

**Universidad de las Ciencias Informáticas**  
**Facultad 2**



*Expediente Productivo e Investigativo del*  
*Estudiante.*

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

**Autores**

Emilio Izquierdo Pérez  
Luis Carlos Aguilar Paneque

**Tutor**

Ing. Anabel Parra Vázquez

Junio, 2008

## 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 firmo la presente a los \_\_\_\_ días del mes de \_\_\_\_\_ del año \_\_\_\_\_.

**<Luis Carlos Aguilar Paneque>**

\_\_\_\_\_  
Firma del Autor

**<Emilio Izquierdo Pérez>**

\_\_\_\_\_  
Firma del Autor

**<Anabel Parra Vázquez>**

\_\_\_\_\_  
Firma del Tutor

### AGRADECIMIENTOS

#### Compartidos:

*A nuestros padres por ser tan especiales y dedicados, dándonos todo el tiempo lo mejor de sí mismos.*

*A Fidel Castro Ruz y a la Revolución que hizo realidad nuestros sueños, la cual nos permitió ser las personas que somos hoy.*

*Agradecemos a todas las personas que de alguna manera han sido parte de los éxitos alcanzados hasta el momento.*

*A nuestros amigos que nos hacen grato el andar, apoyándonos en cada paso.*

*A nuestros maestros y profesores que nos guiaron y educaron a lo largo de nuestra vida estudiantil, para que hoy seamos profesionales.*

*Agradecimientos especiales a Osay González Fuentes, Alberto Tamayo, Maidelis Milanés Luque, Alberto Pantoja Ponce, Yonelbys, Maiquel, Jorge Luis y para no caer en olvido agradecerle a todos los compañeros que de una forma u otra hicieron posible este trabajo.*

#### Luis Carlos Aguilar Paneque

*Agradezco:*

*En primer lugar a mis padres por el apoyo que me han dado desde mis primeros pasos, en los buenos y en los malos momentos.*

*A Gisela y Reynaldo por tratarme como un hijo y sus muestras de amor y cariño.*

*A Yisell por ser compañera, amiga, incansable en la exigencia de todo tipo, por su apoyo incondicional ante cualquier problema y muy especial en el estudio. A ella le agradezco los resultados obtenidos.*

*Muchas Gracias.*

*Agradecerle muy especialmente a Dayami por acogerme como a un hijo, por exigirme como tal, por estar pendiente de mí en todo momento por cualquier razón que haya sido. Por ser mi madre durante 5 años, muchas gracias.*

*A Francisco y Bety por dejarme llegar y permitirme ser parte de ellos. Muchas gracias por su amistad y atención incondicional.*

*A mi primo René y su familia, por el apoyo que me dieron.*

*A mis hermanos y sobrinos, que siempre han estado presentes.*

*A mis abuelos, que siempre han estado presentes. A toda mi familia en general, tíos, tías y primos que me han dado de sí todo el amor y el cariño que una persona puede recibir. Por estar siempre pendientes de mí.*

*A todos mis amigos, amistades y conocidos que me han apoyado siempre.*

*A la Revolución por permitirme ser el profesional que soy y todo lo que me ha dado durante mi vida.*

### Emilio Izquierdo Pérez

*Agradezco:*

*Dedicasles mi agradecimiento a Barbby y Cary por ser tan buenas amigas y estar siempre pendientes de mí, a ellas dos que las quiero mucho desde lo más profundo de mi corazón.*

*A Deisy, mi hermana, Dulcita, mi sobrino y a Héctor por estar siempre presentes.*

*Agradecerle también a mis amigos que me apoyaron siempre, en especial a mi cuñado Osay por brindarme su ayuda incondicional en los momentos mas difíciles. A mi cuñada Yaima por estar siempre preocupada por mi, a pesar de estar tan lejos siempre estuvo presente su apoyo y preocupación.*

*Agradecer a toda la familia de mi novia en general, a su mamá y su papá por considerarme como otro hijo más; agradecerles mucho también a Neni y a Lulo por brindarme su más sincero cariño y quererme como un nieto. Agradecerles por último y no por ser menos importantes a Alberto y Mary por ayudarme tanto aquí en la Habana, por contar con su ayuda en todo momento.*

*A todos ellos quiero agradecerles por ayudarme y compartir conmigo buenos y malos momentos.*

### DEDICATORIA

*A mi mamá y mi papá, quienes me han forjado, guiado y apoyado durante todos los años de vida que tengo. Por ser los principales impulsores a que estudiara y pudiera ser hoy un profesional.*

*Luis Carlos*

*En ocasiones, los seres humanos suelen hacer la memoria más frágil y olvidadiza de lo que realmente es y la acusan de torpeza imperdonables. Quiero ante todo apalearse mi torpeza y respetar una de las virtudes que más aprecio, la gratitud.*

*A quienes dieron lo mejor de sí por hacer cumplir este día, a esas personas que sobrepusieron mis necesidades por encima de las suyas, quienes supieron apoyarme tanto en los buenos como en los malos momentos. A mi madre, mi padre, mi hermano, mi abuela que me apoyaron siempre y son la razón de mi fruto y mi trabajo.*

*Me gustaría también dejar plasmado en esta dedicatoria, la misma huella imborrable que tengo en mi corazón, quiero agradecerle muchísimo a mi novia Yadira L. Rodríguez Peláez por estar siempre brindándome su amor, su apoyo y su ayuda incondicional.*

*A la Revolución cubana que me dio la posibilidad de ser la persona que soy, de permitirme estudiar y convertirme hoy en un profesional.*

*Emilio*

### RESUMEN

La Universidad de las Ciencias Informáticas cuenta con un amplio movimiento productivo. Los estudiantes, desde el primer momento se vinculan a la producción, lo que les ayuda a una mejor preparación como futuros profesionales. Además se encuentra en su continuo desarrollo digital y esto se debe a la existencia de diversos proyectos, actividades de investigación y desarrollo como son: Jornadas Científicas, Mi Web por Cuba, Seminario Juvenil Martiano, Copas y Olimpíadas, UCIENCIA, Concursos de Corte investigativo/productivo y otras. El presente trabajo de diploma tiene como propósito desarrollar un sistema informático para gestionar el Expediente Productivo e Investigativo del Estudiante, lo que permitirá centralizar toda la información productiva e investigativa del estudiante de la F2. Igualmente permitirá generar la nota de Práctica Profesional a partir del análisis de todas estas actividades. Para la realización de esta tesis se revisaron 29 documentos bibliográficos, entre ellos referencias de páginas web de internet, documentos electrónicos, libros, entre otros. También se presentaron algunas razones por las cuales se utilizó PostgreSQL como gestor de base de datos y PHP como lenguaje de programación. Finalmente la propuesta de solución se sometió a un estudio de factibilidad concluyendo que su implementación es realmente viable.

## INDICE GENERAL

AGRADECIMIENTOS.....	I
DEDICATORIA.....	III
RESUMEN.....	III
INTRODUCCIÓN.....	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA.....	4
INTRODUCCIÓN.....	4
1.1 PROCESOS ASOCIADOS A LA ACTIVIDAD PRODUCTIVA E INVESTIGATIVA.....	4
1.2 SISTEMAS EXISTENTES.....	4
1.3 PROCESO DE DESARROLLO DE SOFTWARE UTILIZADO.....	4
1.3.1 Metodología Rational Unified Process (RUP).....	5
1.4 UNIFIED MODELING LANGUAGE (UML).....	6
1.5 TECNOLOGÍAS ACTUALES A UTILIZAR.....	6
1.5.1 Modelo Cliente / Servidor.....	6
1.5.2 Software Libre.....	8
1.5.3 Servidor Web Apache.....	9
1.5.4 Ajax.....	9
1.5.5 Lenguajes de programación para la Web.....	10
1.5.5.1 Del lado del cliente.....	10
1.5.5.2 Del lado del servidor.....	10
1.5.6 Framework a Utilizar.....	11
1.5.7 Sistema Gestor de Base de Datos.....	12
1.5.7.1 PostgreSQL.....	12
1.6 HERRAMIENTAS UTILIZADAS.....	12
1.6.1 Macromedia Dreamweaver.....	12
1.6.2 Dreamweaver 8.0.....	12
1.6.3 Adobe Photoshop.....	13
1.6.4 Zend Studio.....	13
1.6.5 Rational Rose.....	13
1.6.5.1 Rational Rose Enterprise.....	13
1.6.6 PgAdmin.....	14
1.6.7 Case Studio.....	14
CONCLUSIONES.....	14
CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA.....	16
INTRODUCCIÓN.....	16
2.1 MODELO DEL DOMINIO.....	16
2.2 REGLAS DEL NEGOCIO.....	18

