencia, reportándose por ramas de la ciencia, carreras y tipo de curso.

R20. Generar modelo 1167-02: Informe de graduados de la educación superior para programas especiales.

20.1 Mostrar en formato PDF los graduados, según provincia de procedencia, reportándose por ramas de la ciencia, carreras y tipo de curso.

R21. Generar modelo 223.001: Matrícula por provincia o municipio de residencia carrera y año de estudio para curso diurno.

21.1 Mostrar en formato PDF la matrícula por carreras y año de estudio y su distribución por provincia o municipio de residencia, en el centro de enseñanza superior.

R22. Generar modelo 223.005: Composición de la matrícula inicial por color de la piel, sexo y por tipo de curso.

22.1 Mostrar en formato PDF la composición de la matrícula inicial por color de la piel, sexo y tipo de curso del Centro de Educación Superior.

R23. Generar modelo 223.008: Matrícula y graduados de estudiantes extranjeros por países.

23.1 Mostrar en formato PDF la matrícula y graduados por tipo de curso de los estudiantes extranjeros según su país de residencia.

2.6.2 Requisitos No Funcionales.

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

Usabilidad.

El sistema podrá ser utilizado por cualquier persona, incluso con mínimos conocimientos en el manejo de la computadora y el ambiente Web en sentido general, debido a que contará con una ayuda a fin de documentar al usuario en su utilización, pero lo deseable sería que el usuario tenga conocimientos básicos de informática o recibiera un período de adiestramiento.

Rendimiento.

El sistema deberá ser rápido ante las solicitudes de los usuarios y en el procesamiento de la información para una mayor rapidez de trabajo en las secretarías docentes del país.

Requerimientos de ayuda y documentación.

Se documentará la aplicación con una ayuda dirigida a los diferentes tipos de usuarios para garantizar el soporte del sistema. Se realizará mantenimiento al sistema a fin de aumentar las funcionalidades del mismo a través de versiones posteriores.

Apariencia o Interfaz externa.

El sistema debe tener una interfaz sencilla, agradable, legible y de fácil uso para el usuario. El contenido será mostrado de manera comprensible.

Soporte.

Se requiere un servidor de bases de datos con las siguientes características:

Soporte para grandes volúmenes de datos y velocidad de procesamiento.

Tiempo de respuesta rápido.

Por parte del cliente se requiere un navegador capaz de interpretar código JavaScript.

Seguridad.

Teniendo en cuenta que la aplicación almacenará información sensible referente a todo el proceso de formación de estudiantes, de la que dependerá la toma de decisiones del Estado:

1. La información estará protegida contra accesos no autorizados utilizando mecanismos de autenticación y autorización que puedan garantizar el cumplimiento de esto: usuario, contraseña y nivel de acceso dado por la asignación de roles, de manera que cada uno tenga datos de acceso propios.
2. El mecanismo para crear los usuarios que tendrán acceso al sistema es el componente de seguridad Sistema de Autenticación, Autorización y Auditoría (SAAA), que define dos roles, el de visualizador y editor, pero en este caso los dos tipos de usuario tendrán sólo los privilegios de visualizar la información.
3. Se harán validaciones de la información tanto en el cliente como en el servidor.

No obstante los usuarios accederán de manera rápida y operativa al sistema sin que los requerimientos de seguridad se conviertan en un retardo para ellos.

Fiabilidad.

El sistema estará disponible durante la jornada laboral, generalmente se le dará mantenimiento en el tiempo extra para no afectar los procesos diarios realizados en las secretarías docentes.

Las respuestas del sistema serán precisas de acuerdo a los datos almacenados.

Eficiencia.

La eficiencia de la aplicación estará determinada en gran medida por el aprovechamiento de los recursos que se disponen en el modelo Cliente/Servidor, y la velocidad de las consultas a la base de datos. Se realizará la validación de los datos en el cliente y en el servidor aquellas que por cuestiones de seguridad, o de acceso a los datos lo requieran. Lográndose así un tiempo de respuesta más rápido, una mayor velocidad de procesamiento, y un mayor aprovechamiento de los recursos.

Requisitos para la documentación de usuarios en línea y ayuda del sistema.

Se documentará la aplicación con una ayuda dirigida a los diferentes tipos de usuarios para garantizar el soporte del sistema. Se realizará mantenimiento al sistema a fin de aumentar las funcionalidades del mismo a través de versiones posteriores.

Interfaces de usuario.

El sistema debe tener una interfaz sencilla, agradable, legible y de fácil uso para el usuario. El contenido será mostrado de manera comprensible y fácil de leer.

Interfaces de Comunicación.

El sistema se comunicara con el sistema Registro de Ubicación y Registro de Unidades de Salud, como miembro de los servicios que ofrece el MINSAP y se encontrará conectado a la red de Infomed.

Requerimientos de Software.

MySQL versión 5.0.18 o cualquier otra que soporte trabajo con procedimientos almacenados, vistas.

Servidor Apache versión 2.0.55 o compatible.

PHP versión 5 que soporte el trabajo orientado a objetos.

Requerimientos de Hardware.

Server de rack de 19 con:

NIC de 1 Gbit ethernet

Array scsi de 3 Hard Disk > 70 Gb.

1 Gbyte RAM.

Dual processor.

Fuente redundante.

Certificación del fabricante para correr Linux (Debian Sarge).

2.7 Definición de los Casos de Uso.

El modelo de casos de uso describe la funcionalidad propuesta del nuevo sistema. Los diagramas de casos de uso sirven para especificar la comunicación y el comportamiento de un sistema mediante su

interacción con los usuarios y/u otros sistemas. O lo que es igual, un diagrama que muestra la relación entre los actores y los casos de uso en un sistema. Se utilizan para ilustrar los requerimientos del sistema al mostrar los eventos que se producen en el mismo. Un caso de uso representa una unidad discreta de interacción entre un usuario (humano o máquina) y el sistema. Un Caso de Uso es una unidad simple de trabajo significativo, escenarios que indican cómo debería interactuar el sistema con el usuario o con otro sistema para conseguir un objetivo específico.

Cada caso de uso tiene una descripción que describe la funcionalidad que se construirá en el sistema propuesto. Un caso de uso puede "incluir" la funcionalidad de otro caso de uso o "extender" a otro caso de uso con su propio comportamiento.

Actores del Sistema

Actor	Descripción
Estadístico	Accede a los informes que brinda el sistema Listados, Resúmenes y a los modelos oficiales.
Registro de Ubicación	Es un servicio web que nos brinda todas las localidades, por provincia y municipios de nuestro país.
Registro de Unidades de Salud	Es un servicio web que nos brinda las unidades de salud al que se accede para obtener la información de las mismas y de los centros docentes.

Tabla 2.3: Actores del sistema.

Casos de Uso del Sistema.

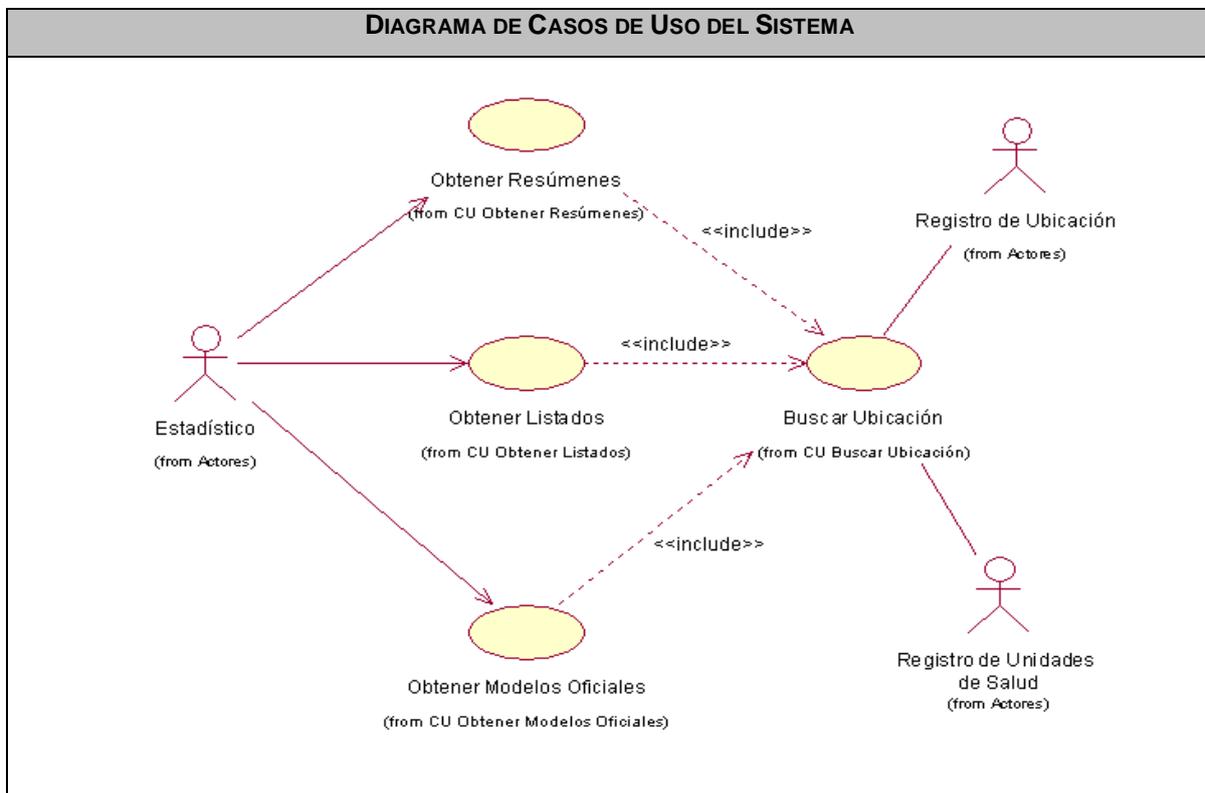


Fig. 2.3 Diagrama de Casos de Uso del Sistema.

La descripción de casos de uso del sistema en cuestión se muestra en el [Anexo #2](#).

Conclusiones.

En este capítulo se abordaron las principales características del módulo de salidas y se comenzó a profundizar en la propuesta de solución, obteniéndose las funcionalidades que va a tener el sistema, las cuales fueron representadas mediante el diagrama de casos de usos. Además se describieron todas las acciones que realizan los actores y las respuestas del sistema. Partiendo de los requisitos presentados en este capítulo se modelará el sistema.

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

Introducción

El objetivo del análisis y diseño es traducir los requisitos a una especificación que describe cómo implementar el sistema. Es necesario definir una arquitectura candidata, creando un esquema inicial de la arquitectura del sistema, identificando clases de análisis y haciendo las realizaciones de los casos de uso con las interacciones de las clases de análisis. Luego se va refinando esta arquitectura hasta llegar a su forma definitiva. El resultado final más importante del análisis y diseño será el modelo de diseño. Este consiste en colaboraciones de clases, que pueden ser agregadas en paquetes y subsistemas.

3.1 Análisis

El análisis consiste en obtener una visión del sistema que se preocupa de ver qué hace, de modo que sólo se interesa por los requisitos funcionales. Cuando se realiza un buen análisis se garantiza un trabajo de éxito. Este análisis constituirá una pieza fundamental para el desarrollo de software, ya que es la base de todo el trabajo posterior, como el diseño y la implementación.

En esta fase se realizan varias actividades como identificar las clases del análisis, así como las realizaciones de los casos de uso con las interacciones de dichas clases. Se reflejan en los diagramas de clases y de interacción, las relaciones entre las clases interfaz, controladora y entidad.

3.1.1 Diagramas de clases del análisis.

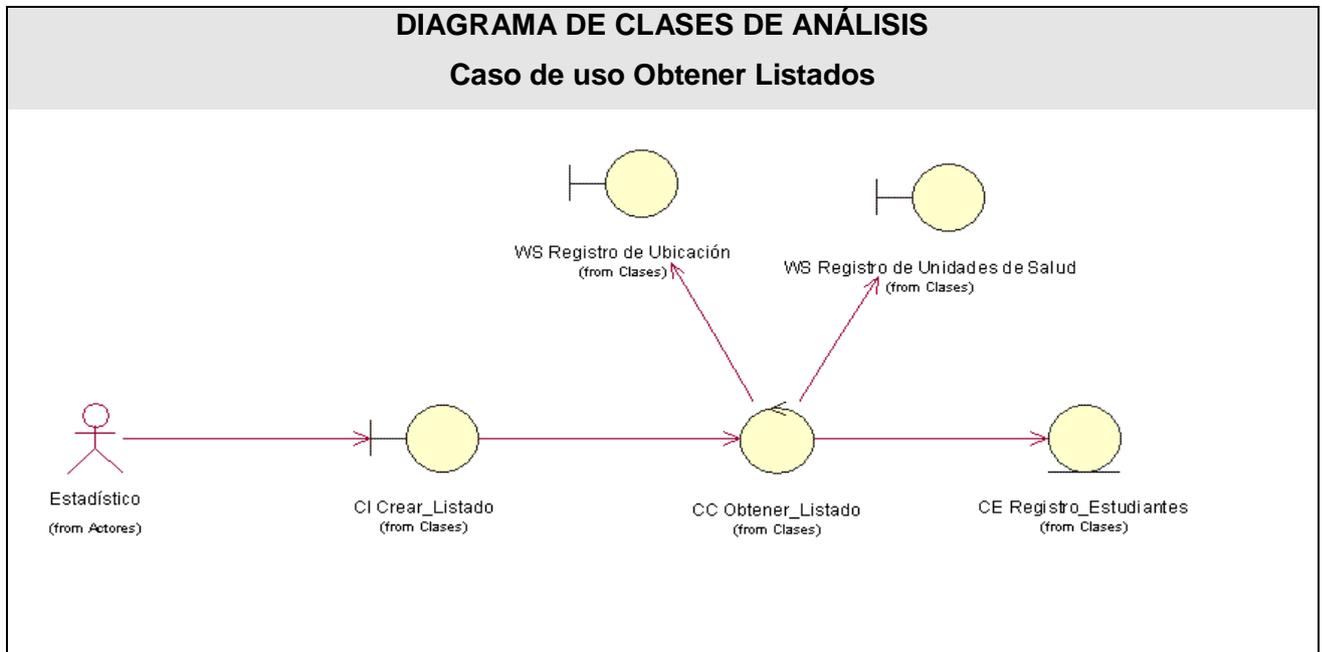


Fig. 3.2 Diagrama de clases de análisis CU Obtener Listados.

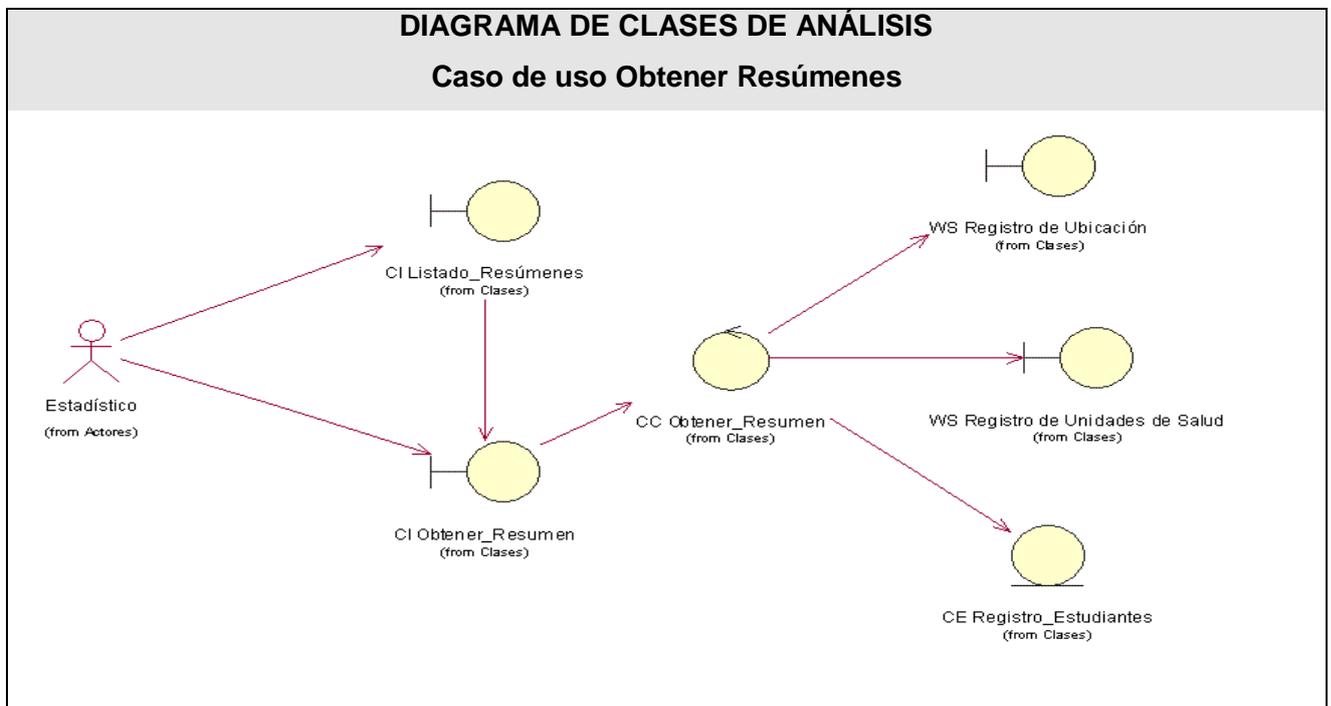


Fig. 3.3 Diagrama de clases de análisis CU Obtener Resúmenes.

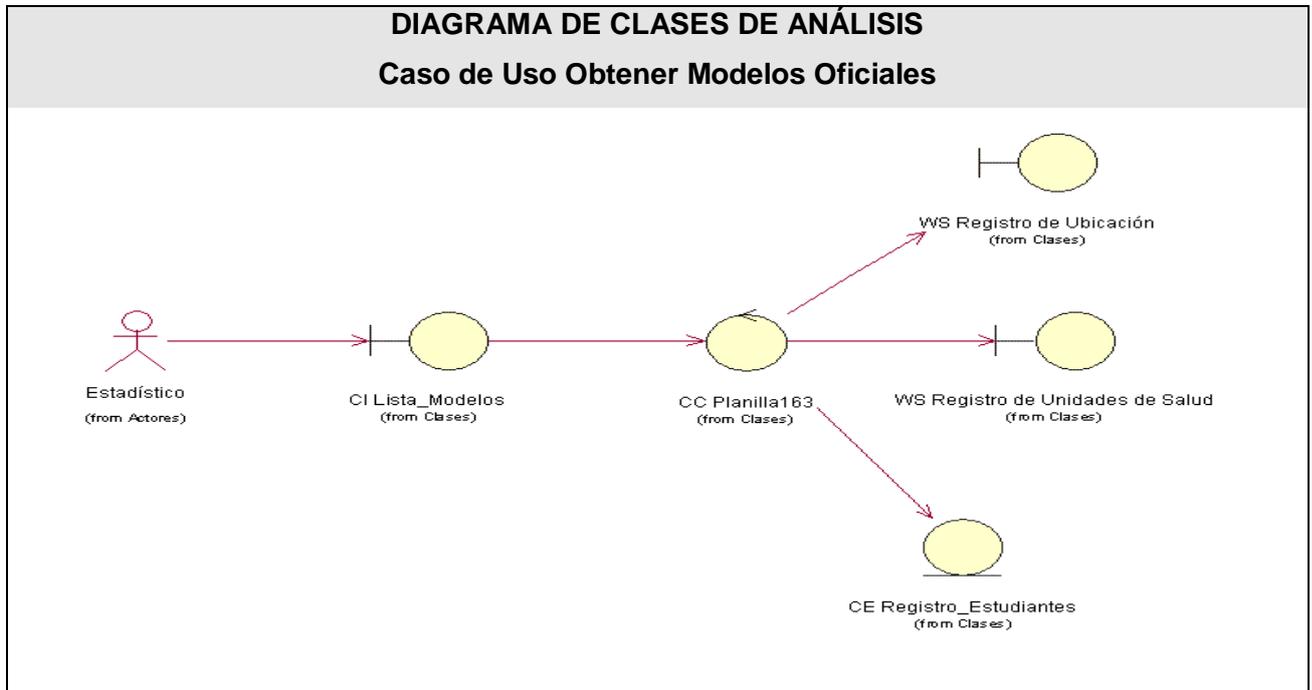


Fig. 3.4 Diagrama de clases de análisis CU Obtener Modelos Oficiales.

3.1.2 Diagramas de interacción

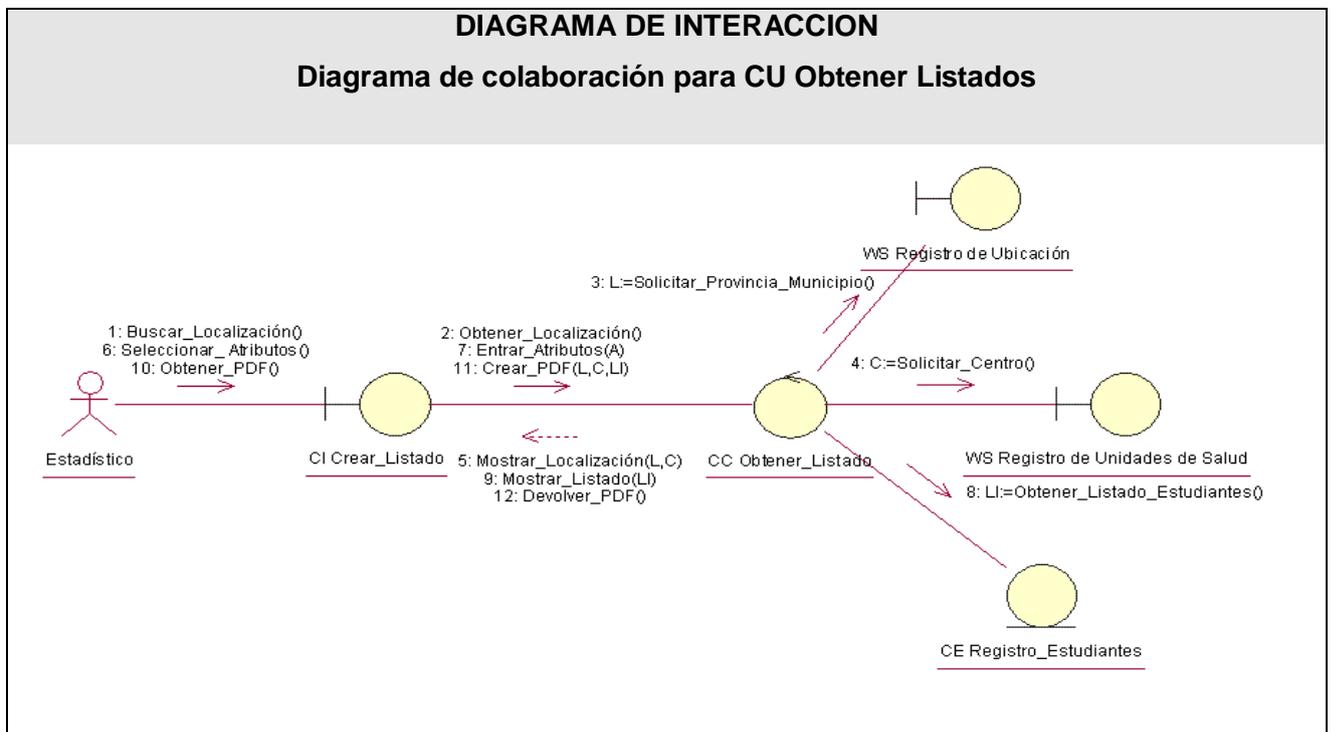


Fig. 3.5 Diagrama de interacción CU Obtener Listados.

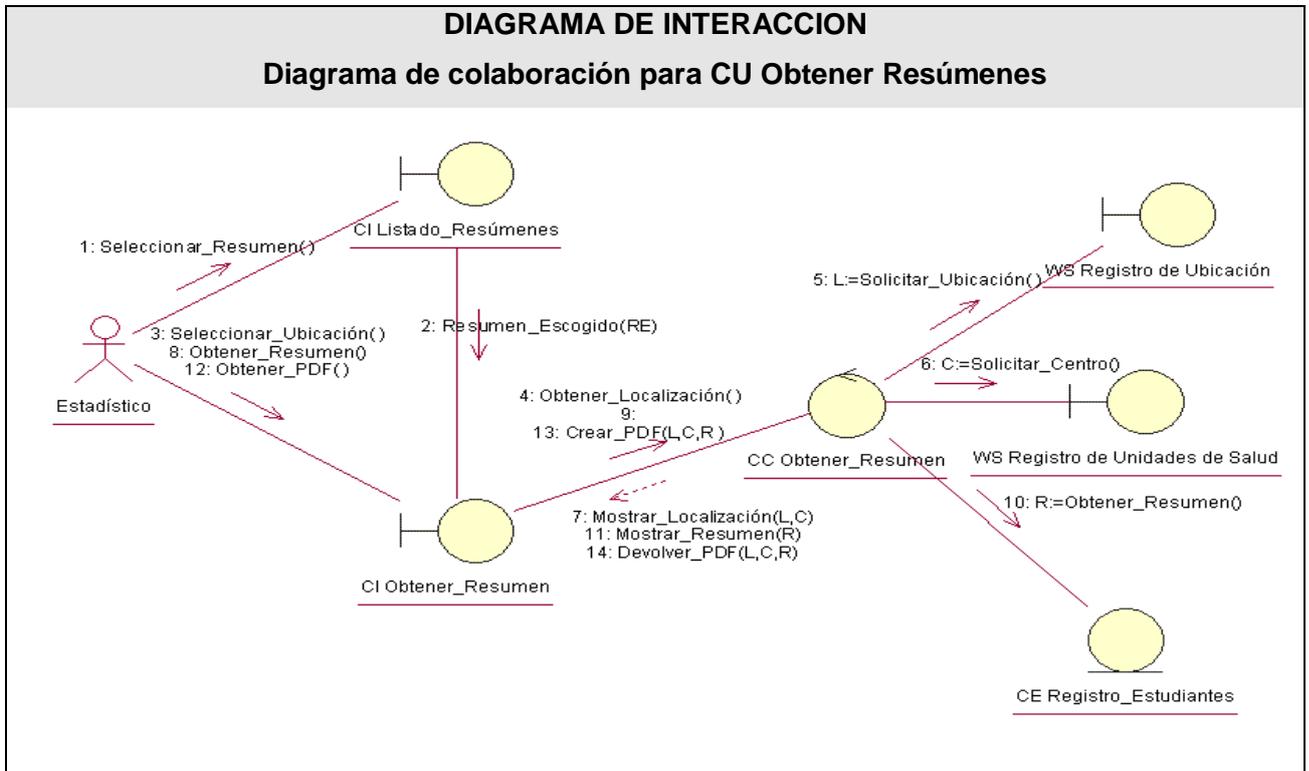


Fig. 3.6 Diagrama de interacción CU Obtener Resúmenes.

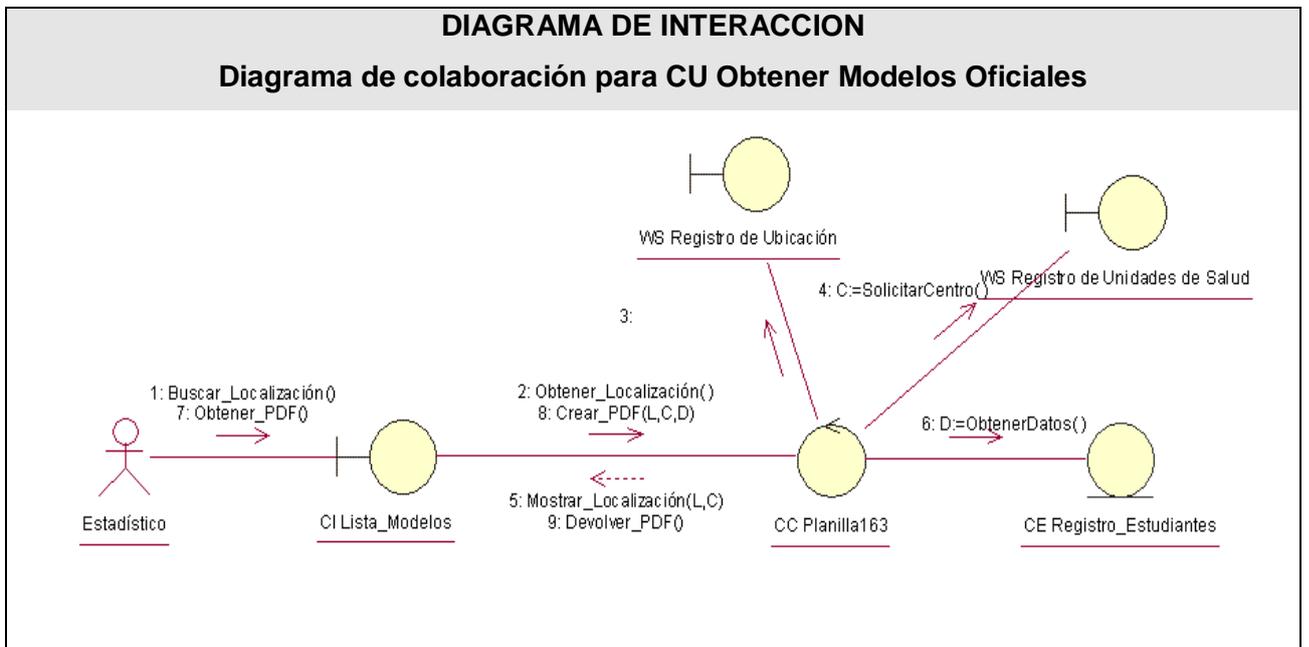


Fig. 3.7 Diagrama de interacción CU Obtener Modelos Oficiales.

3.2 Diseño

El diseño es un refinamiento del análisis que tiene en cuenta los requisitos no funcionales, en definitiva cómo cumple el sistema sus objetivos. El diseño debe ser suficiente para que el sistema pueda ser implementado sin ambigüedades. Es el proceso de aplicar distintas técnicas y principios con el propósito de definir un producto con los suficientes detalles como para permitir su realización física. Es la primera etapa técnica del proceso de Ingeniería del Software, que consiste en producir un modelo o representación técnica del software que se va a desarrollar.

- El diseño es el proceso sobre el que se asienta la calidad del software.
- El diseño de software es un proceso iterativo a través del cual se traducen los requisitos en una representación del software.
- El diseño se representa a un alto nivel de abstracción, un nivel que se puede seguir hasta requisitos específicos de datos, funcionales y de comportamiento.

Como es una aplicación Web, en esta fase se realizarán los diagramas de clases del diseño utilizando los estereotipos definidos para ello, así como los diagramas de interacción de las clases, estos últimos se podrán apreciar en el modelo de diseño [Anexo #3](#).

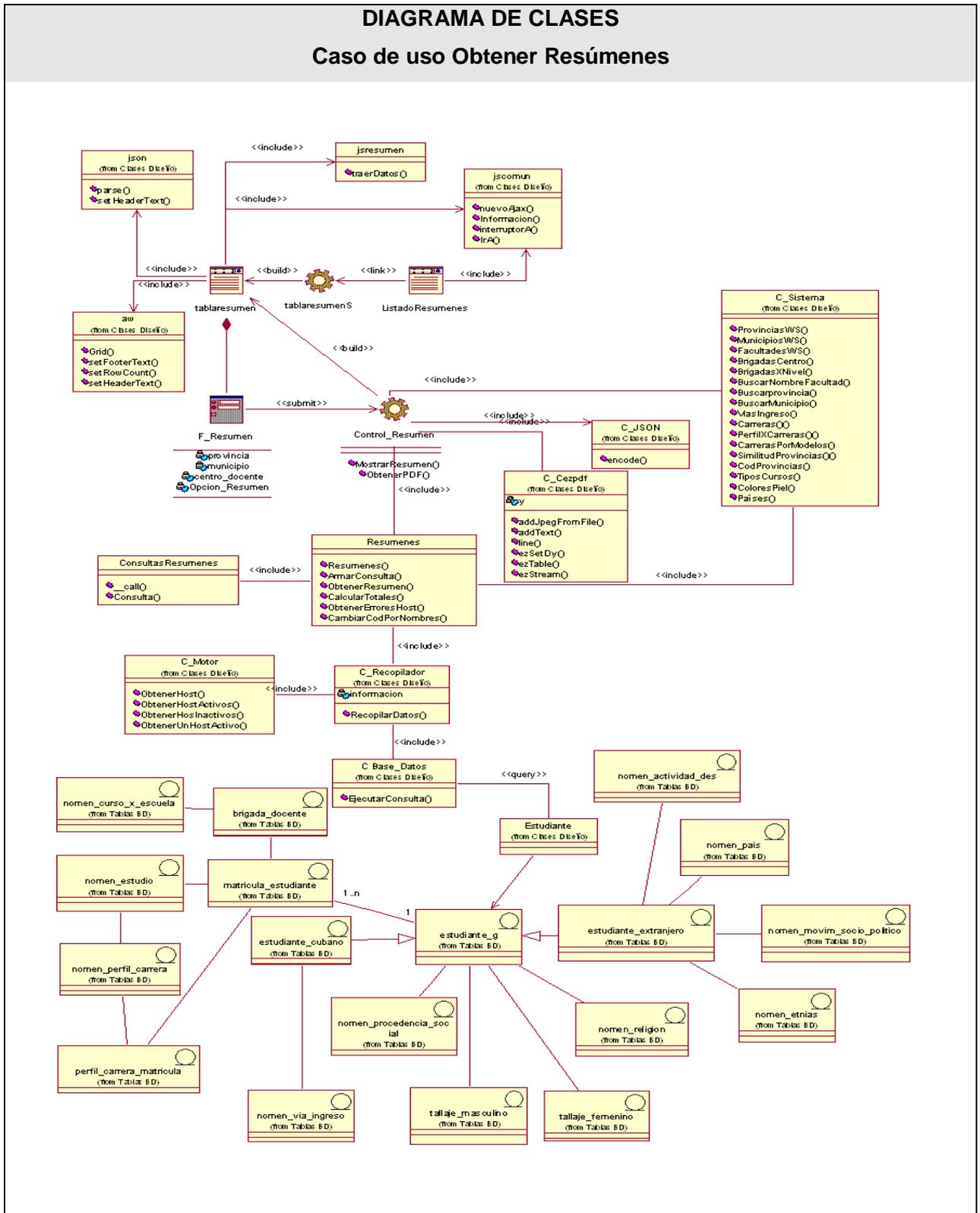


Fig. 3.9 Diagrama de clases CU Obtener Resúmenes.

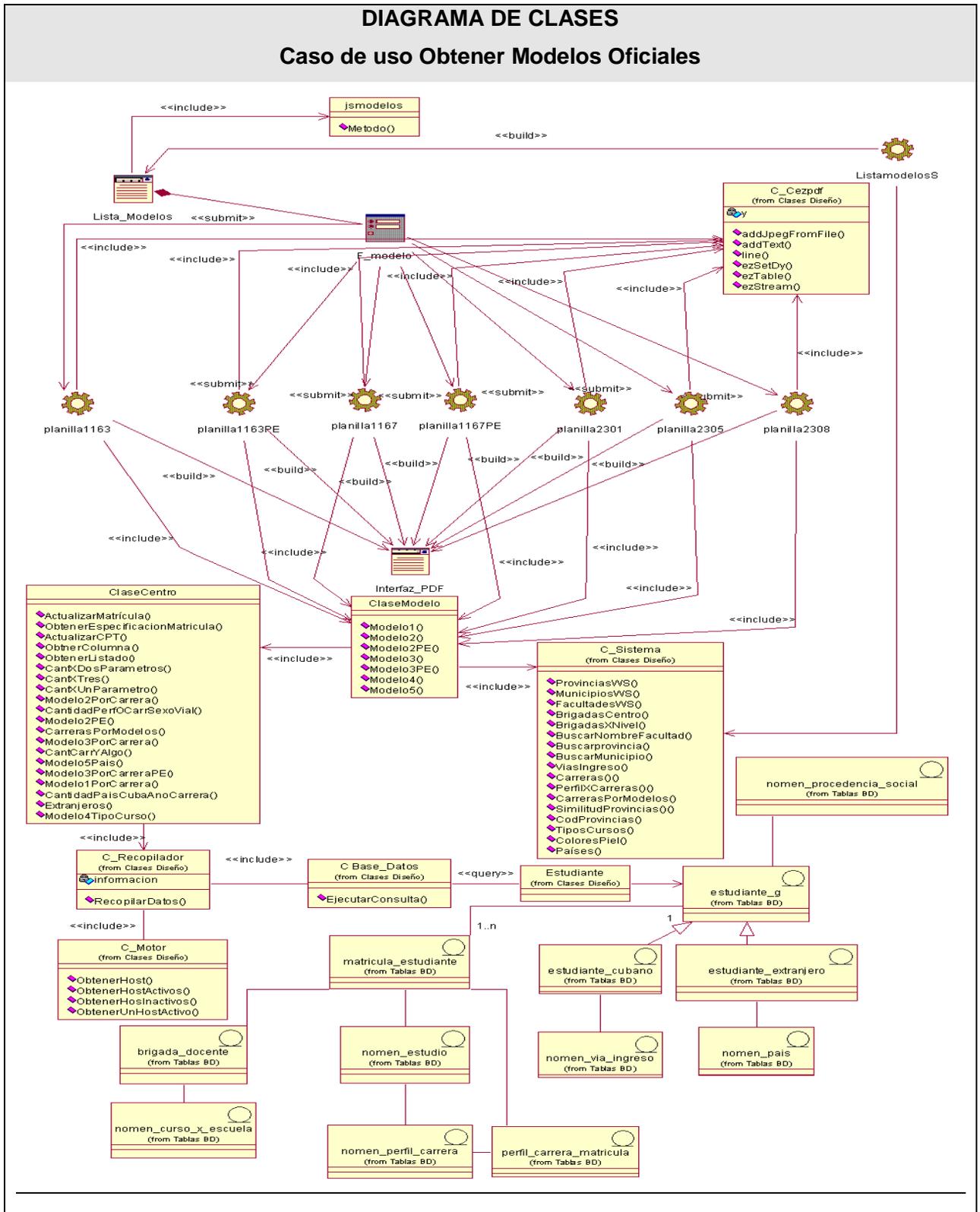


Fig. 3.10 Diagrama de clases CU Obtener Modelos Oficiales.