2.3 REQUERIMIENTOS FUNCIONALES .....	19
2.4 REQUERIMIENTOS NO FUNCIONALES .....	22
2.5 ACTORES DEL SISTEMA A AUTOMATIZAR .....	23
TABLA 2: DEFINICIÓN DE ACTORES DEL SISTEMA A AUTOMATIZAR.....	<b>¡ERROR! MARCADOR NO DEFINIDO.</b>
2.6 PAQUETES Y SUS RELACIONES.....	24
2.7 DIAGRAMAS CASOS DE USOS DEL SISTEMA.....	25
2.7.1 Descripción de los casos de uso.....	25
Paquete de Administración .....	25
CONCLUSIONES .....	48
<b>CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA .....</b>	<b>50</b>
INTRODUCCIÓN .....	50
3.1 ANÁLISIS DE SISTEMA.....	50
3.2 DIAGRAMA DE CLASES DE ANÁLISIS.....	50
3.2.1 Diagramas de interacción.....	51
3.2.2 Diagramas de Colaboración.....	52
3.3 CLASES BASES.....	54
3.4 DISEÑO DEL SISTEMA.....	60
3.4.1 Diagramas de clases del diseño .....	60
3.4.1.1 Diagrama de clases Web .....	60
3.4.1.2 Principios de diseño.....	65
3.5 ARQUITECTURA EN CAPAS.....	66
3.5.1 Modelo –Vista- Controlador (MVC).....	66
3.5.2 Arquitectura cliente/servidor .....	67
3.6 DISEÑO DE LA BASE DE DATOS.....	67
3.6.1 Modelo de datos .....	67
CONCLUSIONES .....	67
<b>CAPÍTULO 4: IMPLEMENTACIÓN DEL SISTEMA.....</b>	<b>69</b>
INTRODUCCIÓN .....	69
4.1 IMPLEMENTACIÓN .....	69
4.2 DIAGRAMA DE COMPONENTE .....	69
4.3 MODELO DE DESPLIEGUE.....	71
CONCLUSIONES .....	71
<b>CAPÍTULO 5. ESTUDIO DE FACTIBILIDAD.....</b>	<b>72</b>
INTRODUCCIÓN .....	72
5.1 PLANIFICACIÓN BASADA EN CASOS DE USO .....	72
5.2 BENEFICIOS TANGIBLES E INTANGIBLES .....	77
5.3 ANÁLISIS DE COSTOS Y BENEFICIOS .....	78
CONCLUSIONES .....	78



CONCLUSIONES.....	79
RECOMENDACIONES.....	80
BIBLIOGRAFÍA.....	81
ANEXOS .....	84
Anexo 1: Autenticarse.....	84
Anexo 2: Gestionar Usuario.....	84
Anexo 3: Gestionar Evento Científico .....	84
Anexo 4: Gestionar Curso Optativo.....	85
Anexo 5: Gestionar Línea Investigativa.....	85
Anexo 6: Gestionar Incidencias.....	85
Anexo 7: Gestionar Evaluación de Proyecto.....	86
Anexo 8: Generar Nota de PP .....	86
Anexo 9: Mostrar Expediente Productivo .....	87
Anexo 10: Mostrar Evento Científico.....	88
Anexo 11: Mostrar Cursos optativo.....	88
Anexo 12: Mostrar Línea de Investigación .....	88
Anexo 13: Mostrar Incidencias .....	89
Anexo 14: Mostrar Evaluación de Proyecto .....	89
Anexo 15: Mostrar Nota de PP .....	89
Anexo 16: Clase de análisis del CU Registrar Cursos Optativo .....	90
Anexo 17: Clase de análisis del CU Registrar Evento Científico.....	90
Anexo 18: Clase de análisis del CU Generar Nota de PP .....	90
Anexo 19: Clase de análisis del CU Registrar Incidencias.....	91
Anexo 20: Clase de análisis del CU Registrar Líneas de Investigación.....	91
Anexo 21: Clase de análisis del CU Mostrar Curso Optativo .....	91
Anexo 22: Clase de análisis del CU Mostrar Evento Científico .....	92
Anexo 23: Clase de análisis del CU Mostrar Expediente Productivo.....	92
Anexo 24: Clase de análisis del CU Mostrar Incidencias .....	92
Anexo 25: Clase de análisis del CU Mostrar Lineas de Investigación .....	93
Anexo 26: Clase de análisis del CU Mostrar Nota de PP .....	93
Anexo 27: Clase de colaboración del análisis del CU Gestionar Usuario .....	93
Anexo 28: Clase de colaboración del análisis del CU Gestionar Evento Científico. ....	94
Anexo 31: Clase de colaboración del análisis del CU Gestionar Incidencias. ....	97
Anexo 32: Clase de colaboración del análisis del CU Generar Nota PP. ....	98
Anexo 33: Clase de colaboración del análisis del CU Mostrar Nota de PP.....	99
Anexo 34: Clase de colaboración del análisis del CU Mostrar Eventos Científicos.....	99
Anexo 35: Clase de colaboración del análisis del CU Mostrar Curso Optativo.....	100
Anexo 36: Clase de colaboración del análisis del CU Mostrar Líneas de Inv.....	100
Anexo 37: Clase de colaboración del análisis del CU Mostrar Incidencias.....	101
Anexo 38: Clase de Diseño del CU Mostrar Curso Optativo.....	101

Anexo 39: Clase de Diseño del CU Mostrar Evento Científico.....	102
Anexo 40: Clase de Diseño del CU Mostrar Incidencias.....	102
Anexo 41: Clase de Diseño del CU Mostrar Líneas de Inv.....	103
Anexo 42: Clase de Diseño del CU Mostrar Nota de PP.....	103
Anexo 43: Clase de Secuencia del Diseño del CU Autenticar. ....	104
Anexo 44: Clase de Diseño del CU Gestionar Usuario. ....	104
Anexo 45: Clase de Diseño del CU Gestionar Curso Optativo. ....	106
Anexo 46: Clase de Diseño del CU Gestionar Evento Científico. ....	106
Anexo 47: Clase de Diseño del CU Mostrar Expediente Productivo.....	107
Anexo 48: Clase de Diseño del CU Generar Nota de PP.....	107
GLOSARIO DE TERMINO.....	108

## INDICE DE FIGURAS

Imagen 1: Diagrama modelo de dominio .....	18
Imagen 2: Diagrama de paquetes. ....	24
Imagen 3: Paquete administración .....	25
Imagen 4: Paquete Registrar Datos .....	29
Imagen 5: Paquete salida de datos.....	41
Imagen 6: Diagrama de clases de análisis CU autenticar.....	51
Imagen 7: Diagrama de clases de análisis CU gestionar usuario.....	51
Imagen 8: Diagrama de colaboración CU autenticar. ....	52
Imagen 9: Diagrama de colaboración del análisis CU Mostrar Expediente Productivo. ....	53
Imagen 10: Diagrama de clases bases. ....	54
Imagen 11: Diagrama de secuencia para el constructor. ....	60
Imagen 12: Diagrama clases de diseño CU Autenticar .....	61
Imagen 13: Diagrama de clases de diseño CU mostrar expediente porductivo.....	62
Imagen 14: Diagrama de clases de diseño CU Gestionar Usuario .....	62
Imagen 15: Diagrama de clases de diseño de CU Gestionar Curso Optativo .....	63
Imagen 16: Diagrama de clases de diseño CU Gestionar Evento Científico .....	63
Imagen 17: Diagrama de clases de diseño CU Generar Nota PP .....	64
Imagen 18: Diagrama de clases de diseño CU Incidencias.....	64
Imagen 19: Diagrama de clases de diseño CU Líneas de Investigación. ....	65
Imagen 20: Diagrama Modelo de Datos.....	67
Imagen 21: Diagrama de paquetes de Componentes. ....	69
Imagen 22: Diagrama de Paquetes de Componentes Administración.....	70
Imagen 23: Diagrama de Paquete de Componentes Registrar Datos .....	70
Imagen 24: Diagrama de Paquete de Componentes Salidas de Datos .....	70
Imagen 25: Diagrama de Despliegue .....	71

INDICE DE TABLAS

Tabla 1: Descripción de los conceptos.....	17
Tabla 2: Actores del sistema a automatizar.....	24
Tabla 3: CU Gestionar Usuario.....	28
Tabla 4: CU Autenticar.....	29
Tabla 5: CU Gestionar evento científico.....	32
Tabla 6: CU Gestionar cursos optativos.....	34
Tabla 7: CU Gestionar líneas de investigación.....	36
Tabla 8: CU Gestionar incidencia.....	38
Tabla 9: CU Gestionar evaluación proyecto.....	40
Tabla 10: CU Generar la nota PP.....	41
Tabla 11: CU Mostrar expediente productivo.....	42
Tabla 12: CU Mostrar evento científico.....	43
Tabla 13: CU Mostrar curso optativo.....	44
Tabla 14: CU Mostrar líneas de investigación.....	45
Tabla 15: CU Mostrar incidencias.....	46
Tabla 16: CU Mostrar evaluación de proyecto.....	47
Tabla 17: CU Mostrar nota de PP.....	48
Tabla 18: Descripción de clases.....	59
Tabla 19: Factor de peso de los actores sin ajustar.....	72
Tabla 20: Factor de peso de los CU sin ajustar.....	73
Tabla 21: Factor de Complejidad Técnica.....	74
Tabla 22: Factor de Ambiente.....	75
Tabla 23: Esfuerzo del proyecto.....	76

### **INTRODUCCIÓN**

La Universidad de Ciencias Informáticas (UCI), se encuentra enfrascada cotidianamente en su digitalización, esto incluye la automatización de todas las esferas de forma progresiva. Desde el punto de vista de la producción, la UCI cuenta con un gran número de estudiantes vinculados a las actividades productivas e investigativas; esto se debe a la existencia de diversos proyectos, actividades de investigación y desarrollo como son: Jornadas Científicas, Mi Web por Cuba, Seminario Juvenil Martiano, Copas y Olimpíadas, UCIENCIA, Concursos de Corte investigativo/productivo, etc.

Actualmente el proceso de registrarle una determinada actividad productiva al estudiante de la UCI se hace de forma manual. Además a la hora de conformar un proyecto, la asignación de roles se realiza en muchas ocasiones de forma inexacta, incurriendo en errores cuando se va a seleccionar las personas que cumplan con lo requerido pues no se conocen las aptitudes y actitudes del estudiante. Toda esta información, tanto productiva como investigativa se va guardando en documentos duros, generando grandes volúmenes de información, lo que trae consigo dificultades a la hora de obtener alguna información, transmitirla ó realizar una búsqueda.