3.2.2 Descripción de las clases.

Leyenda definida para especificar en que caso de uso participa cada clase.

Caso de uso Obtener Listados (CU - L)

Caso de uso Obtener Resúmenes (CU - R)

Caso de uso Obtener Modelos Oficiales (CU - M)

Nombre: Estudiante_g (CU – L, R, M)	
Tipo de clase: Entidad.	
Atributo	Tipo
Id	String
Nombre	String
Segundo_nombre	String
Primer_apellido	String
Segundo_apellido	String
Fecha_nacimiento	String
Sexo	String
Color_piel	String
Es_creyente	Boolean
ID_religion	Integer
Estado_civil	String
No_hijos	Integer
Calle_residencia	String
No_residencia	String
Entre_calle	String
Ciudad_o_pueblo	String
Telefono	String
ID_consejo_popular	Integer
Fecha_matricula	String
Responsabilidad:	
Nombre:	
Descripción:	

Tabla 3.1: Clase Estudiante General.

Nombre: Estudiante_Cubano (CU – L, R, M)	
Tipo de clase: Entidad.	
Atributo	Tipo
Id	String
Condiciones_vivienda	String
Solvencia_economica	String
Índice_academico	String
Valor_escalafon	String
Zona	String
Concepto_ingreso	String
Situacion_defensa	String
Centro_procedencia	String
Responsabilidad:	
Nombre:	
Descripción:	

Tabla 3.2: Clase Estudiante Cubano.

Nombre: Estudiante_Extranjero (CU – L, R, M)	
Tipo de clase: Entidad.	
Atributo	Tipo
Id	String
Observaciones	String
Tipo_ingreso	String
ID_ modalidad_residencia	Integer
No_pasaporte	String
Fecha_vencimiento_ pasaporte	Date
Es_indigena	Bolean

Correo_electronico	String
No_cedula	String
ID_pais	Integer
ID_nacionalidad	Integer
ID_etnia	Integer
ID_pais	Integer
ID_nacionalidad	Integer
Responsabilidad:	
Nombre:	
Descripción:	

Tabla 3.3: Clase Estudiante Extranjero.

Nombre: Matricula (CU – L, R, M)	
Tipo de clase: Entidad.	
Atributo	Tipo
ID	Integer
ID_Estudio	Integer
Fecha_matricula	Date
Año_academico	Date
ID_Brigada	String
Tipo_curso	Date
Responsabilidad:	
Nombre:	
Descripción:	

Tabla 3.4: Clase Matrícula.

Nombre: Estudio (Nomenclador) (CU, L, R, M)	
Tipo de clase: Entidad.	
Atributo	Tipo

Nombre	String
Descripción	String
Duración	String
Responsabilidad:	
Nombre:	
Descripción:	

Tabla 3.5: Clase Nomenclador Estudio.

Nombre: Perfil_carrera (Nomenclador) (CU – L, R, M)	
Tipo de clase: Entidad	
Atributo	Tipo
Nombre	String
ID_cod_estudios	Integer
Responsabilidad:	
Nombre:	
Descripción:	

Tabla 3.6: Clase Nomenclador Perfil Carrera.

Nombre: Curso docente (Nomenclador) (CU – L, R, M)	
Tipo de clase: Entidad.	
Atributo	Tipo
Nombre curso	String
ID	Integer
Responsabilidad:	
Nombre:	
Descripción:	

Tabla 3.7: Clase Nomenclador Curso Docente.

Nombre: Actividad que desarrollaba antes de la carrera. (Nomenclador) (CU – L, R)	
Tipo de clase: Entidad.	
Atributo	Tipo
Nombre	String
ID	Integer
Responsabilidad:	
Nombre:	
Descripción:	

Tabla 3.8: Clase Nomenclador Actividad que desarrollaba antes de la carrera.

Nombre: Religión. (Nomenclador) (CU – L, R)	
Tipo de clase: Entidad.	
Atributo	Tipo
ID	Integer
Nombre institucion	String
Nombre ministerio	String
Nombre corriente religiosa	String
Nombre movimiento	String
Responsabilidad:	
Nombre:	
Descripción:	

Tabla 3.9: Clase Nomenclador Religión.

Nombre: Etnias (Nomenclador) (CU – L, R)	
Tipo de clase: Entidad.	
Atributo	Tipo
Nombre	String
ID	Integer
Responsabilidad:	

Nombre:	
Descripción:	

Tabla 3.10: Clase Nomenclador Etnias.

Nombre: Nivel educacional. (Nomenclador) (CU – L)	
Tipo de clase: Entidad.	
Atributo	Tipo
Nombre	String
ID	Integer
Responsabilidad:	
Nombre:	
Descripción:	

Tabla 3.11: Clase Nomenclador Nivel educacional.

Nombre: Fuente de aceptación. (Nomenclador) (CU – L)	
Tipo de clase: Entidad.	
Atributo	Tipo
Nombre	String
ID	Integer
Descripción	String
Responsabilidad:	
Nombre:	
Descripción:	

Tabla 3.12: Clase Nomenclador Fuente de aceptación.

Clase Base_Datos: Su objetivo fundamental es conectarse a una Base de Datos MySQL y ejecutar consulta SQL a la misma.

Nombre: C Base_Datos (CU – L, R, M)	
Tipo de clase	
Atributo	Tipo
Responsabilidad:	
Nombre:	EjecutarConsulta()
Descripción:	Es la funcionalidad que permite ejecutar las consultas a la Base de Datos.

Tabla 3.13: Clase Base de Datos.

Clase Motor: La encargada de crear, modificar, obtener la dirección de los servidores de Bases de Datos que interactúan con el sistema.

Nombre: C_Motor (CU – L, R, M)	
Tipo de clase	
Atributo	Tipo
Responsabilidad::	
Nombre:	ObtenerHost()
Descripción:	Obtener los host de los servidores de Bases de Datos que interactúan con el sistema.
Nombre:	ObtenerHostActivos()
Descripción:	Obtener los host de los servidores de Bases Datos activos.
Nombre:	ObtenerHosInactivos()
Descripción:	Obtener los host de los servidores de Bases Datos inactivos.

Tabla 3.14: Clase Motor.

Clase Recopilador: Hereda de la clase Base_Datos. Tiene una instancia a la clase Motor para obtener las direcciones de conexión con los servidores. Su objetivo es poder recolectar los datos, que devuelve la ejecución de una consulta SQL, de diferentes Bases de Datos ubicadas en diferentes servidores de Bases de Datos.

Nombre: C_Recopilador (CU – L, R, M)	
Tipo de clase	
Atributo	Tipo
informacion	string
Responsabilidad::	
Nombre:	RecopilarDatos()
Descripción:	Recolecta los datos que devuelve la ejecución de una consulta SQL.

Tabla 3.15: Clase Recopilador.

Clase Cezpdf: Librería que posibilita crear un fichero con extensión PDF desde una página PHP.

Nombre: C_Cezpdf (CU – L, R, M)	
Tipo de clase	
Atributo	Tipo
y	Es una librería que permite obtener la posición en el eje x del cursor en el pdf.
Responsabilidad:	
Nombre:	addJpegFromFile()
Descripción:	Inserta una foto en formato jpg en el documento pdf.
Nombre:	addText()
Descripción:	Posiciona una cadena de texto en una posición (x,y) indicada.
Nombre:	line()
Descripción:	Traza una línea en el documento pdf, pasándole un inicio y un final.
Nombre:	ezSetDy()
Descripción:	Posiciona el cursor en el eje y del documento pdf.
Nombre:	ezTable()
Descripción:	Crea una table.
Nombre:	ezStream()
Descripción:	Muestra en el navegador el documento pdf.

Tabla 3.16: Clase Cezpdf.

Clase ConsultasResumenes: Almacena las consultas propias de cada resumen.

Nombre: ConsultasResumenes (CU –R)	
Tipo de clase	
Atributo	Tipo
Responsabilidad:	
Nombre:	__call()
Descripción:	Es la funcionalidad que se encarga de ejecutar las consultas de los resúmenes.
Nombre:	Consulta()
Descripción:	Almacena la consulta para luego devolverla.

Tabla 3.17: Clase Consulta de resúmenes.

Clase Listado: Es la clase encargada de crear la consulta con todos los atributos que el cliente solicitó, listar y obtener la recopilación de los datos mediante la instancia a la clase Recopilar para posteriormente devolver toda la información.

Nombre: Listado (CU – L)	
Tipo de clase: controladora	
Atributo	Tipo
Responsabilidad:	
Nombre:	Analizador()
Descripción:	Se encarga de analizar todos los datos del listado.
Nombre:	ObtenerListado()
Descripción:	Es la funcionalidad que obtiene el listado para luego devolverlo.
Nombre:	ActualizarCodigosPr()
Descripción:	Actualiza los códigos de la provincia que el usuario seleccionó para obtener la información.
Nombre:	ActualizarCodigosMun()

Descripción:	Actualiza los códigos de los municipios que el usuario seleccionó para obtener la información.
--------------	--

Tabla 3.18: Clase Listado.

Clase Resúmenes: Clase encargada de formar un resumen. Obtiene de la clase ConsultasResumenes la consulta propia del resumen que se desea obtener, obtiene el criterio de restricción del resumen, conforma la consulta y se la pasa al recopilador que le devuelve el resultado para después obtener los totales y devolver el resumen.

Nombre: Resúmenes (CU –R)	
Tipo de clase :controladora	
Atributo	Tipo
Responsabilidad:	
Nombre:	Resumenes()
Descripción:	Almacena y devuelve los resúmenes.
Nombre:	ArmarConsulta()
Descripción:	Conforma la consulta de acuerdo al resumen seleccionado y se la pasa al recopilador que le devuelve el resultado.
Nombre:	ObtenerResumen()
Descripción:	Obtiene el resumen elaborado para después obtener los totales y devolverlo.
Nombre:	CalcularTotales()
Descripción:	Calcula los totales para mostrar el resumen completo
Nombre:	ObtenerErroresHost()
Descripción:	Obtiene los errores que se pueden detectar a partir de los host que se les solicita la información.
Nombre:	CambiarCodPorNombres()
Descripción:	Actualiza y pone en vez del código que recogió, el nombre del estudiante correspondiente.

Tabla 3.19: Clase Resúmenes.

Clase Modelo: Clase creada para obtener los modelos oficiales, apoyándose de la clase Centro de donde se obtienen los datos en forma de cantidad de estudiantes.

Nombre: ClaseModelo (CU – M)	
Tipo de clase: controladora	
Atributo	Tipo
Responsabilidad:	
Nombre:	Modelo1()
Descripción:	Almacena el modelo correspondiente, o sea a cada modelo se le asignó un número más corto para saber en cada caso a qué modelo se refiere..
Nombre:	Modelo2()
Descripción:	Almacena el modelo correspondiente para cursos diurnos.
Nombre:	Modelo2PE()
Descripción:	Almacena el modelo correspondiente para programas especiales
Nombre:	Modelo3()
Descripción:	Almacena el modelo correspondiente para cursos diurnos.
Nombre:	Modelo3PE()
Descripción:	Almacena el modelo correspondiente para programas especiales
Nombre:	Modelo4()
Descripción:	Almacena el modelo correspondiente.
Nombre:	Modelo5()
Descripción:	Almacena el modelo correspondiente.

Tabla 3.20: Clase Modelo.

Clase Centro: Se implementa para obtener la información de un centro docente. Su objetivo es obtener todos los estudiantes que están matriculados en un centro con las características de estos, además de brindar métodos para el procesamiento de estas características.

Nombre: ClaseCentro (CU – M)
Tipo de clase: controladora

Atributo	Tipo
Responsabilidad:	
Nombre:	ActualizarMatricula()
Descripción:	Almacena los datos actualizados de la matrícula.
Nombre:	ObtenerEspecificacionMatricula()
Descripción:	Obtiene los datos referentes al tipo de matrícula
Nombre:	ActualizarCPT()
Descripción:	Obtiene los datos de los cursos para trabajadores.
Nombre:	ObtenerColumna()
Descripción:	Obtener las columnas que se mostrarán en el modelo.
Nombre:	ObtenerListado()
Descripción:	Obtener el listado de los datos obtenidos.
Nombre:	CantXDosParametros()
Descripción:	Cantidad de estudiantes de acuerdo a dos parámetros escogidos.
Nombre:	CantXTres()
Descripción:	Cantidad de estudiantes de acuerdo a tres parámetros escogidos.
Nombre:	CantXUnParametro()
Descripción:	Cantidad de estudiantes de acuerdo al parámetro escogido.
Nombre:	Modelo2PorCarrera()
Descripción:	Almacena el modelo definido como número 2 por carreras.
Nombre:	CantidadPerfOCarrSexoVial()
Descripción:	Busca la cantidad de estudiantes por perfil, carrera y sexo.
Nombre:	Modelo2PE()
Descripción:	Almacena el modelo definido como número 2 para programas especiales.
Nombre:	CarrerasPorModelos()
Descripción:	Es para almacenar la distribución de carreras que lleva cada modelo.
Nombre:	Modelo3PorCarrera()
Descripción:	Almacena el modelo definido como número 3 por carrera para cursos diurnos.
Nombre:	CantCarrYAlgo()
Descripción:	Cuenta las carreras que se muestran en el listado.

Nombre:	Modelo5Pais()
Descripción:	Almacena el modelo correspondiente a matrícula y graduados de estudiantes extranjeros por países.
Nombre:	Modelo3PorCarreraPE()
Descripción:	Almacena el modelo definido como número 3 por carrera para programas especiales.
Nombre:	Modelo1PorCarrera()
Descripción:	Almacena el modelo definido como número 1 por carrera.
Nombre:	CantidadPaisCubaAnoCarrera()
Descripción:	Devuelve la cantidad de estudiantes por carrera y año de estudio.
Nombre:	Extranjeros()
Descripción:	Cantidad de estudiantes extranjeros.
Nombre:	Modelo4TipoCurso()
Descripción:	Almacena el modelo definido como número 4 por tipo de curso.

Tabla 3.21: Clase Centro.

Clase Sistema: Encargada de brindar los datos esenciales para el correcto funcionamiento de la aplicación como son las provincias, municipios, centros y brigadas docentes. Además de tener funciones con el objetivo de lograr un mejor funcionamiento como son PodarNombresLargos(), buscar el nombre de una provincia, municipio, o facultad por su código. Incluye la funcionalidad de obtener el criterio de restricción de búsqueda por brigadas según el nivel de acceso.

Nombre: C_Sistema (CU – L, R, M)	
Tipo de clase: controladora	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	ProvinciasWS()
Descripción:	Es la funcionalidad que permite obtener todas las provincias almacenada en el web service Registro de Ubicación.
Nombre:	MunicipiosWS()

Descripción:	Es la funcionalidad que permite obtener todos los municipios almacenados en el web service Registro de Ubicación.
Nombre:	FacultadesWS()
Descripción:	Es la funcionalidad que permite obtener todas unidades de salud almacenadas en el web service Registro de Unidades de Salud.
Nombre:	BrigadasCentro()
Descripción:	Permite obtener todas las brigadas de un centro determinado.
Nombre:	BrigadasXNivel()
Descripción:	Es la funcionalidad que permite buscar brigadas por nivel.
Nombre:	BuscarNombreFacultad()
Descripción:	Es la funcionalidad que permite buscar el nombre de la facultad seleccionada en el web service Registro de Unidades de Salud
Nombre:	Buscarprovincia()
Descripción:	Es la funcionalidad que permite buscar la provincia seleccionada en el web service Registro de Ubicación.
Nombre:	BuscarMunicipio()
Descripción:	Es la funcionalidad que permite buscar el municipio seleccionado en el web service Registro de Ubicación.
Nombre:	ViasIngreso()
Descripción:	Permite buscar los estudiantes por vías de ingreso.
Nombre:	Carreras()
Descripción:	Permite buscar los estudiantes por carreras
Nombre:	PerfilXCarreras()
Descripción:	Permite obtener el perfil de acuerdo al tipo de carrera.
Nombre:	CarrerasPorModelos()
Descripción:	Permite agrupar las carreras por modelos.
Nombre:	SimilitudProvincias()
Descripción:	Comparar la provincial seleccionada con el resto para devolver la información sólo de la deseada.
Nombre:	CodProvincias()
Descripción:	Devuelve el código de las provincias.

Nombre:	TiposCursos()
Descripción:	Devuelve los datos de los estudiantes por tipos de cursos.
Nombre:	ColoresPiel()
Descripción:	Devuelve los datos de los estudiantes por colores de piel.
Nombre:	Países()
Descripción:	Devuelve los datos de los estudiantes por países.

Tabla 3.22: Clase Sistema.

Clase JSON: Acrónimo de "JavaScript Object Notation", es un formato ligero para el intercambio de datos. JSON es un subconjunto de la notación literal de objetos de JavaScript que no requiere el uso de XML. En el servidor se usa el json.php para codificar un arreglo y convertirlo en texto plano que es el la información que viaja por la red y en el cliente se decodifica el texto plano recibido y se decodifica obteniendo así el mismo arreglo que se obtuvo en php pero ahora en JavaScript.

Nombre: C_JSON (CU – L, R)	
Tipo de clase	
Atributo	Tipo
Responsabilidad:	
Nombre:	encode()
Descripción:	Codifica el arreglo.

Tabla 3.23: Clase JSON.

Los **json** para validar y realizar la interacción cliente servidor con ajax.

Nombre: json (CU – L, R)	
Tipo de clase	
Atributo	Tipo
Responsabilidad:	
Nombre:	parse()

Descripción:	Para almacenar las funciones que se van a usar para validar.
Nombre:	setHeaderText()
Descripción:	Para establecer la cabecera del texto.

Tabla 3.24: Clase json.

Para validar la sección de los resúmenes y realizar la interacción cliente servidor con ajax.

Nombre: jsresumen (CU –R)	
Tipo de clase	
Atributo	Tipo
Responsabilidad:	
Nombre:	traerDatos(opcion)
Descripción:	Almacena los datos que se necesitan validar de los resúmenes

Tabla 3.25: Clase jsresumen.

Para validar las cosas que tienen en común los modelos, listados y resúmenes y realizar la interacción cliente servidor con ajax.

Nombre: jscomun (CU – L, R)	
Tipo de clase	
Atributo	Tipo
Responsabilidad:	
Nombre:	nuevoAjax()
Descripción:	Almacena la sección de datos que se refrescará.
Nombre:	Informacion()
Descripción:	Guarda la información común entre listados y resúmenes.
Nombre:	interruptorA()
Descripción:	Para cambiar la sección de datos.

Nombre:	IrA()
Descripción:	Para ir a la sección de datos que se requiera según la necesidad del usuario.

Tabla 3.26: Clase jscomun.

Para validar la sección de los listados y realizar la interacción cliente servidor con ajax.