El Departamento Central de Práctica Profesional (PP) presentó una propuesta de un modelo de Expediente Productivo que recoja toda esta información. El objetivo de este expediente es crear competencias en los estudiantes de forma sistemática y creciente desde la actividad profesional, que desarrollen su capacidad de ejecutar tareas en roles específicos dentro de los equipos de desarrollo de software. Para lograr esto es de suma importancia controlar y describir la trayectoria científico-productiva del estudiante, lo cual tendrá seguimiento desde el Expediente Productivo. Esto es de vital importancia ya que de segundo a quinto año se hace necesario obtener la nota de PP por cada estudiante, la cual estará dada por el análisis final de la documentación actualizada del Expediente Productivo.

La Facultad 2 presenta el mismo problema de la universidad y siguiendo las indicaciones propuestas por el departamento central, esta lo ha ajustado según sus necesidades para cumplir con su perfil. La información se registra en un Excel lo que permite obtener la nota de

Práctica Profesional. Este trabajo es muy difícil ya que esta herramienta no brinda un trabajo individual con cada estudiante, imposibilita tener observaciones, detalles de los trabajos, evaluaciones de los mismos y resulta muy engorrosa la obtención de la nota de cada uno. Con los medios existentes hasta el momento la obtención de la información de los estudiantes y su nota en esta asignatura, se hace demasiado lento, así como su almacenamiento que debe ser organizado y clasificado atendiendo a diversos criterios. Además no existe una herramienta automatizada para la confección del Expediente Productivo e Investigativo del Estudiante, que faciliten el flujo de información, haciendo su transmisión más rápida y eficaz con el menor costo posible.

Luego de estudiar la situación problemática se definió la necesidad de resolver el siguiente **problema**: ¿Cómo erradicar los problemas de gestión del Expediente Productivo e Investigativo del Estudiante? Este problema se enmarca en el **objeto de estudio**: Los procesos asociados a la actividad productiva e investigativa del estudiante en la UCI. El **campo de acción** abarcado es: El proceso de gestión del Expediente Productivo e Investigativo del Estudiante en la Facultad 2. Se le da solución al problema con el **objetivo general**: Desarrollar un sistema informático para gestionar el Expediente Productivo e Investigativo del Estudiante.

Las **tareas** que se llevarán a cabo para darle cumplimiento al objetivo trazado son:

- ✓ Estudiar cómo se realiza el seguimiento, control y asignación de las actividades de cada estudiante.
- ✓ Analizar la forma en que la Facultad 2 obtiene la nota de la asignatura de Práctica Profesional de cada estudiante.
- ✓ Estudiar las posibles herramientas de Software Libre para el desarrollo de este sistema.
- ✓ Desarrollar y caracterizar de los primeros flujos de trabajo de RUP.
- ✓ Analizar y diseñar el sistema cumpliendo con la aprobación del cliente.
- ✓ Diseñar e implementar una base de datos capaz de almacenar de manera organizada la información referente al Expediente Productivo e Investigativo del Estudiante.
- ✓ Implementar una aplicación Web que automatice los procesos necesarios para una correcta y eficiente gestión del Expediente Productivo e Investigativo del Estudiante.

- ✓ Realizar el estudio de factibilidad de la propuesta de solución.

El trabajo está formado por cinco capítulos donde se abarcan los siguientes temas:

Capítulo 1: Se hace un estudio de las principales herramientas para la implementación de la aplicación, así como el lenguaje de programación y la metodología de desarrollo de software a utilizar.

Capítulo 2: En este capítulo se representa el Modelo de Dominio. Se exponen los requisitos del sistema, tanto funcionales como no funcionales y se realiza el diagrama de casos de uso con sus respectivas descripciones.

Capítulo 3: Se realiza el análisis y diseño del sistema. Se obtiene la modelación de las clases utilizando las extensiones del UML y se presentan a través de diagramas de clases Web. Se analiza la arquitectura a utilizar y la implementación del patrón modelo vista controlador.

Capítulo 4: En este capítulo se presentarán el modelo de implementación mediante los diagramas de componentes por paquetes y el de despliegue que resultaron del diseño realizado de cada uno de los casos de uso del sistema planteado.

Capítulo 5: En este capítulo se describe el estudio de factibilidad realizado correspondiente al sistema propuesto, teniendo en cuenta el costo estimado y los beneficios que reportará al ser implantado.

### **CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA**

#### **Introducción**

Este capítulo aborda un estudio de las tendencias y tecnologías actuales sobre las que se apoyará la propuesta de solución. Se fundamentan las herramientas y lenguaje a emplearse para el análisis y modelado de un sistema acorde con los objetivos, teniendo en cuenta el criterio del cliente y la línea que persigue la Facultad de desarrollar el software libre.

#### **1.1 Procesos asociados a la actividad Productiva e Investigativa**

A los estudiantes de la Facultad 2, se les hace un seguimiento de toda la actividad productiva e investigativa que realizan a lo largo de su carrera. Se debe tener un control de los estudiantes vinculados o no a proyectos, de pertenecer a un proyecto es necesario saber su evaluación, criterio del jefe de proyecto acerca de su trabajo, si existiera un llamado de atención al estudiante esta se registra. Tanto para el estudiante que pertenece a un proyecto como para el que no pertenece, se recogen sus notas de cursos optativos, la participación en eventos científicos, de esto y mucho más se necesita tener observaciones, detalles de los trabajos y evaluaciones de los mismos. Al final de todo el curso escolar este registro va a hacer de valiosa importancia a la hora de gestionar la nota de Práctica Profesional, la cual estará dada por el análisis final de la documentación actualizada en el Expediente Productivo e Investigativo del Estudiante.

#### **1.2 Sistemas existentes**

En la actualidad no existe un sistema que resuelva la problemática anteriormente descrita. El Departamento Central de Práctica Profesional de la UCI propuso un modelo de Expediente Productivo e Investigativo que controle y describa la trayectoria científico-productiva del estudiante.

#### **1.3 Proceso de desarrollo de software utilizado**

Para controlar y planificar la propuesta que presenta este trabajo, se decidió utilizar como proceso de desarrollo de software el Proceso Unificado de Modelado (RUP), por sus características y las facilidades que aporta a todo el proceso.



### 1.3.1 Metodología Rational Unified Process (RUP)

RUP es una de las metodologías más generales de las que existen en la actualidad, pues está pensada para adaptarse a cualquier proyecto, no sólo de software. Se basa en casos de uso para describir lo que se espera del software y está muy orientado a la arquitectura del sistema, documentándose lo mejor posible, basado en UML (Lenguaje Unificado de Modelado) como herramienta principal.

El Proceso Racional Unificado es un proceso de desarrollo de software y constituye una metodología estándar para el análisis, diseño, implementación y documentación de sistemas orientados a objetos. Pone a disposición la forma disciplinada de asignar tareas y responsabilidades, además de implementar las mejores prácticas en la Ingeniería de Software como son: desarrollo iterativo e incremental, control de cambios, verificación de la calidad, administración de requisitos, entre otros. [1]

#### **Características de RUP**

##### **✓ Dirigido por Casos de Uso**

El proceso de desarrollo lleva un hilo, avanza a través de una serie de flujos de trabajo que parten de los casos de uso. Los casos de uso se especifican, se diseñan. Los casos de uso guían la arquitectura del sistema. [1]

##### **✓ Centrado en la arquitectura**

La arquitectura surge de las necesidades de la empresa. Incluye los aspectos más significativos del sistema y se ve influida por muchos otros factores como la plataforma en la que tiene que funcionar el software (arquitectura hardware, sistema operativo, sistema de gestión de base de datos), las consideraciones de implantación, sistemas heredados y requisitos no funcionales. [1]

##### **✓ Iterativo e incremental**

Es práctico dividir el trabajo en partes más pequeñas o mini proyectos. Cada mini proyecto es una iteración que resulta en un incremento. En cada iteración, los desarrolladores identifican y especifican los casos de uso relevantes, crean un diseño utilizando la arquitectura

seleccionada como guía, implementan el diseño mediante componentes, y verifican que los componentes satisfacen los casos de uso. [1]

### 1.4 Unified Modeling Language (UML)

El Lenguaje Unificado de Modelado (UML), por sus siglas en inglés, (Unified Modeling Language) es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad. Es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema de software. [2]

Ofrece un estándar para describir un modelo del sistema, 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. [2]

UML es un método formal de modelado que aporta las siguientes ventajas:

- ✓ Mayor rigor en la especificación.
- ✓ Permite realizar una verificación y validación del modelo realizado.
- ✓ Se pueden automatizar determinados procesos y permite generar código a partir de los modelos y a la inversa.

Las principales funciones de UML son:

- ✓ Visualizar: Permite expresar de una forma gráfica un sistema de forma que otro lo puede entender.
- ✓ Especificar: Permite especificar cuáles son las características de un sistema antes de su construcción.
- ✓ Construir: A partir de los modelos especificados se pueden construir los sistemas diseñados.
- ✓ Documentar: Los propios elementos gráficos sirven como documentación del sistema desarrollado que pueden servir para su futura revisión.