Nombre: jslistados (CU – L)	
Tipo de clase	
Atributo	Tipo
Responsabilidad:	
Nombre:	TraerDatos()
Descripción:	Almacena los datos que se mostrarán y comprueba que sean legibles.
Nombre:	VerListados()
Descripción:	Valida la sección de ver los listados que debe hacer el usuario antes de mostrarlos.
Nombre:	ValidarAtributos()
Descripción:	Valida los atributos que se van a mostrar en el listado.
Nombre:	OrdenarPor()
Descripción:	A la hora de ordenar tiene en cuenta los criterios que se seleccionaron en el mismo orden, o sea el usuario elige como quiere que se muestren sus datos y esta funcionalidad permite ordenarlos por la preferencia escogida.

Tabla 3.27: Clase jslistados.

Clase aw: Librería en JavaScript para crear tablas dinámicas en el cliente.

Nombre: aw (CU – L, R)	
Tipo de clase	
Atributo	Tipo

Responsabilidad:	
Nombre:	Grid()
Descripción:	Para insertar los datos recolectados.
Nombre:	setFooterText()
Descripción:	Para insertar el texto en el pie de página.
Nombre:	setRowCount()
Descripción:	Para insertar la cantidad de filas necesarias.
Nombre:	setHeaderText()
Descripción:	Para establecer la cabecera del texto.

Tabla 3.28: Clase aw.

Para validar la sección de los modelos y realizar la interacción cliente servidor con ajax.

Nombre: jsmodelos (CU – M)	
Tipo de clase	
Atributo	Tipo
Responsabilidad:	
Nombre:	Metodo()
Descripción:	Recoger los datos que se llevará el modelo y verificar que son reales.

Tabla 3.29: Clase jsmodelos.

3.3 Definiciones de diseño.

Diseñar requiere principalmente consideraciones funcionales y estéticas. Esto necesita de numerosas fases de investigación, análisis, modelado, ajustes y adaptaciones previas a la producción definitiva del objeto. Además comprende multitud de disciplinas y oficios dependiendo del objeto a diseñar y de la participación en el proceso de una o varias personas. Es una tarea compleja y dinámica. Es la integración de requisitos técnicos, sociales y económicos, necesidades biológicas, con efectos psicológicos y materiales, forma, color, volumen y espacio, todo ello pensado e interrelacionado con el

medio ambiente que rodea a la humanidad y con el entorno donde se pondrá en práctica el diseño realizado.

El proceso de diseño del software exige buena calidad a través de la aplicación de principios fundamentales de diseño, metodología sistemática y una revisión exhaustiva. Es un proceso iterativo. Inicialmente se presenta en un nivel de abstracción alto y cada vez se refina más. La calidad de su evolución se evalúa mediante revisiones técnicas formales. Se trató de diseñar un sistema lo más flexible y extensible posible, es decir, se puede ajustar fácilmente como componente tratando de fomentar su reusabilidad. Además de un diseño bastante uniforme, tanto en la interfaz como en la codificación para hacer más entendible y reutilizable el sistema.

3.3.1 Diseño de las Salidas.

Para la mayoría de los usuarios la salida es la única razón para el desarrollo de un sistema y la base de evaluación de su utilidad. En este caso sí, porque el sistema sólo es capaz de brindar información a los usuarios, o sea, estos no pueden actualizar ni modificar ninguna información, sólo obtenerla. Todas las salidas que se ofrecen se podrán obtener en formato PDF.

Para el diseño del formato de los PDF se tuvo en cuenta las condiciones de impresión en los Centro de Enseñanza Médica de la Salud para lograr una mayor calidad a la hora de imprimir una determinada salida. Dentro de los aspectos que se tuvieron en cuenta se encuentran los colores, el tamaño y tipo de letra, el formato de las tablas, un diseño uniforme para todas las salidas para una mejor visibilidad para el cliente. Brinda además dos tipos de formatos a la hora de mostrar el PDF en dependencia de las posibilidades del cliente, presenta un formato de colores por si el cliente quiere visualizar la información de una manera más atractiva y tiene los recursos para imprimirla de esa forma y otra variante que es solo con matices de grises.

3.4 Tratamiento de errores.

El tratamiento de errores es parte importante de todo desarrollo de sistemas informáticos. Es necesario dar a conocer al usuario lo que está haciendo mal y cómo precisamente debe hacerlo y mostrarle los resultados que obtiene. Es preciso validar en un sistema la mayor cantidad de cosas posibles para asegurar un mejor almacenamiento de la información y procesamiento de los datos y evitar las páginas

de error de las aplicaciones, que suelen ser páginas por defecto que ofrecen información detallada acerca del error, información que puede ser aprovechada por el cliente para un uso malintencionado.

En este sistema como el usuario la única operación que puede realizar es obtener información, ni insertar ni modificar, se minimizan los errores que pueden provocar la introducción manual de los datos. El tratamiento de errores en la aplicación está presente del lado del cliente, utilizando código Java Script combinado con el uso de expresiones regulares que brindan una fortaleza garantizando rapidez de ejecución del sistema ya que se garantiza que la información recibida por el servidor esté correcta, en caso de existir errores, se mostrarán mensajes que aclaren al usuario sobre su proceder. Otros errores pueden ser generados por el Gestor de Base de Datos, estos se capturan antes de mostrarse al usuario y una vez tratados por el sistema, son mostrados al usuario de una manera entendible.

3.5 Seguridad.

La seguridad en sistemas de información es un tema en constante crecimiento y difusión. La gran necesidad de asegurar que la información no sea robada ni eliminada ni modificada por terceros ajenos al sistema es hoy uno de los más fuertes retos. Entre los problemas de seguridad más frecuentes, se encuentran la apropiación indebida de información confidencial, propiedad intelectual, datos financieros o registros médicos, abuso de privilegios de Internet y virus informáticos.

El tema de la seguridad es un punto fundamental por las características de la información que se maneja. Este módulo será integrado al Sistema de Gestión de la Información en el Proceso de Formación de Recursos Humanos en la Salud que cuenta con un módulo de Administración que es el encargado de establecer los permisos y privilegios a los usuarios que podrán acceder al Módulo de Salidas para solo visualizar la información, implementando un modelo de autenticación, autorización y auditoría para los usuarios, logrando un alto grado de:

- **Confiable:** La información manejada por el sistema está protegida de acceso no autorizado.
- **Integridad:** La información manejada por el sistema será objeto de cuidadosa protección contra la corrupción y estados inconsistentes.

- Disponibilidad: Los usuarios autorizados tendrán acceso a la información en todo momento.

En lo que respecta a la prevención de inyecciones de código malicioso en el sistema, que permita mostrar información a intrusos, se evita este tipo de ataque utilizando funciones propias de PHP combinadas con expresiones regulares y validaciones implementadas por los desarrolladores.

3.6 Interfaz.

El diseño de la interfaz describe cómo se comunica el software con los sistemas que operan junto con él y con los operadores y usuarios que lo emplean. Es otro de los puntos fundamentales a tratar a la hora de la presentación de la aplicación teniendo en cuenta que esa es la capa de presentación al usuario y por lo tanto debe ser lo más amigable posible, por lo que la interfaz debe resultar coherente, sensata y comprensible para el usuario corriente.

En este caso la aplicación está dirigida a profesionales de la salud que en ocasiones tienen un conocimiento mínimo del uso de la computadora, teniendo en cuenta esto, los esfuerzos de diseño están orientados a lograr una interfaz clara y fácil de usar, ya que a través de la misma es que se interactúa con los recursos que se ponen a disposición del usuario en el ordenador. Con este principio, en el diseño se tuvieron en cuenta aspectos como: organización de los elementos en la pantalla y cómo se estructura la misma, para lo cual todas las páginas tienen un diseño uniforme garantizando que el usuario no se pierda en la navegación, a continuación se explican algunos puntos tenidos en cuenta:

- El equilibrio en la organización de la información.
- La optimización de la cantidad de elementos en la pantalla, ayudando al fácil manejo y mejor comprensión de la información mostrada en la misma.
- La uniformidad de todos los elementos de la pantalla, donde cada uno de ellos se diseñará siguiendo un patrón de tamaño, colores y formas, utilizando hojas de estilos.

- Anticipar lo que el usuario quiere y necesita, brindándole toda la información y herramientas necesarias para cada paso del proceso.
- Lograr una autonomía propia, es decir, la aplicación y el entorno pertenecen al usuario, pero esto no significa que se abandonen las reglas ofreciéndole cierta amplitud de maniobrabilidad.
- Mantener un nivel alto de consistencia a través de toda la aplicación, reutilizando los criterios con que se diseñaron los íconos, diálogos, formularios, mensajes informativos, etc.
- Lograr que los elementos que se repitan en las distintas páginas, se sitúen en un mismo lugar para mejor manejo de la información.
- Trabajar sobre la base de que las páginas no se encuentren muy cargadas, solo la información necesaria para mayor claridad.

3.7 Concepción de la ayuda.

En muchas ocasiones se cuenta con un sistema que puede ser muy útil para el usuario, pero por no conocer cómo manipularlo no se utiliza. De ahí la importancia que tiene la confección de una Ayuda o Manual de Usuarios que permita al cliente aprender y familiarizarse rápidamente con el sistema. En el país existe hoy muy poca experiencia en el manejo de los sistemas informáticos por parte de la mayoría de los secretarios docentes y como ellos son los usuarios del sistema se confeccionará una ayuda que explicará de forma detallada cómo funciona el sistema para cada uno de los niveles de acceso, facilitándoles el manejo del mismo.

Esta ayuda se le incluirá como una opción al sistema para que el cliente pueda consultarla cada vez que existan dudas respecto a su funcionamiento lo que le proporcionará una mayor satisfacción en su trabajo y llevará a cabo una mejor explotación del producto.

Conclusiones.

En este capítulo se ha logrado modelar todo el sistema que ha sido objeto de estudio durante el transcurso del trabajo. Se realizó la etapa de análisis y diseño, obteniendo diagramas de clases del

análisis, mostrando las relaciones que se establecen entre las clases interfaz, controladoras y entidad. En la etapa de diseño quedaron definidos los diagramas de interacción, los que se elaboraron a partir de artefactos de la etapa de análisis. Esto permitió asignar responsabilidades a las clases. También se realizaron los diagramas de clases del diseño y finalmente se ha obtenido una arquitectura base para el sistema sobre la cual se trabajará en la fase de implementación.

CAPÍTULO 4: IMPLEMENTACIÓN DEL SISTEMA

Introducción

En este capítulo se muestra la implementación del sistema en términos de componentes, es decir, una materialización del análisis y diseño obtenidos en el capítulo anterior en ficheros de código fuente, scripts, y similares. Además se describe cómo se organiza el sistema de acuerdo a los nodos específicos en el modelo de despliegue.

4.1 Diagramas de componentes.

Los diagramas de componentes como su nombre lo indica contienen componentes, pero además de eso interfaces y relaciones entre ellos. También pueden contener paquetes utilizados para agrupar elementos del modelo. Un diagrama de componentes muestra las organizaciones y dependencias lógicas entre componentes software. En cuanto a los componentes, sólo aparecen tipos de componentes, ya que las instancias específicas de cada tipo se encuentran en el diagrama de despliegue.

Dado que los diagramas de componentes muestran los componentes de software que constituyen una parte reusable, sus interfaces, y sus interrelaciones, en muchos aspectos se puede considerar que un diagrama de componentes es un diagrama de clases a gran escala. A continuación se muestra como se concibió el modelo de implementación para el Módulo de Salidas, de acuerdo al diseño realizado para la implementación del mismo.

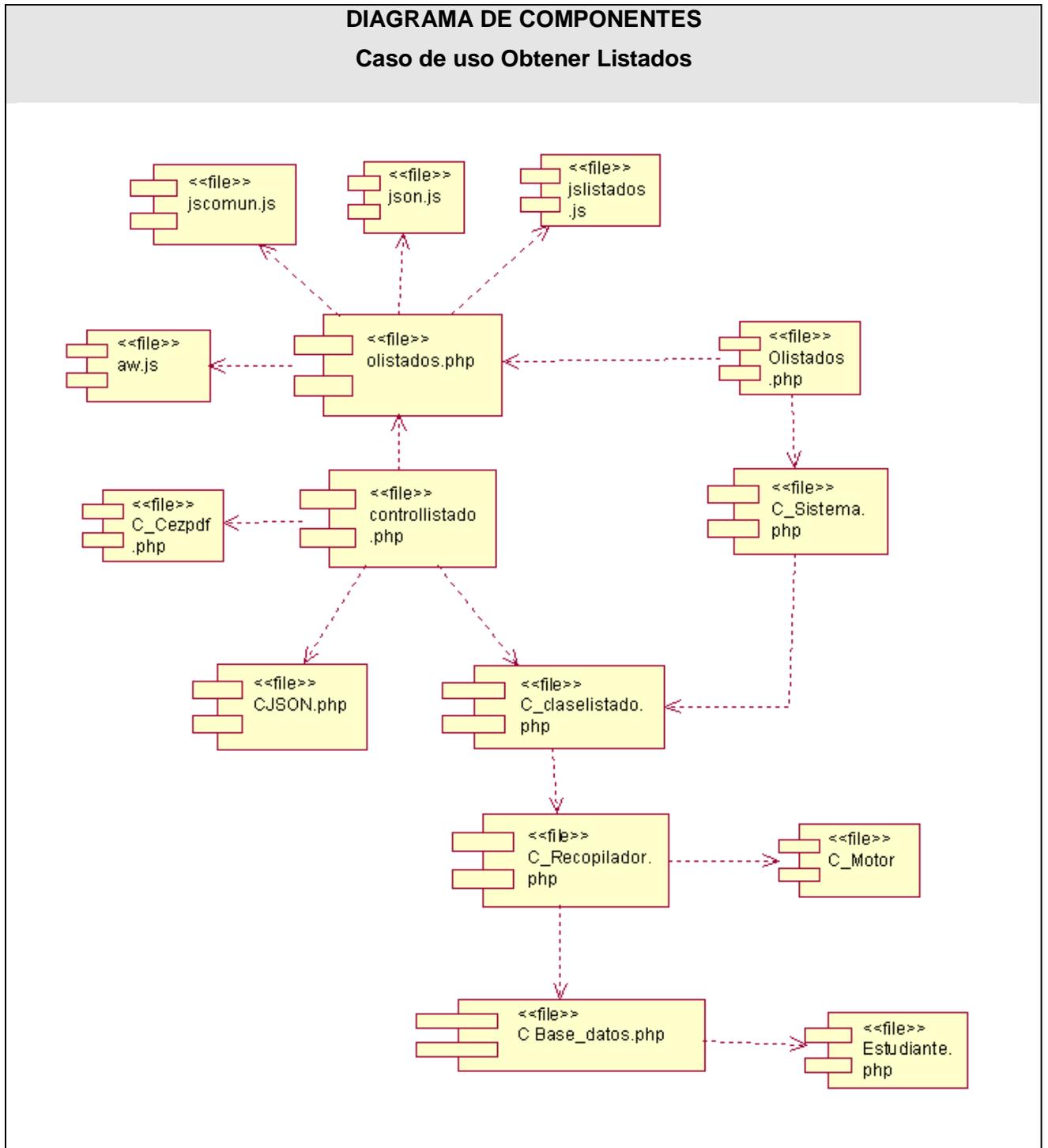


Fig. 4.1 Diagrama de componentes CU Obtener Listados.

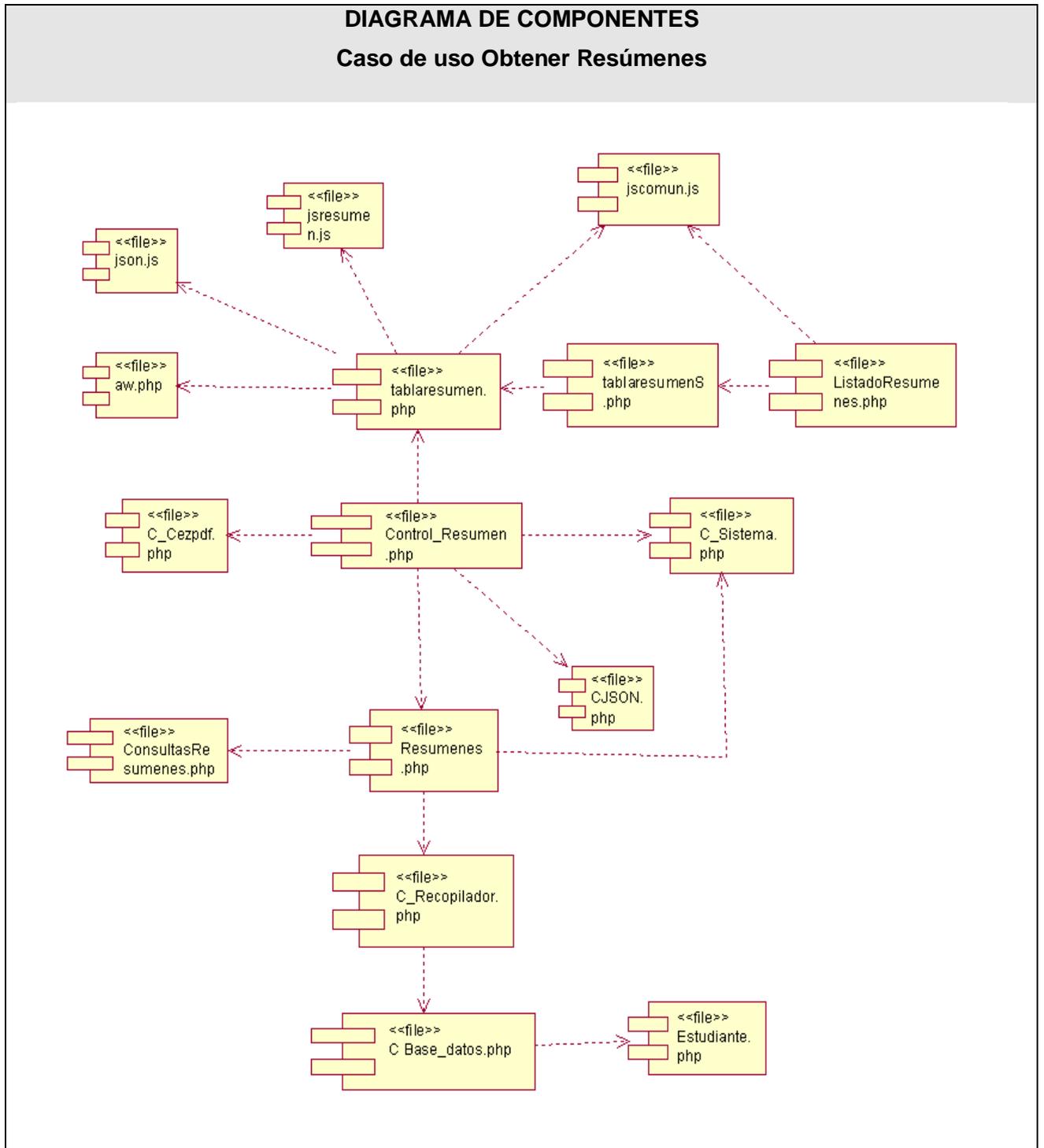


Fig. 4.2 Diagrama de componentes CU Obtener Resúmenes.

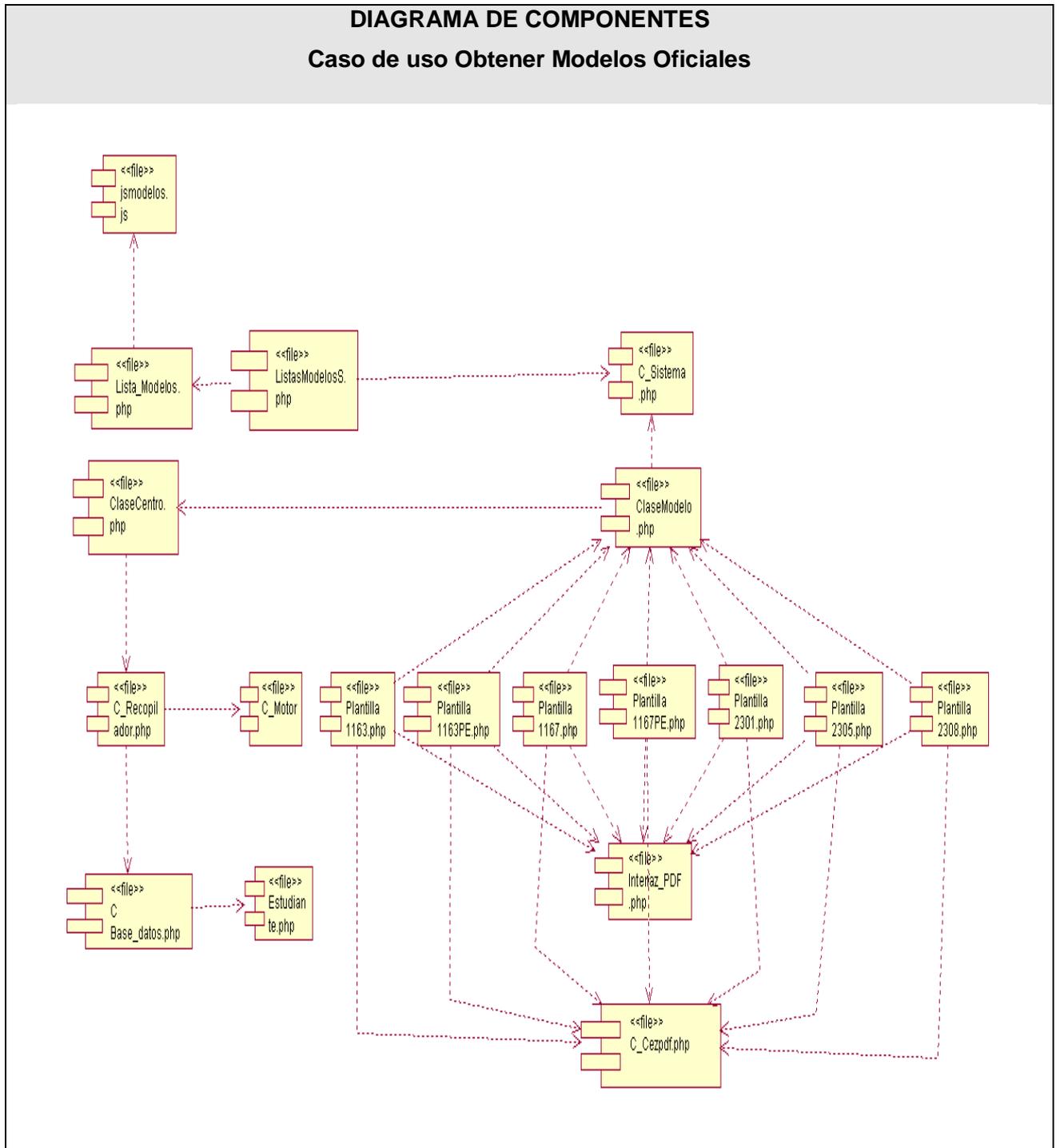


Fig. 4.3 Diagrama de componentes CU Obtener Modelos Oficiales.

4.2 Diagrama de despliegue.

Un diagrama de despliegue muestra las relaciones físicas entre los componentes hardware y software en el sistema final, es decir, la configuración de los elementos de procesamiento en tiempo de ejecución y los componentes de software, muestra la disposición física de los distintos nodos que componen un sistema y la ubicación de los componentes en dichos nodos.

A continuación se muestra una propuesta de despliegue que es como se distribuirá el sistema, la cual puede variar en dependencia de las posibilidades y recursos con que cuenta la institución cliente que en este caso es el Ministerio de Salud Pública (MINSAP). Actualmente el servidor de Aplicación es el mismo que el de Base de Datos pero pueden surgir variantes.

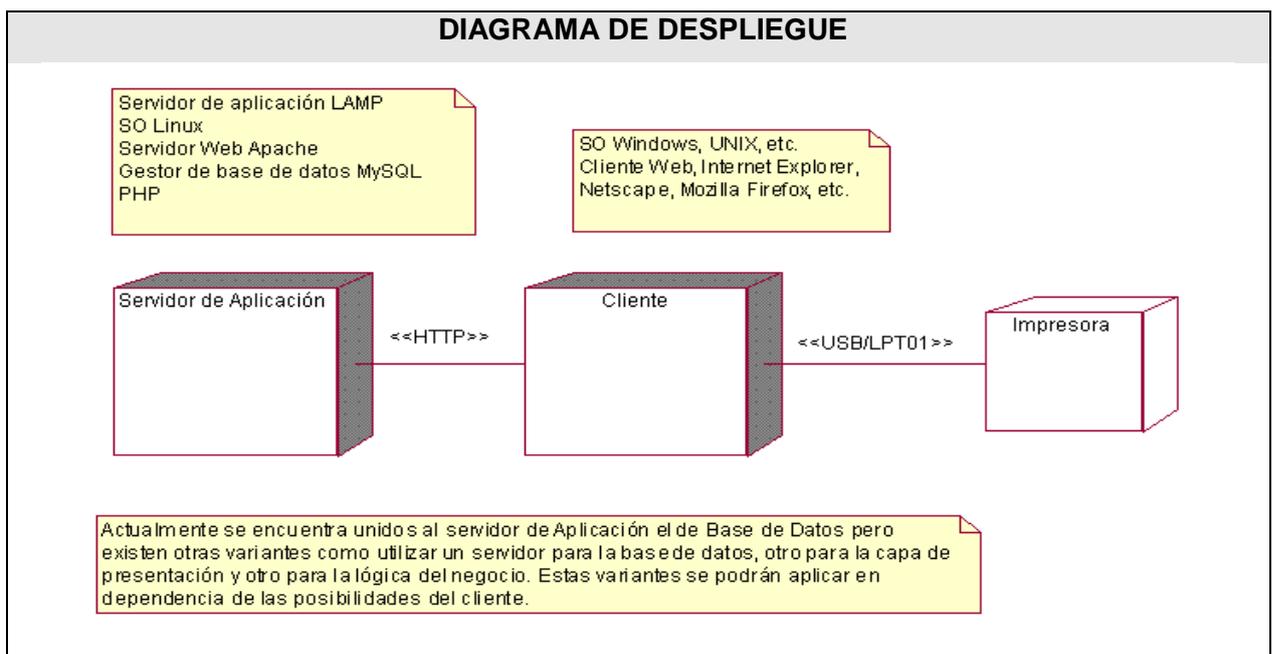


Fig. 4.4 Diagrama de despliegue del sistema.

Conclusiones.

En este capítulo se desarrolló la fase de implementación del Módulo de Salidas, desarrollándose en la misma el diagrama de despliegue, que muestra una distribución de los nodos físicos que tendrá el

mismo, así como los diagramas de componentes que reflejan la organización del código fuente en cada uno de ellos.

CONCLUSIONES

Al culminar el desarrollo del Módulo de Salidas para el Sistema de Gestión de Información en el Proceso de Formación de Recursos Humanos en Salud es válido plantear que el mismo estuvo regido y guiado por las tareas de la investigación propuestas, poniendo en práctica los conocimientos adquiridos en los cursos anteriores, cumpliendo con los objetivos trazados, y llegando a las siguientes conclusiones:

- Se profundizaron los conocimientos relacionados con la información estadística de la docencia en pregrado del MINSAP.
- Se aplicaron en la solución las tecnologías y arquitectura definidas por el área temática para el desarrollo del Módulo de Salidas.
- Se realizó el diseño e implementación de un módulo que permite el control y acceso de la información estadística relacionada con los procesos de gestión durante pregrado, el cual ya fue integrado al sistema y se espera por la etapa de implantación.

RECOMENDACIONES

- A la dirección de la Facultad que se continúe con el desarrollo del proyecto, pues pueden surgir cambios en las necesidades del cliente.
- Capacitar al personal que tendrá acceso al módulo de Salidas del sistema, para un óptimo aprovechamiento del mismo.
- Continuar con el estudio de las nuevas tecnologías para aplicarlas y renovar las utilizadas.
- Añadir nuevas funcionalidades a las ya existentes.
- Permitir obtener los reportes en otros formatos.

BIBLIOGRAFÍA

Agapea Factory S.A. Agapea Libros Urgentes. [En línea] [Citado el: 02 de 02 de 2008.]
<http://www.agapea.com/JavaScript-n10668i.htm>.

EspaWeb Internet S.L. EspaWeb Internet . [En línea] [Citado el: 10 de 02 de 2008.]
http://www.espaWeb.com/respuestas_online/PHP.html.

Facultad de Informática de la Universidad de Las Palmas de Gran Canaria. Conceptos Basicos. [En línea][Citado el: 28 de 01 de 2008.]http://gias720.dis.ulpgc.es/Gias/Cursos/Tutorial_html/concepto.htm#html.

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

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

Gran Álvarez, M. A., & Castañeda Abascal, I. (2000). Estadísticas Sanitarias.Temas docentes. Ciudad de la Habana.

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

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>.

Jorge Riverón, O., & Yera Calleja, H. (2007). Modelación de un generador de informes para sistemas de supervisión, control y adquisición de datos.

Miguel, Adoracion de y Pianttini, Mario G. Diseño de Bases de Datos. Problemas Resueltos. [En línea] [Citado el: 13 de 02 de 2008.]
<http://www.eubd.ucm.es/html/personales/enred/mantonia/docauto/tema1/Concepto.htm>.

Milenium. Diseño Web. [En línea] [Citado el: 24 de 4 de 2008.]
<http://www.informaticamilenium.com.mx/Paginas/espanol/sitioWeb.htm#dinternet>.

Osmosis Latina. Osmosis Latina. [En línea] [Citado el: 13 de 02 de 2008.]

<http://tecenciente.osmosislatina.com/curso/navegadores.htm>.

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

Programación en castellano. PHP en Castellanos. [En línea] [Citado el: 13 de 02 de 2008.]

<http://www.programacion.com/php/articulo/porquephp/>.

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

Servidor HTTP Apache. [En línea] [Citado el: 20 de 02 de 2008.]

http://es.wikipedia.org/wiki/Servidor_HTTP_Apache

Sistemas de Gestión Universitaria. [En línea] [Citado el: 12 de 11 de 2007.]

<http://www.cecam.sld.cu/pages/desarrollo/gestionuniv.htm>.

Tecnología de la Información. [En línea] [Citado el: 10 de 12 de 2007.]

www.mic.gov.cu/HThemEmp.aspx?1.

Tecnologías de la información y la comunicación . [En línea] [Citado el: 10 de 12 de 2007.]

<http://es.wikipedia.org/wiki/TIC>

Ventajas y Desventajas. [En línea] [Citado el: 12 de 1 de 2008.]

<http://www.csi.map.es/csi/silice/Global75.html>.

Wikipedia. Web Taller. [En línea] [Citado el: 06 de 02 de 2008.]

<http://www.Webtaller.com/maletin/articulos/que-es-ajax.php>.

William, Edu. Aplicaciones basadas en Web: ventajas y desventajas a nivel económico y empresarial.

[En línea] [Citado el: 20 de 01 de 2008.]

REFERENCIAS BIBILOGRÁFICAS

1. Tecnologías de la información y la comunicación . [En línea] [Citado el: 10 de 12 de 2007.]
<http://es.wikipedia.org/wiki/TIC> .
2. Tecnología de la Información. [En línea] [Citado el: 10 de 12 de 2007.]
www.mic.gov.cu/HThemEmp.aspx?1.
3. Jorge Riverón, O., & Yera Calleja, H. (2007). Modelación de un generador de informes para sistemas de supervisión, control y adquisición de datos.
4. Sistemas de Gestión Universitaria. [En línea] [Citado el: 12 de 11 de 2007.]
<http://www.cecam.sld.cu/pages/desarrollo/gestionuniv.htm>.
5. Gran Álvarez, M. A., & Castañeda Abascal, I. (2000). Estadísticas Sanitarias.Temas docentes. Ciudad de la Habana.
6. Idem Referencia 5
7. Idem Referencia 5
8. Milenium. Diseño Web. [En línea] [Citado el: 24 de 4 de 2008.]
<http://www.informaticamilenium.com.mx/Paginas/espanol/sitioWeb.htm#dinternet>.
9. William, Edu. Aplicaciones basadas en Web: ventajas y desventajas a nivel económico y empresarial. [En línea] [Citado el: 20 de 01 de 2008.]
10. Ventajas y Desventajas. [En línea] [Citado el: 12 de 1 de 2008.]
<http://www.csi.map.es/csi/silice/Global75.html>.
11. Facultad de Informática de la Universidad de Las Palmas de Gran Canaria. Conceptos Basicos. [En línea][Citado el: 28 de 01 de 2008.]
http://gias720.dis.ulpgc.es/Gias/Cursos/Tutorial_html/concepto.htm#html.

12. Agapea Factory S.A. Agapea Libros Urgentes. [En línea] [Citado el: 02 de 02 de 2008.]
<http://www.agapea.com/JavaScript-n10668i.htm>.
13. Wikipedia. Web Taller. [En línea] [Citado el: 06 de 02 de 2008.]
<http://www.Webtaller.com/maletin/articulos/que-es-ajax.php>.
14. EspaWeb Internet S.L. EspaWeb Internet . [En línea] [Citado el: 10 de 02 de 2008.]
http://www.espaWeb.com/respuestas_online/PHP.html.
15. Programación en castellano. PHP en Castellanos. [En línea] [Citado el: 13 de 02 de 2008.]
<http://www.programacion.com/php/articulo/porquephp/>.
16. Idem Referencia15.
17. Osmosis Latina. Osmosis Latina. [En línea] [Citado el: 13 de 02 de 2008.]
<http://tecncliente.osmosislatina.com/curso/navegadores.htm>.
18. Servidor HTTP Apache. [En línea] [Citado el: 20 de 02 de 2008.]
http://es.wikipedia.org/wiki/Servidor_HTTP_Apache
19. Miguel, Adoracion de y Pianttini, Mario G. Diseño de Bases de Datos. Problemas Resueltos. [En línea] [Citado el: 13 de 02 de 2008.]
<http://www.eubd.ucm.es/html/personales/enred/mantonia/docauto/tema1/Concepto.htm>.
20. Gestor de Base de Datos MySQL. [En línea] [Citado el: 25 de 02 de 2008.]
<http://www.mysql.com/>, <http://mysql.conclase.net/curso/index.php>.
21. Smarty. [En línea] [Citado el: 25 de 02 de 2008.] <http://es.wikipedia.org/wiki/Smarty>.
22. 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>.

23. Lenguaje Unificado de Modelado. UML. [En línea] [Citado el: 25 de 01 de 2008.]
http://es.wikipedia.org/wiki/Lenguaje_Unificado_de_Modelado.
24. Rational Suite. [En línea] [Citado el: 13 de 02 de 2008.]
<http://www.vico.org/TallerRationalRose.pdf>.
25. Control de versiones. [En línea] [Citado el: 25 de 02 de 2008.]
http://es.wikipedia.org/wiki/Control_de_versiones.
26. Modelos estadísticos. [Digital] Ciudad de La Habana: s.n., 2008.

ANEXO # 1: MODELO DE NEGOCIO.

Especificación de los Casos de Uso.

Caso de uso del negocio	Obtener Listados
Actores	Solicitante
Resumen	El caso de uso se inicia cuando el solicitante pide al Área de Estadística de la Secretaría Docente un listado al estadístico docente y éste busca la información solicitada en los archivos de información, una vez encontrados los datos procede a elaborar el listado con los criterios solicitados y entrega el listado al cliente.
Casos de uso asociados	
Acción del actor	Respuesta del proceso de negocio
1. El solicitante pide un listado.	2. El estadístico atiende la solicitud.
	3. Busca la información solicitada en los archivos de información.
	4. Elabora un listado con la información solicitada.
	5. Entrega el listado al solicitante.
6. El solicitante recibe el listado.	
Otras secciones	-
Mejoras propuestas	Los listados van a ser configurados directamente por el solicitante según los criterios que desee y se brindaran de una manera más rápida y eficiente de forma automática.

Tabla 1: Descripción del caso de uso Obtener Listados.

Caso de uso del negocio	Obtener Modelos Oficiales
Actores	Cliente
Resumen	El caso de uso se inicia cuando el solicitante pide un reporte contenedor de información estadística del centro al Área de Estadística de la Secretaría Docente y el estadístico procede a buscar la información solicitada en los archivos de información, una vez encontrada elabora el modelo con los datos solicitados y los entrega al cliente.
Casos de uso asociados:	
Acción del actor	Respuesta del proceso de negocio
1. El solicitante pide un modelo oficial estadístico.	2. El estadístico atiende la solicitud.
	3. Busca la información solicitada en los archivos de información.
	4. Elabora un modelo con la información solicitada.
	5. Entrega el modelo al solicitante.
6. El solicitante recibe el modelo.	
Otras secciones	-
Mejoras propuestas	Los modelos van a ser generados automáticamente.

Tabla 2: Descripción del caso de uso Obtener Modelos Oficiales.

Caso de uso del negocio	Obtener Resúmenes
Actores	Cliente
Resumen	El caso de uso se inicia cuando el solicitante pide un resumen al Área de Estadística de la Secretaría Docente, el estadístico procede a buscar la información solicitada en los archivos de información, una vez

	encontrados los datos elabora el resumen y entrega el resumen con los datos de la solicitud realizada.
Casos de uso asociados	
Acción del actor	Respuesta del proceso de negocio
1. El solicitante pide un resumen.	2. El estadístico atiende la solicitud.
	3. Busca la información solicitada en los archivos de información.
	4. Elabora un resumen con la información solicitada.
	5. Entrega el resumen al solicitante.
6. El solicitante recibe el resumen.	
Otras secciones	-
Mejoras propuestas	Los resúmenes van a ser generados de forma automática.

Tabla 3: Descripción del caso de uso Obtener Resúmenes.

Diagramas de actividades.