### 1.5 Tecnologías actuales a utilizar

#### 1.5.1 Modelo Cliente / Servidor

La modalidad o arquitectura Cliente/Servidor es aquella en la que confluyen una serie de aplicaciones basadas en dos categorías que cumplen funciones diferentes (una requiere servicios y la otra los brinda) pueden realizar tanto actividades en forma conjunta como independientemente. Esas dos categorías son justamente cliente y servidor. [3]

En el caso del **cliente**, es aquel que requiere un servicio del servidor. En esta categoría se realizan funciones de software basándose en el hardware pero en caso de no tener la capacidad de procesar los datos necesarios, recurre al servidor y espera a que este le brinde los servicios solicitados. [3]

Por el contrario, el **servidor** es la máquina desde la que se suministran servicios y que está a la espera del requerimiento del cliente. Una vez hecho, busca la información solicitada y le envía la respuesta al cliente; incluso puede enviar varios servicios a la vez, lo que es posible porque ellos están conectados mediante redes LAN o WAN. [3]

### Algunas Ventajas

- ✓ Aumento de la productividad

Los usuarios pueden utilizar herramientas con las que estén familiarizados, 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. [4]

- ✓ Menores costes de operación

Permiten un mejor aprovechamiento de los sistemas existentes, protegiendo la inversión. 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. [4]

- ✓ 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. [4]

### 1.5.2 Software Libre

Software libre (en inglés: free software) es el software que, una vez obtenido, puede ser usado, copiado, estudiado, modificado y redistribuido libremente. El software libre suele estar disponible gratuitamente en Internet, o a precio del coste de la distribución a través de otros medios; sin embargo no es obligatorio que sea así y, aunque conserve su carácter de libre, puede ser vendido comercialmente. Análogamente, el software gratis o gratuito (denominado usualmente freeware) incluye en algunas ocasiones el código fuente; sin embargo, este tipo de software no es libre en el mismo sentido que el software libre, al menos que se garanticen los derechos de modificación y redistribución de dichas versiones modificadas del programa. [5]

De acuerdo con tal definición, el software es "libre" si garantiza las siguientes libertades: [5]

- ✓ "libertad 0", ejecutar el programa con cualquier propósito (privado, educativo, público, comercial, etc.)
- ✓ "libertad 1", estudiar y modificar el programa (para lo cual es necesario poder acceder al código fuente)
- ✓ "libertad 2", copiar el programa de manera que se pueda ayudar al vecino o a cualquiera.
- ✓ "libertad 3", mejorar el programa, y hacer públicas las mejoras, de forma que se beneficie toda la comunidad.

Es importante señalar que las libertades 1 y 3 obligan a que se tenga acceso al código fuente. La "libertad 2" hace referencia a la libertad de modificar y redistribuir el software libremente licenciado bajo algún tipo de licencia de software libre que beneficie a la comunidad.

### 1.5.3 Servidor Web Apache

El servidor Web Apache es uno de los mayores triunfos del software libre. Entre sus características fundamentales podemos destacar: [6]

- ✓ Es flexible, rápido y eficiente.
- ✓ Continuamente actualizado y evoluciona a mayor velocidad.
- ✓ Multiplataforma.
- ✓ Se desarrolla de forma abierta.
- ✓ Modular: Puede ser adaptado a diferentes entornos y necesidades, con los diferentes módulos de apoyo que proporciona, y con la API de programación de módulos, para el desarrollo de módulos específicos.
- ✓ Incentiva la realimentación de los usuarios, obteniendo nuevas ideas, informes de fallos y parches para la solución de los mismos.

### 1.5.4 Ajax

AJAX es la unión de varias tecnologías que unidas pueden lograr cosas realmente impresionantes. AJAX, es el acrónimo para **Asynchronous JavaScript + XML** y el concepto es: Cargar y renderizar una página, luego mantenerse en esa página mientras scripts y rutinas van al servidor buscando, en background, los datos que son usados para actualizar la página solo re-renderizando la página y mostrando u ocultando porciones de la misma.

En vez de cargar una pagina Web, al inicio de la sesión, el navegador carga al motor AJAX (escrito en JavaScript y usualmente “sacado” en un frame oculto). Este motor es el responsable por renderizar la interfaz que el usuario ve y por comunicarse con el servidor en nombre del usuario. El motor AJAX permite que la interacción del usuario con la aplicación suceda asincrónicamente (independientemente de la comunicación con el servidor). Así el usuario nunca estará mirando una ventana en blanco del navegador y un icono de reloj de arena esperando a que el servidor haga algo. [7]

AJAX incorpora:

- ✓ Presentación basada en estándares usando XHTML y CSS.
- ✓ Exhibición e interacción dinámicas usando el Document Object Model.
- ✓ Intercambio y manipulación de datos usando XML y XSLT.

- ✓ Recuperación de datos asincrónica usando XMLHttpRequest.

### 1.5.5 Lenguajes de programación para la Web

Existen dos grupos de lenguajes de programación para la Web clasificándose en dependencia de dónde se implementan siguiendo la arquitectura Cliente/Servidor. De esta forma un grupo es llamado lenguajes de lado del Cliente y el otro grupo lenguajes del lado del Servidor. En el primer grupo entre los más usados se encuentran: **Java Script, Visual Basic Script, Java**. En el segundo grupo se destacan: **PERL, ASP, JSP, PHP**.

#### 1.5.5.1 Del lado del cliente

##### JavaScript

JavaScript es utilizado para crear pequeños programas encargados de realizar acciones dentro de una página Web. Con JavaScript se puede crear efectos especiales en las páginas y definir interactividades con el usuario. Entre las acciones típicas que se pueden realizar en JavaScript se tienen dos vertientes. Por un lado los efectos especiales sobre páginas Web, para crear contenidos dinámicos y elementos de la página que tengan movimiento, cambien de color o cualquier otro dinamismo. Por el otro, JavaScript nos permite ejecutar instrucciones como respuesta a las acciones del usuario, con lo que podemos crear páginas interactivas con programas como calculadoras, agendas, o tablas de cálculo. [8]

#### 1.5.5.2 Del lado del servidor

##### PHP

PHP (acrónimo de PHP: Hypertext Preprocessor), es un lenguaje interpretado de alto nivel embebido en páginas HTML y ejecutado en el servidor. La meta del lenguaje es permitir rápidamente a los desarrolladores la generación dinámica de páginas. Con PHP se puede hacer cualquier cosa que podemos realizar con un script CGI, como el procesamiento de información en formularios, foros de discusión, manipulación de cookies y páginas dinámicas. También ofrece la integración con varias bibliotecas externas, que permiten que el desarrollador haga casi cualquier cosa, desde generar documentos en pdf, hasta analizar código XML. [9]

PHP ofrece una solución simple y universal para las paginaciones dinámicas de la Web de fácil programación. Su diseño elegante lo hace perceptiblemente más fácil de mantener y ponerse al día en comparación con el código de otros lenguajes. Debido a su amplia distribución PHP está perfectamente soportado por una gran comunidad de desarrolladores. Como producto de código abierto, PHP goza de la ayuda de un gran grupo de programadores, permitiendo que los fallos de funcionamiento se encuentren y se reparan rápidamente. El código se pone al día continuamente con mejoras y extensiones de lenguaje para ampliar las capacidades de PHP. [9]

### 1.5.6 Framework a Utilizar

#### CodeIgniter

CodeIgniter es un framework, utilizado por una gran comunidad de usuarios. Construido para codificadores PHP que necesitan una herramienta de desarrollo fácil para crear aplicaciones web simples y elegantes. Entre sus características podemos encontrar su compatibilidad con PHP 4 y PHP 5, incorpora el patrón MVC, soporte para múltiples bases de datos, plantillas, validaciones, no requiere instalación, podemos encontrar una librería con un gran número de clases. Podemos descargarlo y descomprimir este paquete dentro un directorio en nuestro servidor web. [10]

#### CodeIgniter es adecuado si:

- ✓ Se quiere un framework con una pequeña huella.
- ✓ Se quiere una amplia compatibilidad con cuentas de hospedaje estándar que corra una variedad de versiones y configuraciones de PHP.
- ✓ Se quiere un framework que exige casi cero configuraciones.
- ✓ Se quiere un framework que no requiere que el cliente use la línea de comandos.
- ✓ Se quiere un framework que no le pida que se adhieran a las restrictivas normas de codificación.
- ✓ Se quiere evitar complejidad, favoreciendo soluciones simples.
- ✓ Se quiere claridad, la documentación completa.

### 1.5.7 Sistema Gestor de Base de Datos

Un sistema gestor de base de datos se define como el conjunto de programas que administran y gestionan la información contenida en una base de datos. Ayuda a realizar las siguientes acciones: [11]

- ✓ Definición de los datos.
- ✓ Mantenimiento de la integridad de los datos dentro de la base de datos.
- ✓ Control de la seguridad y privacidad de los datos.
- ✓ Manipulación de los datos.

#### 1.5.7.1 PostgreSQL

PostgreSQL es un Sistema de Gestión de Bases de Datos Objeto-Relacionales (ORDBMS) que ha sido desarrollado de varias formas desde 1977. PostgreSQL proporciona un gran número de características que normalmente sólo se encontraban en las bases de datos comerciales de alto calibre.

Soporta transacciones y desde la versión 7.0 claves ajenas (con comprobaciones de integridad referencial). Tiene mejor soporte para triggers y procedimientos en el servidor. Además, tiene ciertas características orientadas a objetos. [12]

## 1.6 Herramientas utilizadas

### 1.6.1 Macromedia Dreamweaver

Para el diseño del sistema, se utilizará como herramienta de desarrollo de Páginas Web a Macromedia Dreamweaver 8.0.

### 1.6.2 Dreamweaver 8.0

Es una herramienta de desarrollo de páginas Web que permite a sus usuarios diseñar, crear y mantener de forma eficaz sitios y aplicaciones Web basadas en normas. Con Dreamweaver 8 se usa un editor de diseño y código de máxima calidad en una sola herramienta. Cuenta con un sistema de ayuda que hace más fácil su uso y guía según se adquieran conocimientos y a medida que las tecnologías Web van evolucionando, facilitando una adopción sencilla y rápida de las nuevas tecnologías y metodologías. [13]



### **1.6.3 Adobe Photoshop**

Adobe Photoshop CS para el tratamiento de los gráficos. Es una herramienta muy poderosa para crear cualquier tipo de gráficos, su integración con Adobe ImageReady hacen que crear complicados gráficos para la Web sea una tarea muy fácil. [14]

### **1.6.4 Zend Studio**

Zend Studio es uno de los ambientes de desarrollo integrado o Integrated Development Environment (IDE) disponible para desarrolladores profesionales que agrupa todos los componentes de desarrollo necesarios para ciclo de desarrollo de aplicaciones PHP. A través de un comprensivo conjunto de herramientas de edición, depurado, análisis, optimización y bases de datos, Zend Studio acelera los ciclos de desarrollo y simplifica los proyectos complejos. [15]

### **1.6.5 Rational Rose**

Existen herramientas CASE de trabajo visual que permiten realizar el modelado del desarrollo de los proyectos, en la actualidad una de las más utilizada es Rational Rose y es la que se utiliza en la modelación de este proyecto. Rational Rose 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. 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. [16]

#### **1.6.5.1 Rational Rose Enterprise**

Rational Rose sirve de soporte para el modelo del negocio, ayudando a entender el negocio alrededor del sistema. Apoya el análisis de los sistemas permitiendo diseñar casos de uso y utilizar diagramas de casos de uso para demostrar la funcionalidad del sistema. Permite diseñar diagramas de interacción demostrando como los objetos trabajan juntos para proporcionar la funcionalidad necesaria. Los diagramas de clase se pueden crear para mostrar las clases en un sistema y como se relacionan la una con la otra. Los diagramas de componentes se pueden desarrollar para ilustrar como existe una traza entre las clases y la

implementación de los componentes. Finalmente, el diagrama de despliegue que se puede desarrollar para el diseño de red del sistema.

Se decidió que se utilizaría el Rational Rose Enterprise Edition 2003, para sustentar la documentación, como modelador visual de la notación UML (Unified Modeling Language) para la confección de los diagramas que se ilustran en este trabajo.

### **1.6.6 PgAdmin**

Es desarrollado por una comunidad de los expertos de PostgreSQL de varias partes del mundo y está disponible en más de una docena de idiomas. Se trata de una herramienta para la administración de bases de datos PostgreSQL. Su uso se puede extender hacia las plataformas de Linux, de FreeBSD, de Solaris, del Mac OSX y de Windows.[17]

### **1.6.7 Case Studio**

Herramienta profesional para el diseño de base de datos. Diseñado para crear visualmente y mantener Diagramas Entidad Relación (ERD) para numerosas bases de datos. Es una elección ideal para las empresas y desarrolladores. Herramienta de gestión utilizando bien organizado ERD, a través de los cuales se obtiene complejo SQL-scripts para la creación física de las tablas de distintas bases de datos (Oracle, DB2 UDB, MS SQL, Interbase, MySQL, PostgreSQL, Sybase, Ingres, Informix y más). Los flujos de datos entre las tablas también pueden ser fácilmente descritos por la creación apropiada de diagramas de flujo de datos. Al implementar ERD se proporciona a cada base de datos de opciones tales como integridad referencial, las limitaciones, dominios, disparadores, etc. [18]

## **Conclusiones**

En este capítulo se ha desarrollado el estudio de algunas de las tecnologías actuales que serán empleadas durante el desarrollo del software, mencionando ciertas características y ventajas que las mismas poseen, la decisión de utilizarlas fue tomada siguiendo fundamentalmente la política de uso de herramientas con soporte multiplataforma y licencias de utilización libre, las cuales fueron escogidas por el cliente. Fueron utilizadas además algunas herramientas de software propietario para apoyar la realización del trabajo.

Las herramientas a utilizar son las siguientes:

- ✓ Gestor de Base Datos: PostgreSQL 8.2
- ✓ Lenguaje de programación del lado del servidor: PHP v5.2.3
- ✓ Lenguaje de programación del lado del cliente: JavaScript y Ajax.
- ✓ Servidor Web: Apache versión 2.2.4
- ✓ Para el desarrollo de la documentación: RUP con notación UML.
- ✓ Herramienta CASE de Modelado UML: Rational Rose.
- ✓ Para apoyar el proceso de implementación del software se utilizarán herramientas de desarrollo como: Dreamweaver v8.0 es, ZendStudio.
- ✓ Para apoyar el proceso de diseño: Adobe PhotoShop v7.0

### CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

#### Introducción

En este capítulo se representa el Modelo de Dominio. Se resaltan los requisitos funcionales y no funcionales que debe tener el sistema que se propone, lo que permite hacer una concepción general de los objetivos del sistema, e identificar mediante un Diagrama de Caso de Uso las relaciones de los actores que interactúan con el sistema, y las secuencias de acciones con las que interactúan.

#### 2.1 Modelo del Dominio

Para capturar correctamente los requisitos y poder construir un sistema correcto se necesita tener un firme conocimiento del funcionamiento del objeto de estudio. Debido a que no están bien estructurados los procesos del negocio estudiado, se arriba a la conclusión de construir un modelo conceptual.

Este permite mostrar visualmente al usuario los principales conceptos que se manejan en el dominio del sistema en desarrollo. Además ayuda a los usuarios, clientes, desarrolladores e interesados a utilizar un vocabulario común para poder comprender el contexto en que se desarrolla el sistema. El objetivo del modelo conceptual es comprender y describir las clases más importantes dentro del contexto del sistema, en otras palabras el modelado conceptual deberá contribuir a una comprensión del problema que el sistema resuelve en relación a su contexto. También se hará uso de un glosario de términos para identificar todos los conceptos que se utilizarán en el diagrama.

Conceptos	Justificación
Vicedecano de Producción	Es el principal beneficiado con todo este proceso del Expediente Productivo e Investigativo del Estudiante, pues es el que se encarga de solicitar y revisar la información contenida en este expediente.
Secretaria Docente	Persona que desempeñará el rol de gestionar

	el Expediente Productivo y registrar todos los datos del estudiante.
Profesor	Es la persona que dirige todo el proceso de los eventos científicos e imparte algunos de los cursos optativos recibidos por los estudiantes.
Estudiante	Es la persona a la cual se le describe todas las actividades que realiza.
Evento Científico	Hace referencia a los eventos realizados por el estudiante como son: Jornada Científica, Mi Web por Cuba, Seminario Juvenil Martiano, Copas y Olimpiadas, UCIENCIA, Concursos de Corte investigativo/productivo.
Proyecto	Es la entidad a la cual puede que pertenezca un estudiante o no.
Jefe de Proyecto	Es la persona que dirige un proyecto.
Curo Optativo	Hace referencia a los cursos optativos impartidos por los estudiantes.
Línea de investigación	Da a conocer las líneas de investigación que tenga el estudiante.

Tabla 1: Descripción de los conceptos.

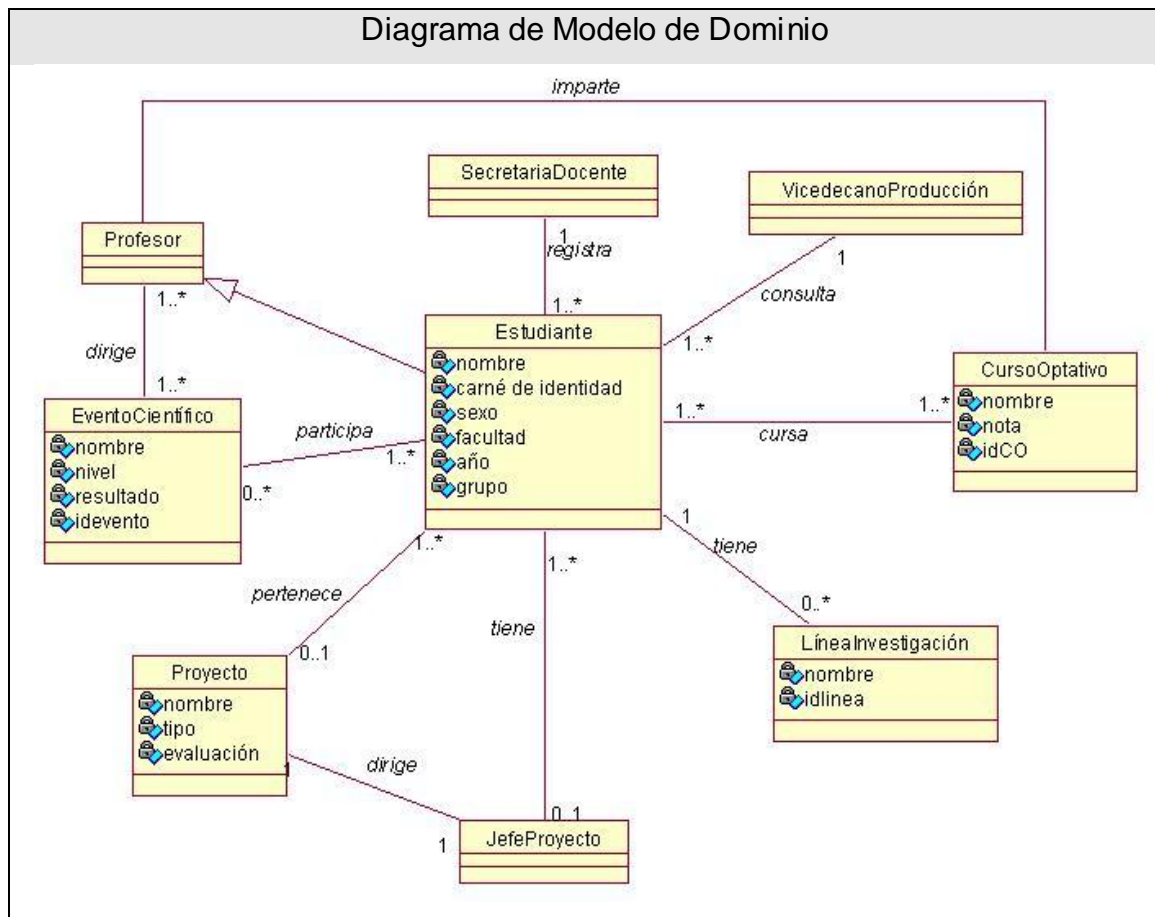


Imagen 1: Diagrama modelo de dominio

### 2.2 Reglas del negocio

Las reglas de negocio describen políticas que deben cumplirse o condiciones que deben satisfacerse, por lo que regulan algún aspecto del negocio, esta son:

1. La secretaria debe registrar el curso docente con el cual se va a trabajar en todo un período.
2. El estudiante nunca va a ser un trabajador del sistema.
3. La secretaria docente es la única que puede modificar los datos registrados en el expediente.
4. Solo se le gestionará la nota de PP al estudiante a partir de segundo año.
5. Un usuario solo va a tener un rol en del sistema.