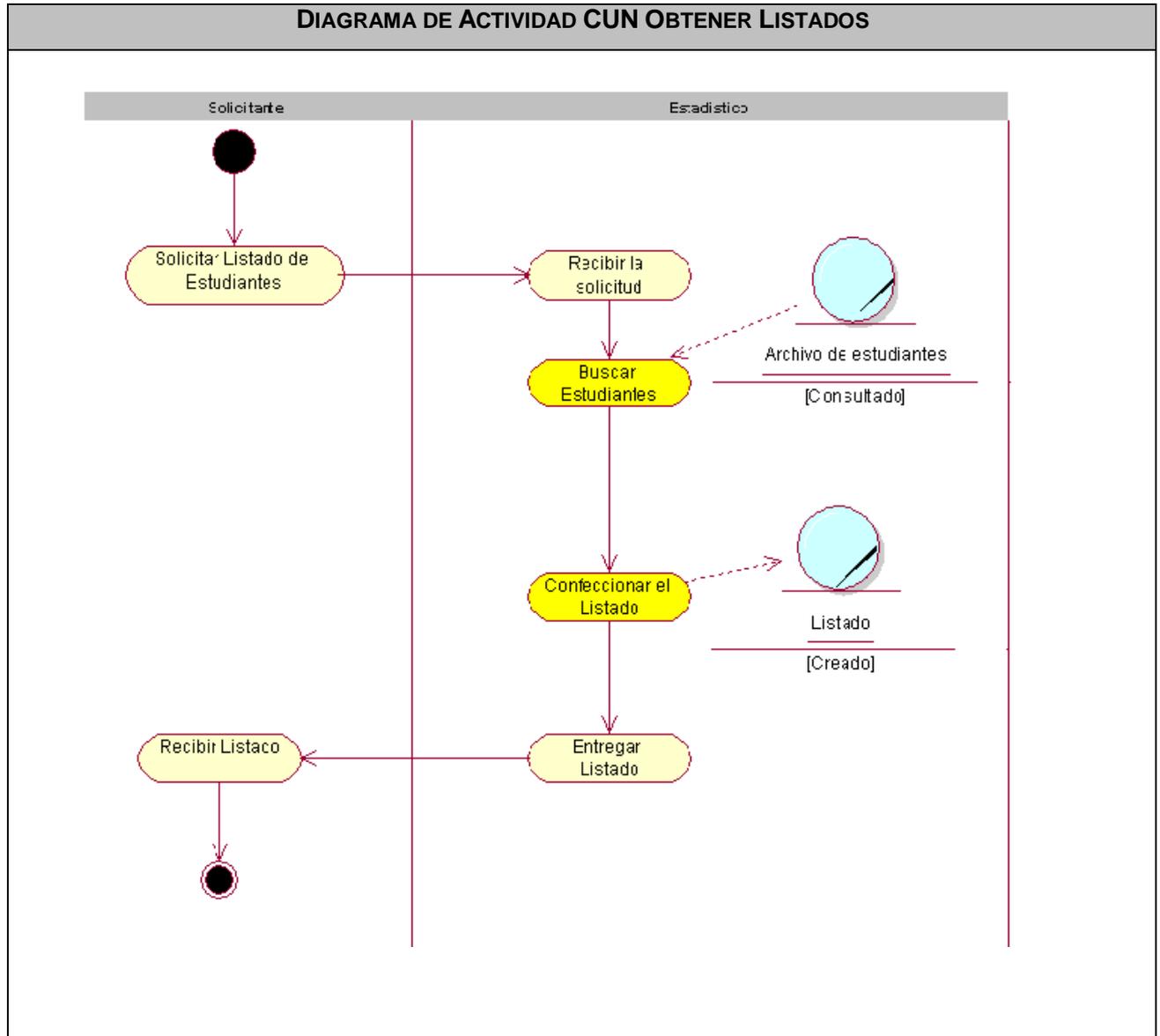


Fig. 1 Diagrama de actividades CU Obtener Listados.

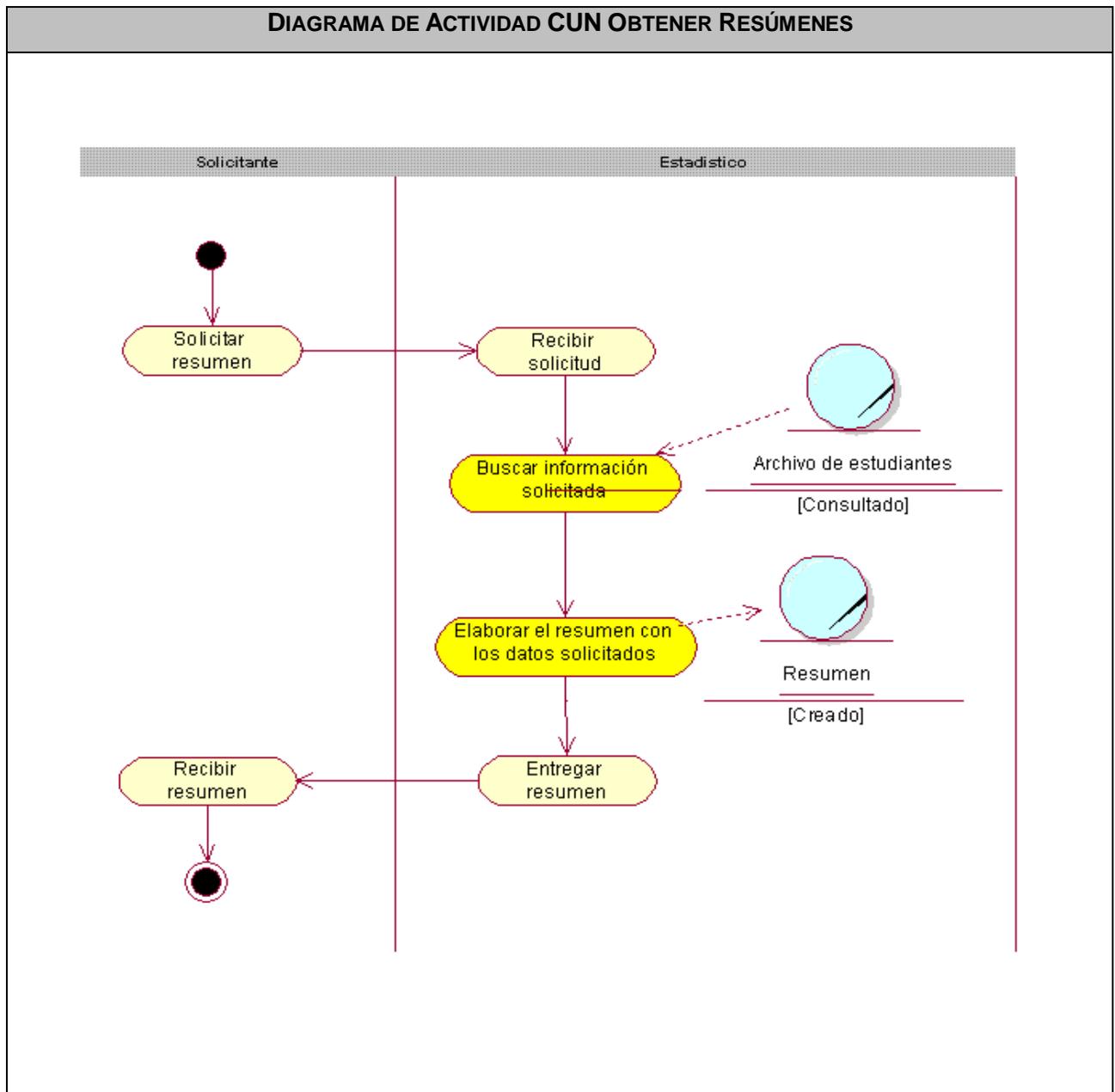


Fig. 2 Diagrama de actividades CU Obtener Resúmenes.

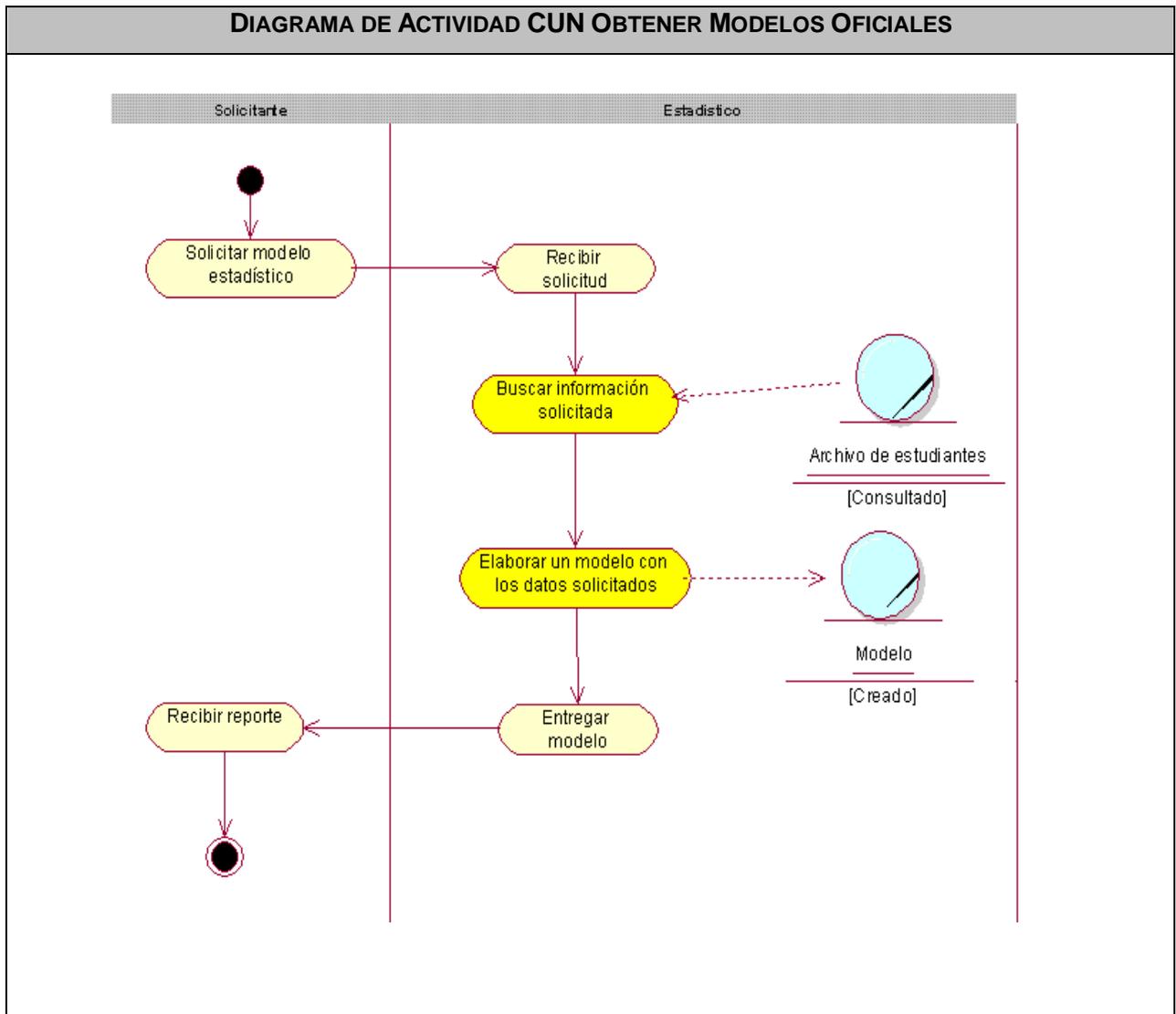


Fig. 3 Diagrama de actividades CU Obtener Modelos Oficiales.

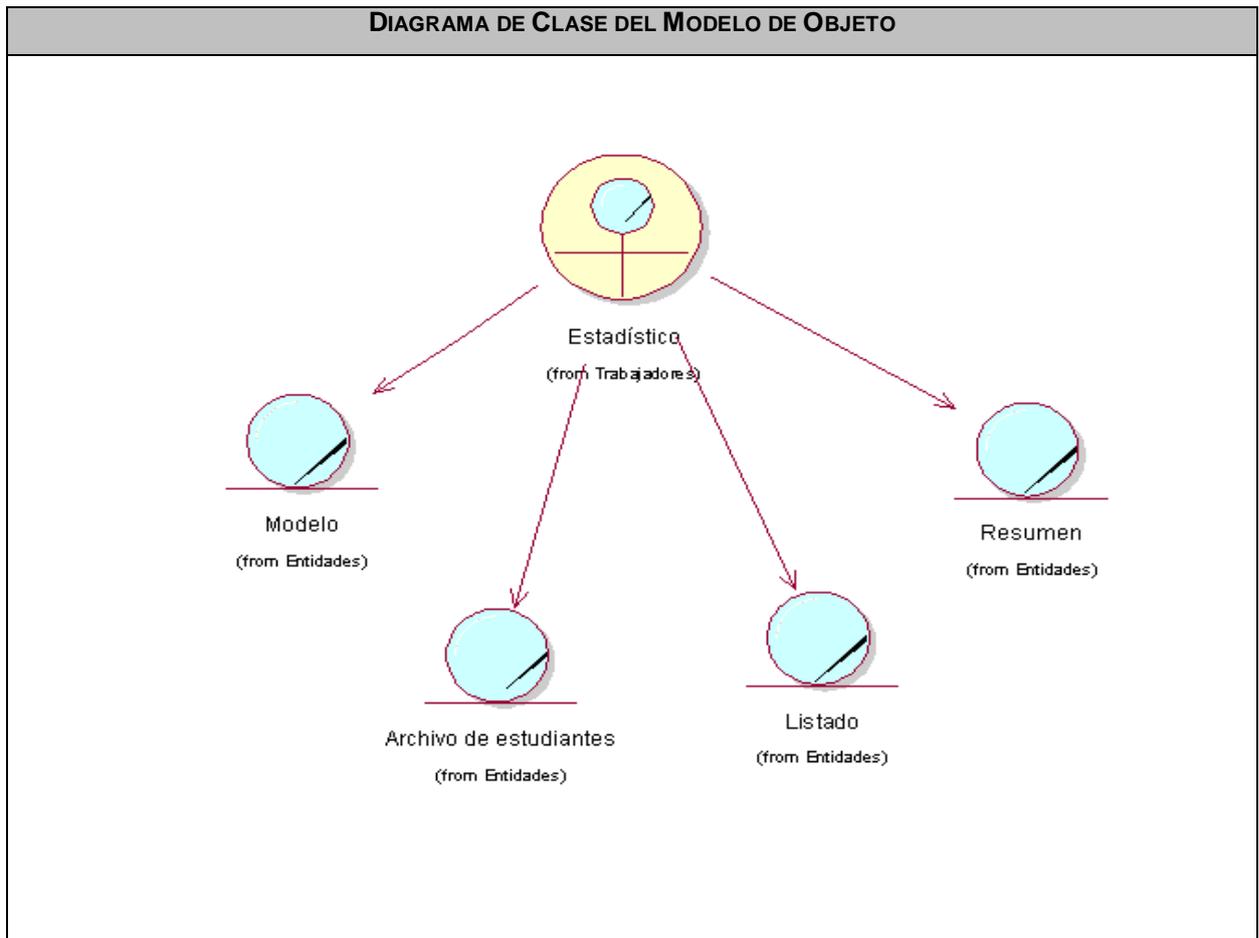


Fig. 4 Diagrama de clase del Modelo de Objeto.

ANEXO # 2: MODELO DE SISTEMA.

Descripción de Casos de uso

Caso de Uso:	Obtener Resúmenes.	
Actores:	Estadístico.	
Resumen:	Este caso de uso se inicia cuando el estadístico necesita obtener resúmenes de estudiantes por diferentes criterios haciendo cualquier tipo de cruces de variables, finalizando así el caso de uso.	
Precondiciones:	Antes de solicitar obtener el resumen debe primero Buscar Ubicación.	
Referencias	R2 - R16	
CU Asociados	Buscar Ubicación.	
Prioridad	Crítico	
Flujo Normal de Eventos		
Sección “”		
Acción del Actor	Respuesta del Sistema	
1. El estadístico accede a la sección generadora de resúmenes de la aplicación.	2. El sistema muestra una interfaz con las principales agrupaciones por tipo de búsqueda (Identificación Personal y Ubicación en Cuba, Datos del País de Procedencia, Ubicación Docente, Movimiento y Documentación y Búsqueda de Estudiantes)	
3. El estadístico despliega una de las cuatros agrupaciones de búsqueda y dentro de esta selecciona el criterio de búsqueda (Re1- Re5).	4. El sistema muestra una interfaz para que el usuario seleccione los parámetros necesarios. Se ejecuta el caso de uso Buscar Ubicación.	
5. Una vez completados los datos el estadístico realiza la búsqueda.	6. El sistema muestra el resumen con los datos seleccionados.	
7. El estadístico selecciona la opción “ PDF ” para mostrar el listado en dicho	8. El Sistema muestra el resultado del resumen en formato PDF.	

formato.	
Prototipo de Interfaz	
Flujos Alternos	
Acción del Actor	Respuesta del Negocio
Prototipo de Interfaz	
Poscondiciones	Se obtiene un resumen final por los criterios seleccionados en el formato PDF.

Tabla 4: Descripción del caso de uso Obtener Resúmenes.

Criterios de búsqueda para los Resúmenes.

Re1 Identificación Personal y Ubicación en Cuba

- Estado Civil
- Estudiantes Indígenas
- Estudiantes por Edad- Sexo
- Estudiantes Etnias Centro
- Estudiantes Por Religión
- Estudiantes por Procedencia Social
- Estudiantes Por Color Piel
- Estudiantes Creyentes y No Creyentes
- Por Número Hijos

Re2 Datos del País de Procedencia

- Ingreso
- Nacionalidad

Re3 Ubicación Docente, Movimiento y Documentación

- Cantidad de Estudiantes por Centro

Re4 Búsqueda de Estudiantes

- Estudiantes por Año Académico
- Estudiantes por Fecha de Matrícula

Re5 Nuevo resumen

- Resumen configurable seleccionando la fila y la columna.

Caso de Uso:	Obtener Listados.
Actores:	Estadístico.
Resumen:	Este caso de uso se inicia cuando el estadístico necesita obtener Listados de estudiantes por diferentes criterios de búsqueda permitiéndole a este seleccionar los datos que necesita finalizando el caso de uso.
Precondiciones:	Antes de solicitar obtener el listado debe primero Buscar Ubicación.
Referencias	R1
CU Asociados	Buscar Ubicación.
Prioridad	Crítico
Flujo Normal de Eventos	
Sección “”	

Anexo #2 Modelo del Sistema

Acción del Actor	Respuesta del Sistema
1. El estadístico accede a la sección generadora de Listados de la aplicación.	2. El sistema muestra una interfaz con criterios de búsqueda de acuerdo al nivel del estadístico. Se ejecuta el caso de uso Buscar Ubicación. Debe especificar los datos personales del estudiante que quiere visualizar.
3. El estadístico accede a la opción Buscar	4. El sistema muestra el listado con los datos seleccionados.
5. El estadístico selecciona la opción " PDF " para mostrar el listado en dicho formato.	6. El Sistema muestra el resultado del listado en formato PDF.
Prototipo de Interfaz	
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
Prototipo de Interfaz	
Poscondiciones	Se obtiene un listado final con los datos seleccionados en el formato PDF.

Tabla 5: Descripción del caso de uso Obtener Listados.

Caso de Uso:	Obtener Modelos oficiales.
Actores:	Estadístico.
Resumen:	Este caso de uso se inicia cuando el estadístico necesita obtener Modelos oficiales estadísticos predeterminados por MINSAP y el sistema le facilita las opciones para elegir finalizando así el caso de uso.
Precondiciones:	Antes de solicitar obtener el modelo debe primero Buscar Ubicación.
Referencias	R17- R23
CU Asociados	Buscar Ubicación.

Prioridad	Crítico
Flujo Normal de Eventos	
Sección “”	
Acción del Actor	Respuesta del Sistema
1. El estadístico accede a la sección generadora de Modelos oficiales de la aplicación.	2. El sistema muestra una interfaz con criterios de búsqueda de acuerdo al nivel del estadístico. Se ejecuta el caso de uso Buscar Ubicación.
3. El estadístico accede a la opción Buscar.	4. El sistema muestra el modelo seleccionado.
5. El estadístico selecciona la opción “ PDF ” para mostrar el modelo en dicho formato.	6. El Sistema muestra el resultado del modelo en formato PDF.
Prototipo de Interfaz	
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
Prototipo de Interfaz	
Poscondiciones	Se obtiene un modelo oficial estadístico en el formato PDF.

Tabla 6: Descripción del caso de uso Obtener Modelos Oficiales.

Caso de Uso:	Buscar Ubicación.
Actores:	Estadístico, Registro de ubicación y Registro de unidades de salud.
Resumen:	El caso de uso se inicia cuando se necesita buscar la ubicación territorial por la provincia, el municipio y la facultad de un centro determinado.
Precondiciones:	Deben estar funcionando los servicios web Registro de ubicación

	y Registro de unidades de salud.
Referencias	R1 – R23
CU Asociados	Obtener Resumen, Obtener Listados y Obtener Modelos Oficiales.
Prioridad	
Flujo Normal de Eventos	
Sección “Búsqueda a nivel de provincia de una facultad”	
Acción del Actor	Respuesta del Sistema
1. El estadístico de acuerdo a su nivel de acceso selecciona los criterios de búsqueda ya sea nivel nacional, provincial, municipal y a nivel de sede.	2. El sistema busca en el Registro de Localidades la distribución de municipios y de acuerdo a la provincia seleccionada.
3. El estadístico selecciona el municipio.	4. El sistema procede a buscar en el Registro de Unidades de Salud las distintas facultades de ese municipio.
5. El estadístico selecciona la facultad donde quiere realizar la búsqueda.	6. El sistema almacena los datos seleccionados para realizar posteriormente la búsqueda.
Prototipo de Interfaz	
Flujos Alternos	
Sección “Búsqueda a nivel nacional de todas los centros”	
Acción del Actor	Respuesta del Sistema
1. El estadístico nacional selecciona la opción para obtener la ubicación de todos los centros del país.	2. El sistema busca en el registro de Localidades todas las provincias y municipios del país y en el Registro de Unidades de Salud las facultades correspondientes a estos.
Prototipo de Interfaz	
Poscondiciones	Se obtiene la ubicación deseada.

Tabla 7: Descripción del caso de uso Buscar ubicación.

ANEXO # 3: MODELO 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.

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 presenta los diagramas de los casos de uso correspondientes.

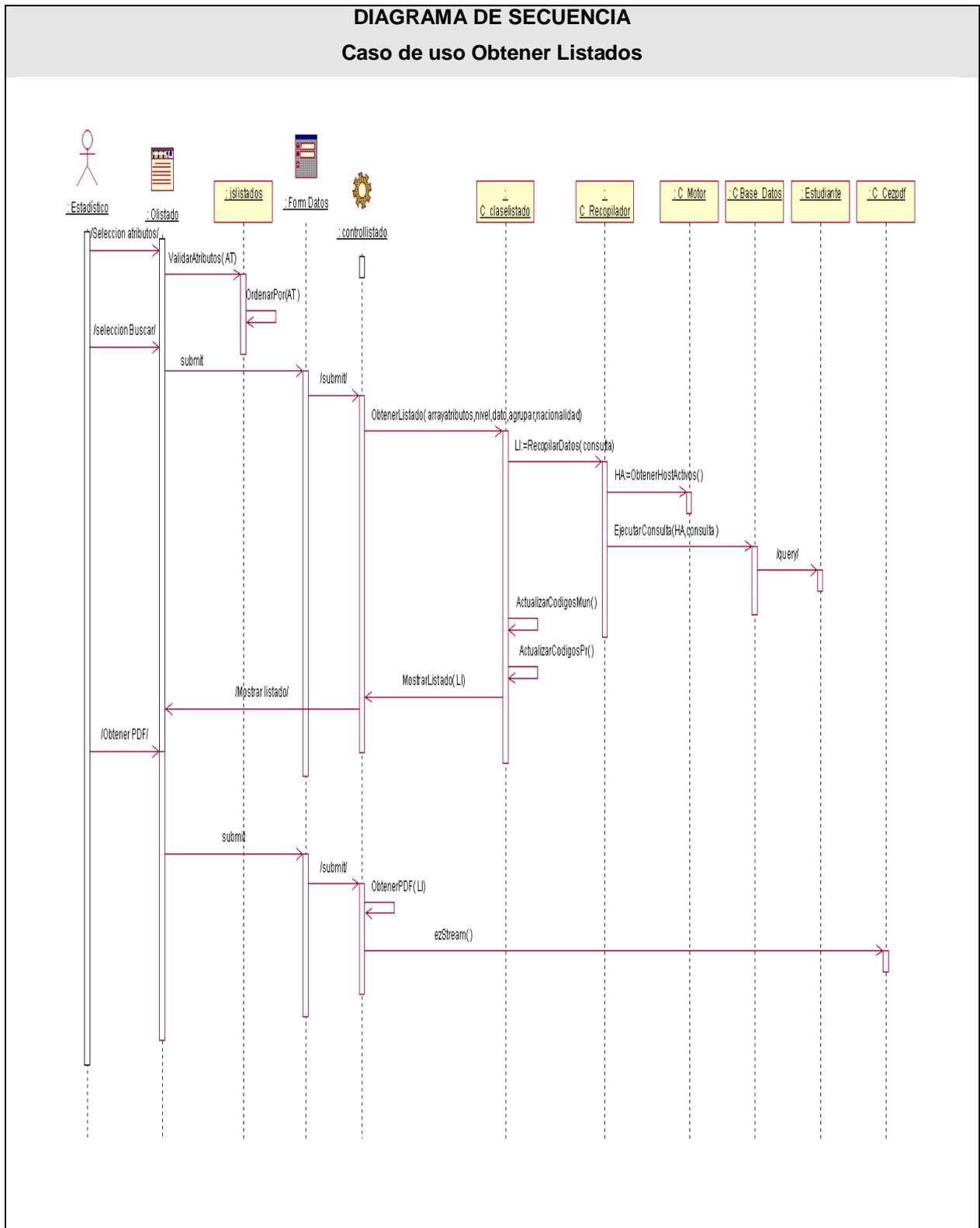


Fig. 5 Diagrama de secuencia CU Obtener Listados.

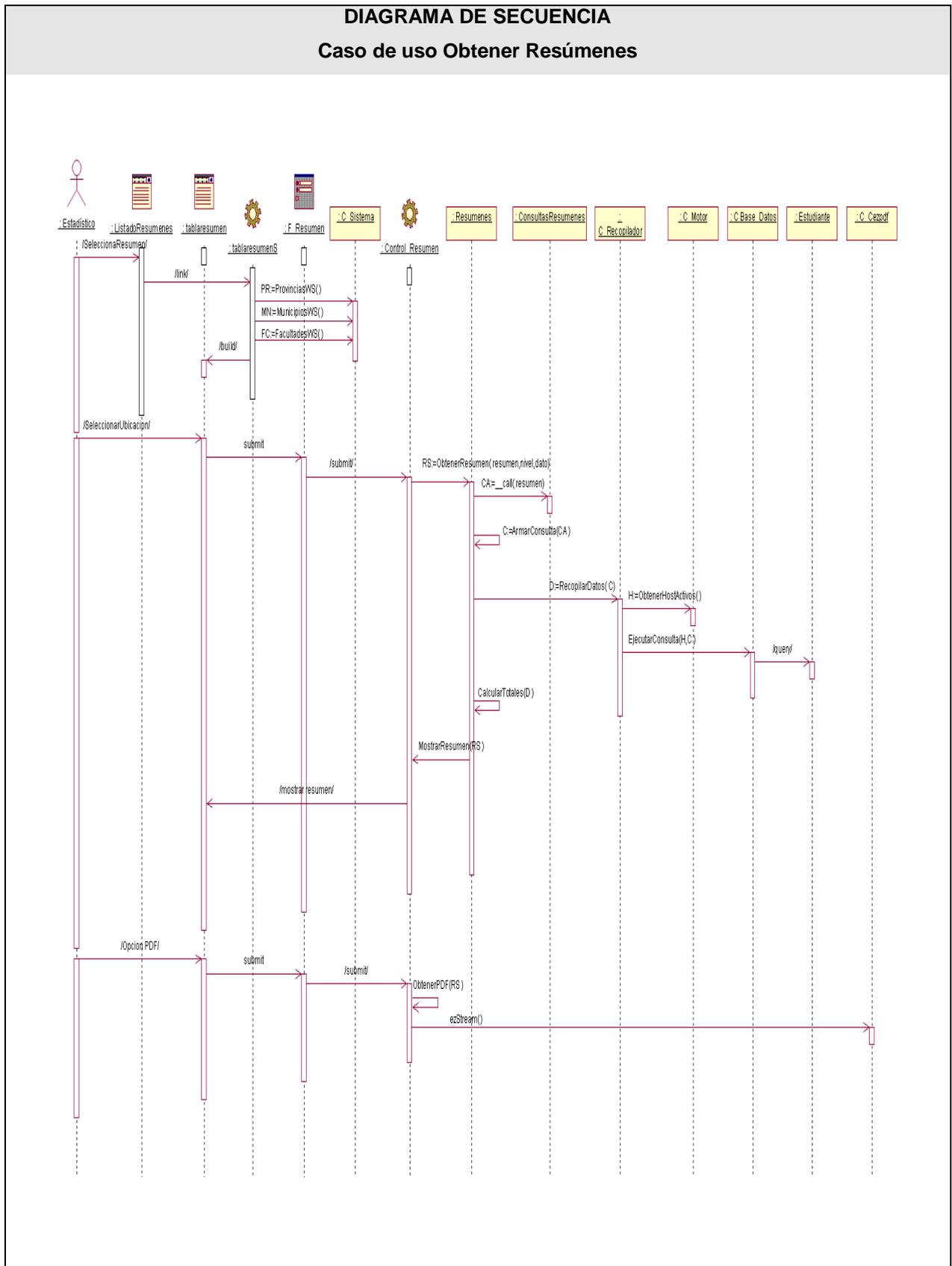


Fig. 6 Diagrama de secuencia CU Obtener Resúmenes.