### 2.3 Requerimientos funcionales

Los requerimientos funcionales son aquellos requisitos que, desde el punto de vista de las necesidades del usuario debe cumplir el sistema y que están fuertemente ligados a las opciones del programa.

Para cumplir con los objetivos propuestos se prevé que el sistema tenga las siguientes funcionalidades:

#### **R1. Administración**

##### R1.1 Insertar Usuario

R1.1.1 Los datos a Insertar son:

R1.1.1.1 Usuario

##### R1.2 Eliminar Usuario

R1.2.1 Listar los usuarios existentes en el sistema

R1.2.1.1 Seleccionar el usuario y eliminar

##### R1.3 Modificar Usuario

R1.3.1 Listar los usuarios existentes en el sistema

R1.3.1.1 Seleccionar el usuario y modificar

##### R1.4 Autenticar

R1.4.1 Los datos a autenticar son:

R1.4.1.1 Usuario

R1.4.1.2 Contraseña

#### **R2. Registrar Datos**

##### R2.1 Gestionar Evento Científico

R2.1.1 Realizar búsqueda del estudiante

R2.1.2 Agregar Evento Científico

R2.1.2.1 Los datos a agregar son:

R2.1.2.1.1 Nombre

R2.1.2.1.2 Nivel

R2.1.2.1.3 Resultado

R2.1.3 Modificar Evento Científico

R2.1.3.1 Los datos a modificar son:

R2.1.3.1.1 Nombre

R2.1.3.1.2 Nivel

R2.1.3.1.3 Resultado

R2.1.4 Eliminar Evento Científico

R2.1.4.1 Listar los Eventos Científicos registrados

R2.1.4.1.1 Seleccionar el evento a eliminar

R2.2 Gestionar Cursos Optativos Impartido

R2.2.1 Realizar búsqueda del estudiante

R2.2.2 Agregar Curso Optativo Impartido

R2.2.2.1 Los datos a agregar son:

R2.2.2.1.1 Nombre del curso

R2.2.3 Modificar Curso Optativo Impartido

R2.2.3.1 Los datos a modificar son:

R2.2.3.1.1 Nombre del curso

R2.2.4 Eliminar Curso Optativo Impartido

R2.2.4.1 Listar los Cursos Optativos Impartidos

R2.2.4.1.1 Seleccionar el curso a eliminar

R2.3 Gestionar Líneas de Investigación

R2.3.1 Registrar Líneas de Investigación

R2.3.1.1 Realizar búsqueda del estudiante

R2.3.1.2 Agregar Líneas de Investigación

R2.3.2 Modificar Líneas de Investigación

R2.3.3 Eliminar Líneas de Investigación

R2.3.3.1 Listar Líneas de Investigación

R2.3.3.2 Seleccionar Líneas de Investigación a eliminar

R2.4 Gestionar Incidencias

R2.4.1 Registrar Incidencias

R2.4.1.1 Realizar búsqueda del estudiante

R2.4.1.2 Agregar la Incidencia

R2.4.2 Modificar Incidencias

R2.4.3 Eliminar Incidencias



R2.4.3.1 Listar las incidencias

R2.4.3.2 Seleccionar la incidencia a eliminar

R2.5 Gestionar Evaluaciones de Proyecto

R2.5.1 Registrar Evaluaciones de Proyecto

R2.5.1.1 Realizar búsqueda del estudiante

R2.5.1.2 Agregar la Evaluaciones de Proyecto

R2.5.2 Modificar Evaluaciones de Proyecto

R2.5.3 Eliminar Evaluaciones de Proyecto

R2.5.3.1 Listar las Evaluaciones de Proyecto

R2.5.3.2 Seleccionar Evaluaciones de Proyecto a eliminar

R2.6 Generar nota de Práctica Profesional

### **R3. Mostrar Datos**

R3.1 Mostrar Expediente Productivo

R3.1.1 Realizar búsqueda del estudiante

R3.1.2 Seleccionar el estudiante

R3.1.3 Imprimir Expediente Productivo

R3.2 Mostrar Eventos Científicos

R3.2.1 Realizar búsqueda del estudiante

R3.2.2 Seleccionar el estudiante

R3.3 Mostrar Cursos Optativos Impartidos

R3.3.1 Realizar búsqueda del estudiante

R3.3.2 Seleccionar el estudiante

R3.4 Mostrar Líneas de Investigación

R3.4.1 Realizar búsqueda del estudiante

R3.4.2 Seleccionar el estudiante

R3.5 Mostrar Incidencias

R3.5.1 Realizar búsqueda del estudiante

R3.5.2 Seleccionar el estudiante

R3.6 Mostrar Evaluaciones de Proyecto

R3.6.1 Realizar búsqueda del estudiante

R3.6.2 Seleccionar el estudiante

R3.7 Mostrar nota de Práctica Profesional

R3.7.1 Realizar búsqueda del estudiante

R3.7.2 Seleccionar el estudiante

### 2.4 Requerimientos No funcionales

Los requerimientos no funcionales son características que describen alguna forma o restricción para la realización de algún requerimiento (funcionalidad) o conjunto de ellas, e inclusive todos los requerimientos. Se consideran los atributos del sistema propiedades que debe tener el producto.

A continuación se muestran los requerimientos no funcionales:

#### Requerimiento de software

Para el servidor:

- El S.O. Windows XP o superior o cualquiera de las distribuciones Linux, en sus versiones de S.O. servidores.
- Un servidor Apache v2.2.4 con módulo PHP 5.2.3 disponible.
- Un servidor de bases de datos PostgreSQL 8.2

Para el cliente:

- S.O. Windows XP o superior o cualquiera de las distribuciones Linux.
- Navegadores Mozilla Firefox 2.0 o Superior.

#### Requerimiento de hardware

Para el servidor:

- En el caso del Servidor de Aplicaciones y Base de Datos se requerirá combo de PC (mouse, teclado, CPU y monitor).
- Tarjeta de red.
- Memoria RAM de un mínimo de 512 MB.
- 80 GB de espacio en disco.

Para el cliente:

- Un combo de PC (mouse, teclado, CPU y monitor).
- Tarjeta de red.

- Memoria RAM de un mínimo de 128 MB.
- Impresora láser o de punto.

### **Apariencia o interfaz externa**

- Resolución de 1024x768 píxeles.

### **Usabilidad**

- El sistema podrá ser usado por cualquier tipo de personas que posean conocimientos básicos en el manejo de la computadora y el ambiente Web en sentido general.

### **Rendimiento**

- Se debe garantizar que el tiempo de respuesta del sistema ante las solicitudes de los usuarios para cada acción a realizar por estos no exceda los tres segundos. El sistema deberá de ser lo más estable y confiable posible.

### **Portabilidad**

- El sistema será multiplataforma (Linux o Windows).

### **Requerimientos de Seguridad**

- El usuario debe autenticarse antes de entrar al sistema.
- Garantizar el acceso controlado al sistema. Este debe influir sobre cómo se presentan las interfaces para cada usuario dependiendo del nivel de acceso que tengan.

### **Restricciones para el diseño e implementación**

- Utilizar los estándares establecidos de codificación, diseño, entre otros.
- Emplear como servidores web y de bases de datos, Apache y PostgreSQL respectivamente.
- Utilizar como lenguaje del lado del servidor al PHP v5.1.2 y del lado del cliente Java Script y Ajax.

## **2.5 Actores del sistema a automatizar**

<b>Nombre del actor</b>	<b>Descripción</b>
Administrador	Es la persona encargado de administrar los usuarios del sistema y asignarles el rol correspondiente, podrá

	agregar ó eliminar un usuario. Encargado de darle soporte y administrar el sistema.
Vicedecano de Producción	Va a ser el cliente del sistema, es la persona que se va a beneficiar de toda la información contenida en este expediente, teniendo la posibilidad de imprimirlo.
Secretaria Docente	Es la persona encargada de gestionar los expedientes de los estudiantes y de registrar toda la información relacionada con la actividad productiva de los mismos.
Jefe de proyecto	Es la persona encargada de registrar las evaluaciones y resultados de los estudiantes vinculados a proyecto.
Estudiante	Es la persona beneficiada con toda la información contenida en el expediente, con la única posibilidad de ver su información.

Tabla 2: Actores del sistema a automatizar

## 2.6 Paquetes y sus relaciones

La mejor forma para las personas que necesitan consultar un sistema y entender el mismo es dividir este en pequeñas unidades. Para satisfacer los objetivos de este trabajo el sistema que se propone debe estar dividido en tres paquetes: Administración, Registrar Datos y Salida de Datos. A continuación se representa el diagrama de paquete y sus relaciones.

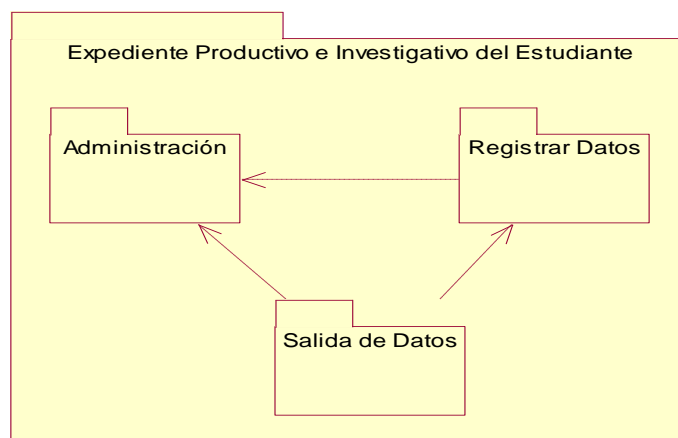


Imagen 2: Diagrama de paquetes.

## 2.7 Diagramas Casos de Usos del Sistema

Los Casos de Uso son “fragmentos” de funcionalidad que el sistema ofrece para aportar un resultado de valor para sus actores. De manera más precisa, un Caso de Uso especifica una secuencia de acciones que el sistema puede llevar a cabo interactuando con sus actores, incluyendo alternativas dentro de la secuencia.

### 2.7.1 Descripción de los casos de uso

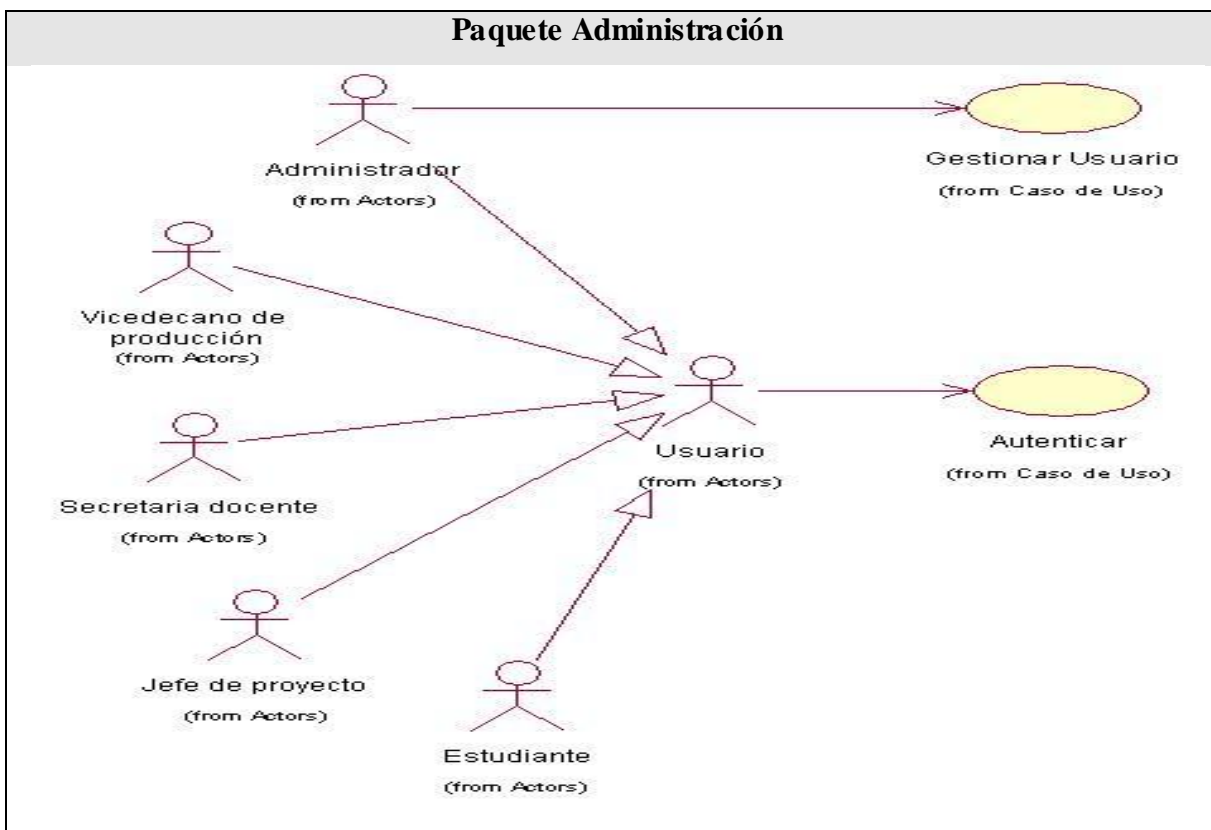


Imagen 3: Paquete administración

### Paquete de Administración

Caso de uso “Gestionar Usuarios”.

<b>Nombre del Caso de Uso</b>	<b>Gestionar Usuarios</b>
<b>Actores</b>	Administrador

<b>Propósito</b>	Registrar los datos de cada Usuario, modificar y eliminar un usuario del sistema.
<b>Resumen</b>	El caso de uso se inicia cuando el Administrador accede al sistema y solicita Gestionar Usuarios, seguidamente se visualizan los usuarios existentes en el sistema, a partir de aquí el sistema brinda la posibilidad de realizar las operaciones de agregar, modificar o eliminar uno de ellos. Termina cuando el administrador realiza una de las operaciones.
<b>Referencias</b>	<b>R1.1, R1.2, R1.3, R1.4</b>
<b>Precondiciones</b>	Cuando se desea agregar un usuario debe existir la instancia del rol que va a ejercer.
<b>Poscondiciones</b>	Para los siguientes procesos: Insertar: se crea una instancia de la clase usuario. Modificar: se modifica la instancia seleccionada de cada usuario. Eliminar: se elimina la instancia seleccionada de la clase usuario.
<b>Requerimientos especiales</b>	
<b>Curso normal de los eventos</b>	
<b>Acciones del actor</b>	<b>Respuestas del Sistema</b>
1. El Administrador decide administrar usuarios y selecciona la opción.	1.1 El sistema muestra una interfaz que posibilita gestionar un usuario.
2. El Administrador selecciona una de las operaciones a realizar.	2.1. El sistema ejecuta una de las siguientes acciones: a) Si decide agregar un usuario, ir a la sección "Agregar Usuario". b) Si decide modificar un usuario, ir a la sección

	<p>“Modificar Usuario”.</p> <p>c) Si decide eliminar un usuario, ir a la sección “Eliminar Usuario”.</p>
<b>2.1a) Sección “Agregar Usuario”</b>	
<p>2.1.a El Administrador introduce los datos del usuario (user, pass, rol)</p>	<p>2.1.a .1 El sistema verifica la validés de los datos introducidos.</p> <p>2.1.a .2 El sistema verifica la existencia del usuario.</p> <p>2.1.a .3 Si no existe se registra en el sistema el usuario. El sistema muestra un mensaje comunicando que el usuario se agregó satisfactoriamente y finaliza el caso de uso.</p>
<b>Curso alternativo</b>	
	<p>2.1.a .2.1 Si existe el usuario se emite un mensaje de error informando que ya existe el usuario a introducir.</p>
<b>2.1b) Sección “Modificar Usuario”</b>	
<p>2.1.b El Administrador selecciona dentro del listado de usuarios al que desea modificar.</p>	<p>2.1.b.1 El sistema posibilita modificar el usuario.</p> <p>2.1.b .2 El sistema verifica que los datos introducidos sean válidos y si esto es así se modifica el usuario.</p>
<b>Curso alternativo</b>	
	<p>2.1.b.2.1 Si los datos no son válidos se emite un mensaje de error para que se llenen los campos obligatorios o se introduzcan correctamente los datos.</p>
<b>2.1c) Sección “Eliminar Usuario”</b>	

2.1.c El Administrador selecciona dentro del listado de usuarios el que desea eliminar.	2.1.c.1 El sistema elimina el usuario, el sistema muestra un mensaje de confirmación y finaliza el caso de uso.
<b>Prototipo</b>	<b>Ver Anexo #1.</b>

Tabla 3: CU Gestionar Usuario

**Caso de uso “Autenticar”.**

<b>Nombre del Caso de Uso</b>	<b>Autenticar</b>
<b>Actores</b>	Administrador del sistema, vicedecano de producción, secretaria docente, jefe de proyecto y estudiante.
<b>Propósito</b>	Autenticarse cada usuario.
<b>Resumen</b>	El caso de uso se inicia cuando el administrador del sistema, vicedecano de producción, secretaria docente, jefe de proyecto o estudiante desean acceder al sistema, a partir de aquí el sistema brinda la posibilidad a cada usuario del sistema de autenticarse. Termina cuando el usuario del sistema se autentica sin dificultad.
<b>Referencias</b>	<b>R1.3</b>
<b>Precondiciones</b>	Para acceder al sistema debe estar registrado el usuario.
<b>Poscondiciones</b>	El usuario es autenticado, teniendo acceso a determinada sección del sistema de acuerdo al rol que desempeñe.
<b>Requerimientos especiales</b>	
<b>Curso normal de los eventos</b>	
<b>Acciones del actor</b>	<b>Respuestas del Sistema</b>
1. El administrador del sistema, vicedecano de producción, secretaria docente, jefe de proyecto	1.1 El sistema muestra una interfaz que posibilita autenticarse.