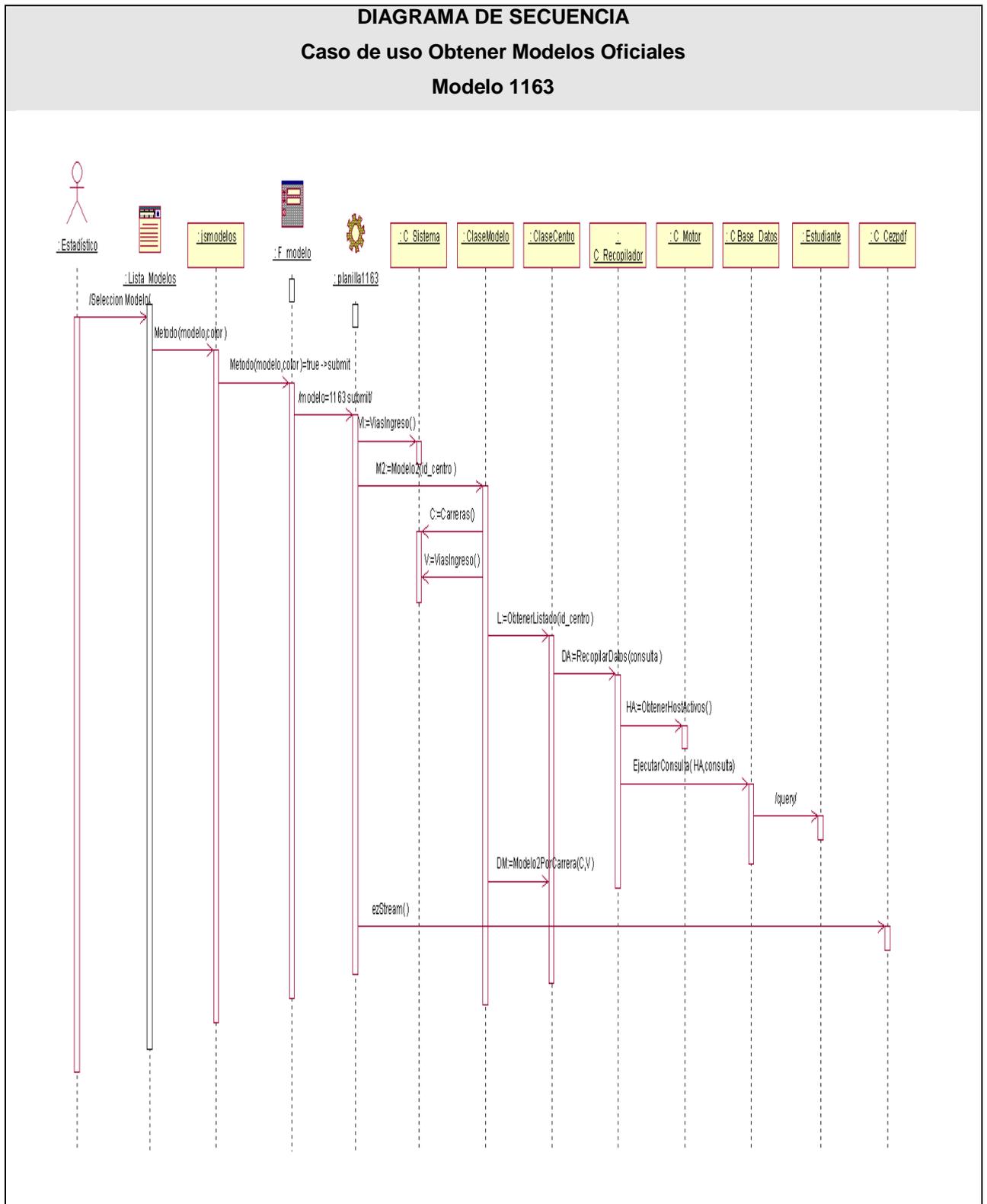


Fig. 7 Diagrama de secuencia CU Obtener Modelos Oficiales

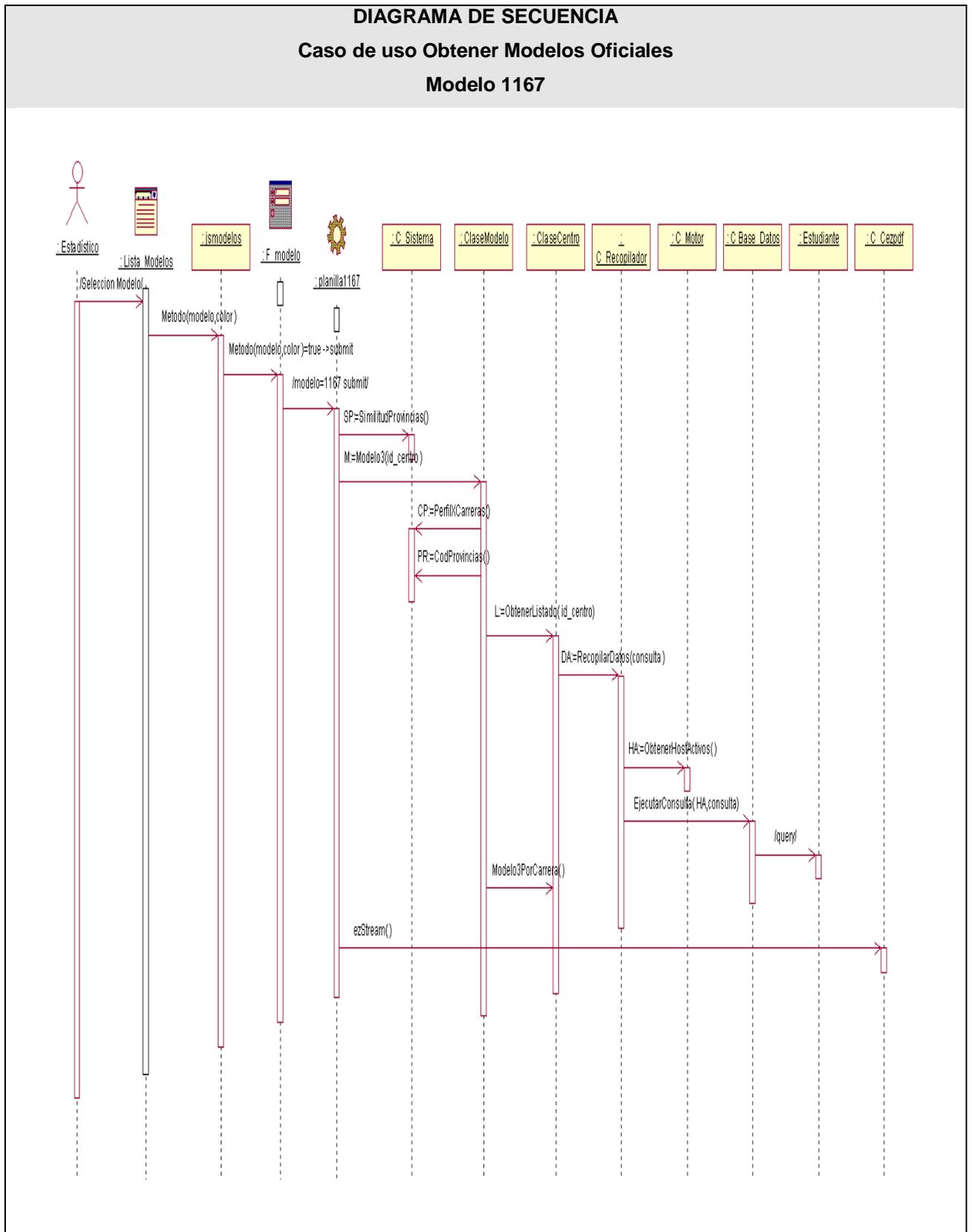


Fig. 8 Diagrama de secuencia CU Obtener Modelos Oficiales. Modelo 1167

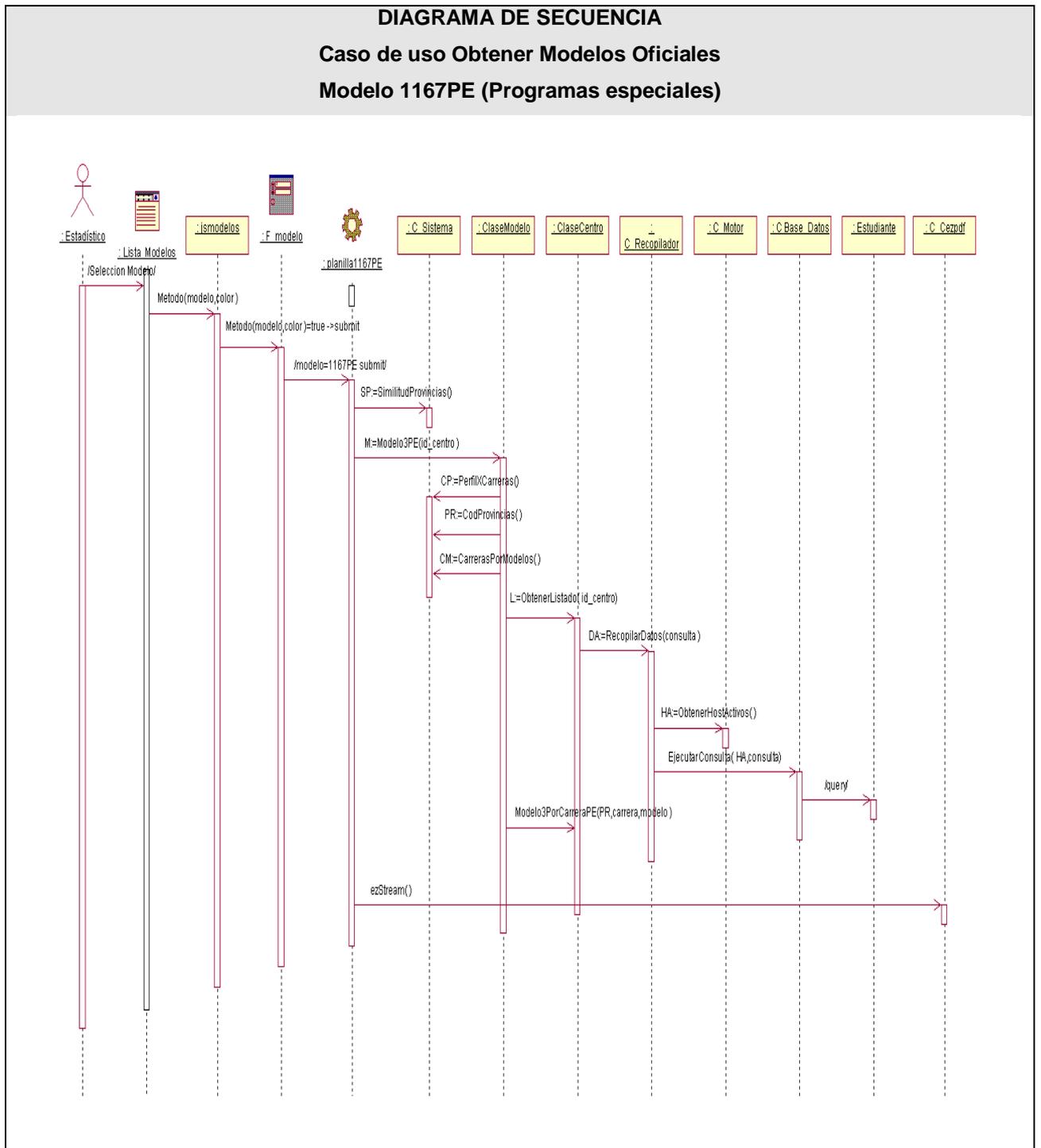


Fig. 9 Diagrama de secuencia CU Obtener Modelos Oficiales. Modelo 1167PE (Programas especiales).

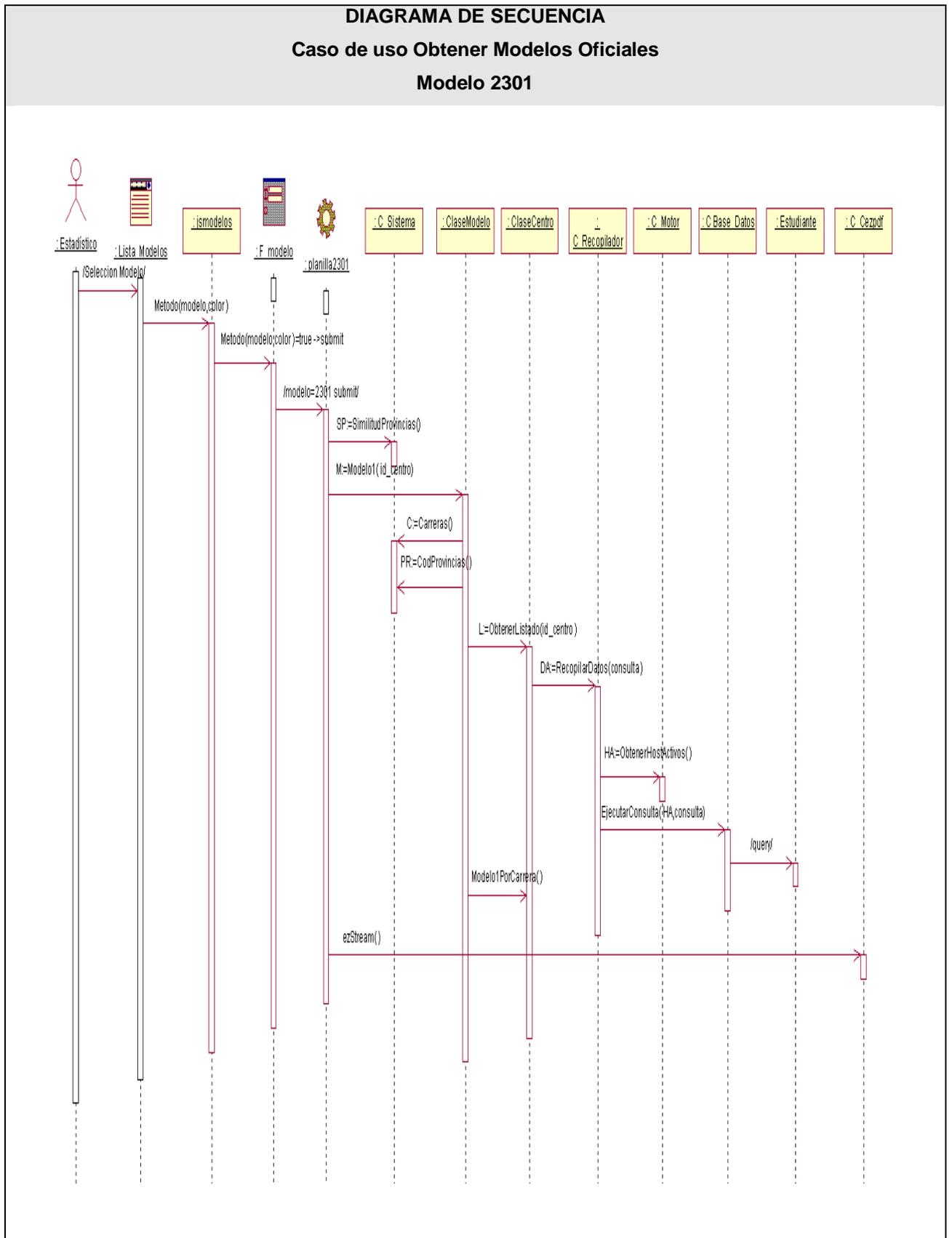


Fig. 10 Diagrama de secuencia CU Obtener Modelos Oficiales. Modelo 2301.

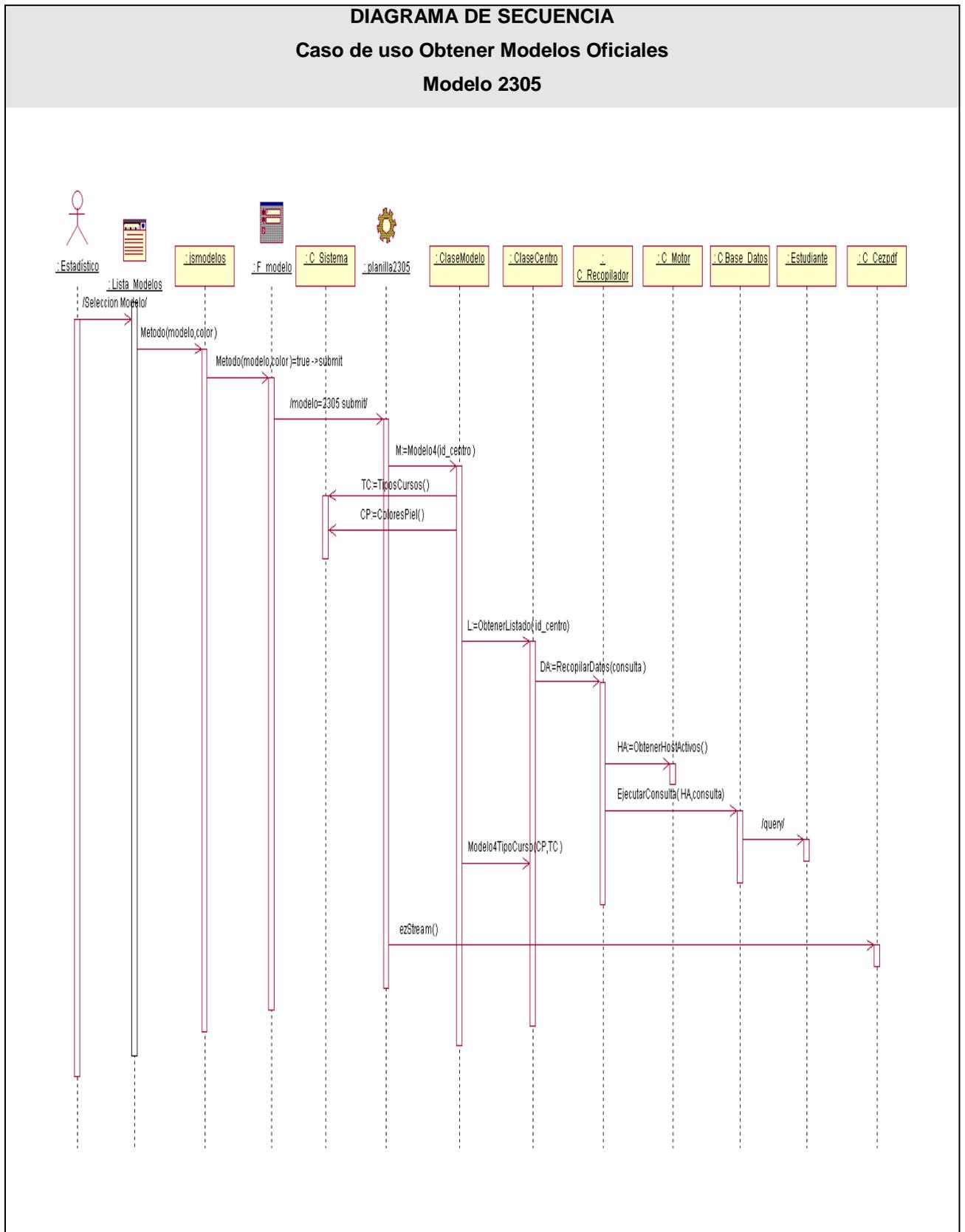


Fig. 11 Diagrama de secuencia CU Obtener Modelos Oficiales. Modelo 2305.

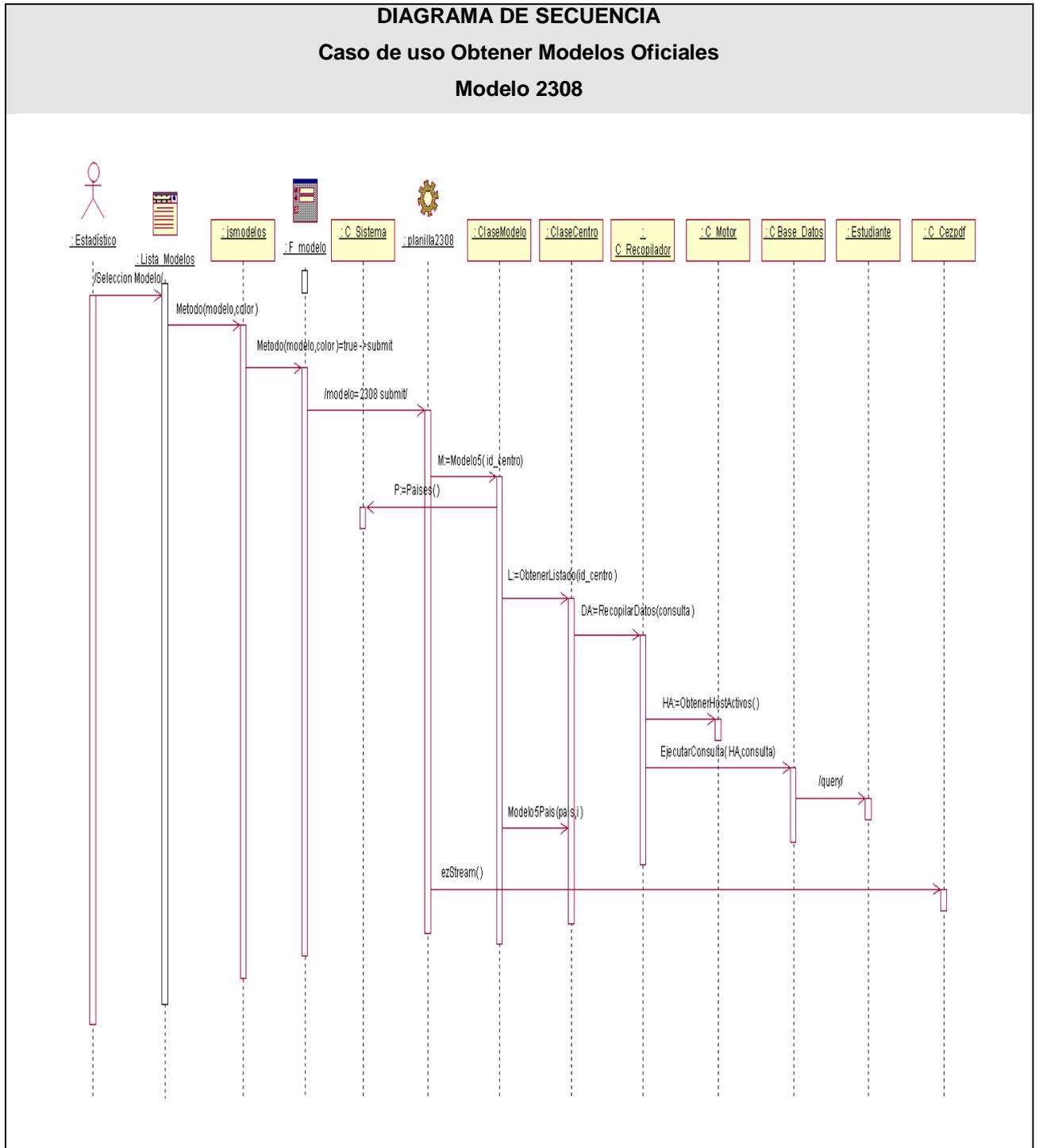


Fig. 12 Diagrama de secuencia CU Obtener Modelos Oficiales. Modelo 2308.

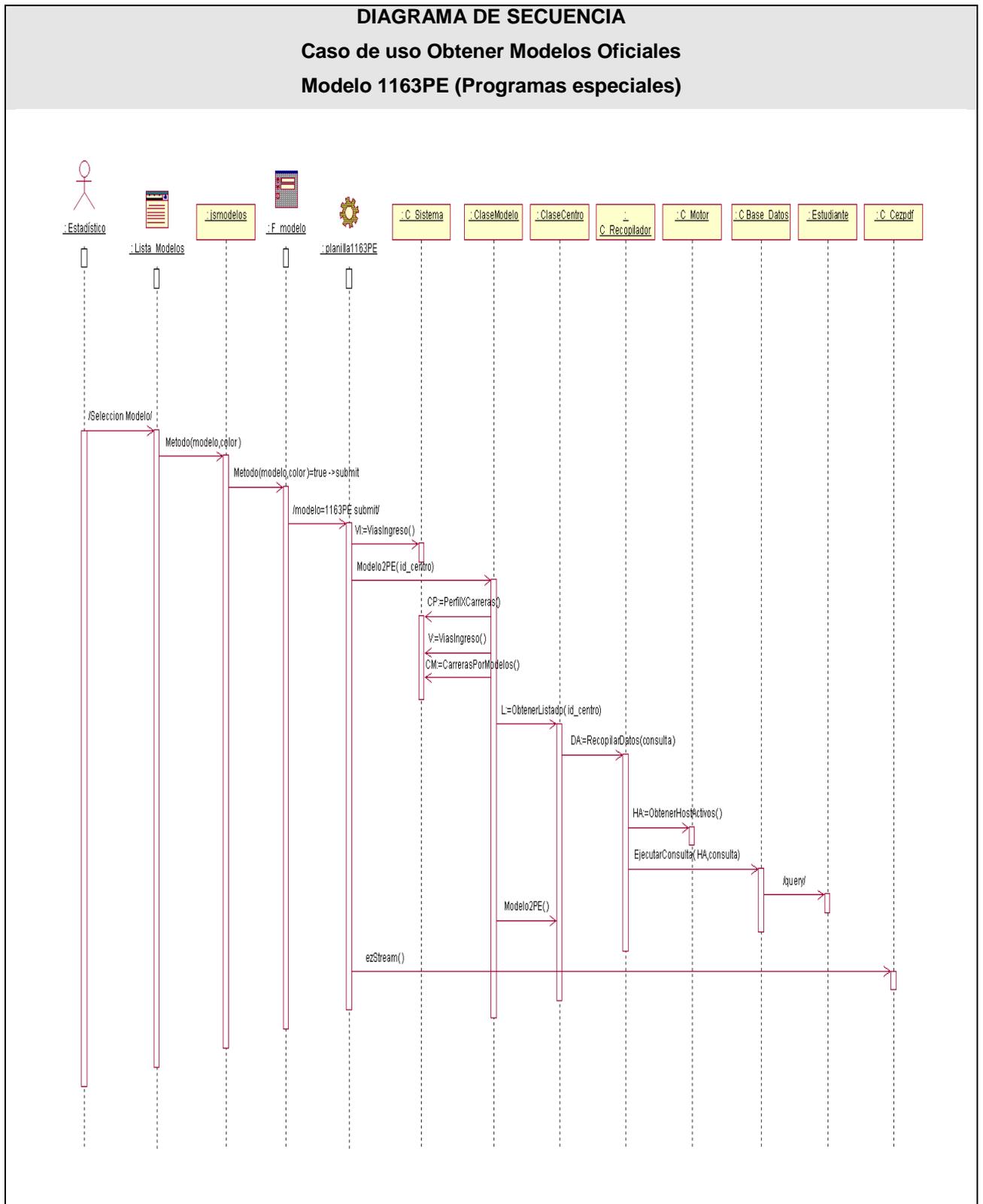


Fig. 13 Diagrama de secuencia CU Obtener Modelos Oficiales. Modelo 1163PE (Programas especiales).

De la misma manera que se cerró el ciclo de análisis, se terminó la etapa de diseño con los diagramas de secuencia. Al terminar la etapa de diseño, se han refinado suficientemente el diagrama de clases y las relaciones entre estas. También se conoce mejor los mensajes que se intercambian los objetos para realizar las tareas necesarias.