o estudiante desean entrar al sistema.	
2. El administrador del sistema, vicedecano de producción secretaria docente, jefe de proyecto o estudiante introducen su usuario y contraseña.	<p>2.1 El sistema verifica que los datos introducidos sean válidos.</p> <p>2.2 Si son válidos se autentica el usuario en el sistema y se accede sin dificultad a la sección a la que tiene permiso el usuario.</p>
<b>Curso alternativo</b>	
	2.1.1 Si no son válidos se emite un mensaje de error para que se llenen los campos obligatorios o se introduzcan correctamente los datos.
<b>Prototipo</b>	<b>Ver Anexo #2.</b>

Tabla 4: CU Autenticar

**Paquete de Registrar Datos**

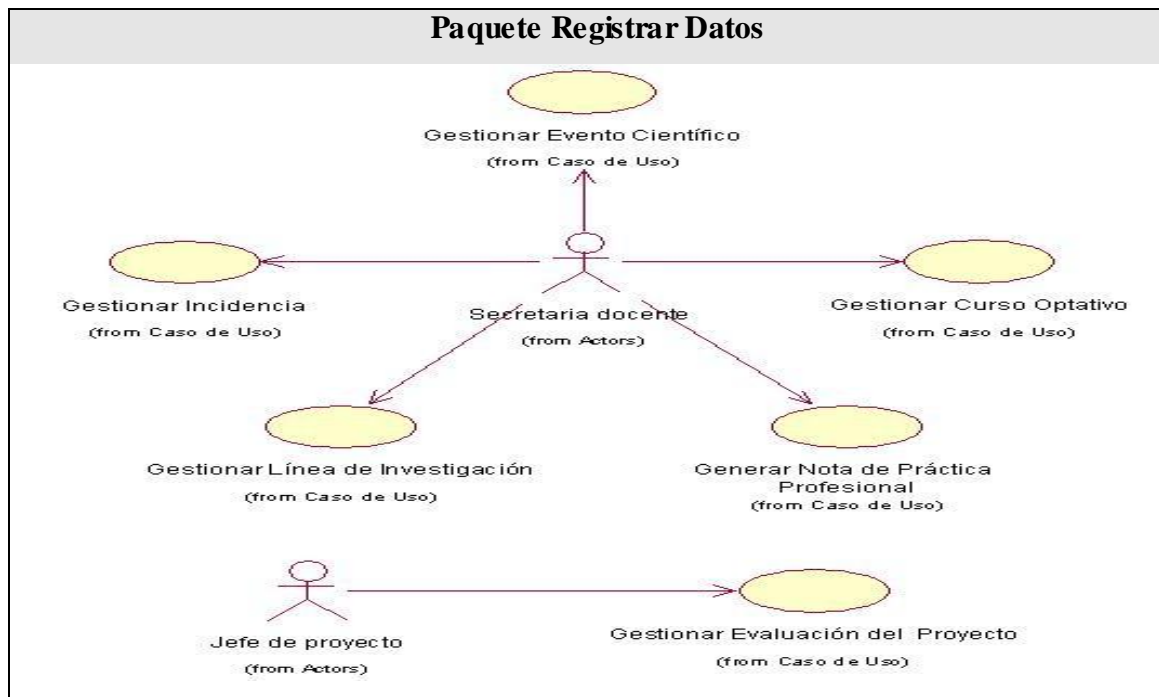


Imagen 4: Paquete Registrar Datos

**Caso de uso “Gestionar Evento Científico”**

<b>Nombre del Caso de Uso</b>		<b>Gestionar Evento Científico</b>
<b>Actores</b>	Secretaria Docente	
<b>Propósito</b>	Agregar, modificar y eliminar evento científico del estudiante.	
<b>Resumen</b>	El caso de uso se inicia cuando la secretaria docente accede al sistema y decide registrar algún evento científico del estudiante.	
<b>Referencias</b>	<b>R2.1</b>	
<b>Precondiciones</b>	Para registrar los eventos científicos del estudiante el mismo debe tener al menos un evento científico.	
<b>Poscondiciones</b>	<p>Para los siguientes procesos:</p> <p>Agregar eventos científicos del estudiante, se registra el nivel alcanzado, resultado y evaluación del evento.</p> <p>Modificar eventos científicos del estudiante, se actualiza el nivel alcanzado, resultado y evaluación del evento.</p> <p>Eliminar eventos científicos del estudiante, se elimina el evento.</p>	
<b>Curso normal de los eventos</b>		
<b>Acciones del actor</b>	<b>Respuestas del Sistema</b>	
1. La Secretaria docente decide Registrar los eventos científicos.	1.1 El sistema muestra una interfaz donde se pueden registrar o modificar los eventos científicos del estudiante.	
2. La Secretaria docente selecciona una de las operaciones a realizar.	2.1 El sistema ejecuta una de las siguientes acciones: <ul style="list-style-type: none"> <li>a) Si decide agregar evento científico ir a la sección</li> </ul>	

	<p>“Agregar Evento Científico”.</p> <p>b) Si decide modificar evento científico ir a la sección “Modificar Evento Científico”.</p> <p>c) Si decide eliminar evento científico ir a la sección “Eliminar Evento Científico”.</p>
<b>2.1a) Sección “Agregar Evento Científico”</b>	
2.1.a Secretaria Docente introduce el nivel alcanzado, resultado y la evaluación del evento científico.	<p>2.1. a.1 El sistema verifica que los datos introducidos sean válidos.</p> <p>2.1. a.2 El evento se registra en el sistema, dando la posibilidad de agregar otro evento y finaliza así el caso de uso.</p>
<b>Curso alternativo</b>	
	2.1. a.1.1 Se emite un mensaje de error para que se introduzcan correctamente los datos.
<b>2.1b) Sección “Modificar Evento Científico”</b>	
2.1.b Secretaria Docente introduce el nivel alcanzado, resultado del evento científico a modificar.	<p>2.1. b.1 El sistema verifica que los datos introducidos sean válidos.</p> <p>2.1. b.2 El evento se registra en el sistema.</p>
<b>Curso alternativo</b>	
	2.1. b.1.1 Se emite un mensaje de error para que se introduzcan

	correctamente los datos.
<b>Prototipo</b>	
<b>2.1c) Sección “Eliminar Evento Científico”</b>	
2.1.c Secretaria Docente selecciona el evento científico a eliminar.	2.1. c.1 El sistema elimina el evento, y se muestra un mensaje de confirmación y finaliza el caso de uso.
<b>Curso alternativo</b>	
<b>Prototipo</b>	<b>Ver Anexo #3.</b>

Tabla 5: CU Gestionar evento científico

**Caso de uso “Gestionar Cursos Optativos”**

<b>Nombre del Caso de Uso</b>	<b>Gestionar Cursos Optativos</b>
<b>Actores</b>	Secretaria Docente
<b>Propósito</b>	Agregar, modificar y eliminar los cursos optativos impartidos por el estudiante.
<b>Resumen</b>	El caso de uso se inicia cuando la secretaria docente accede al sistema y decide registrar, modificar o eliminar los cursos optativos impartidos por el estudiante.
<b>Referencias</b>	<b>R2.2</b>
<b>Precondiciones</b>	Para registrar los cursos optativos del estudiante el mismo debe tener al menos un curso optativo recibido o impartido.
<b>Poscondiciones</b>	Para los siguientes procesos: Agregar curso optativo impartido por el estudiante, se registra nombre del curso optativo impartido. Modificar curso optativo impartido por el estudiante, se modifican el curso optativo impartido. Eliminar curso optativo impartido por el estudiante, se elimina curso optativo impartido.

<b>Curso normal de los eventos</b>	
<b>Acciones del actor</b>	<b>Respuestas del Sistema</b>
1. La Secretaria docente decide Registrar los cursos optativos impartidos por el estudiante.	1.1 El sistema muestra una interfaz donde se pueden registrar, modificar o eliminar los cursos optativos del estudiante.
✓ La Secretaria docente selecciona una de las operaciones a realizar.	2.1 El sistema ejecuta una de las siguientes acciones: a) Si decide adicionar un curso optativo impartido ir la sección "Agregar Curso Optativo Impartido". b) Si decide modificar curso optativo impartido ir a la sección "Modificar Curso Optativo Impartido". c) Si decide eliminar un curso optativo impartido ir la sección "Eliminar Curso Optativo Impartido".
<b>2.1a) Sección "Agregar Curso Optativo Impartido"</b>	
2.1.a La Secretaria Docente introduce el nombre del curso optativo impartido.	2.1. a.1 El sistema verifica que los datos introducidos sean válidos. 2.1. a.2 Se registra el curso en el sistema y finaliza así el caso de uso.
<b>Curso alternativo</b>	
	2.1. a.1.1 Se emite un mensaje de error para que se introduzcan correctamente los datos.
<b>2.1b) Sección "Modificar Curso Optativo Impartido"</b>	
2.1.b La Secretaria Docente introduce la modificación del nombre al curso optativo	2.1.b.1 El sistema verifica que los datos introducidos sean válidos.

impartido.	2.1.b.2 Se actualiza el nombre del curso optativo en el sistema y finaliza así el caso de uso.
<b>Curso alternativo</b>	
	2.1.b1.1 Se emite un mensaje de error para que se introduzcan correctamente los datos.
<b>Prototipo</b>	
<b>2.1c) Sección “Eliminar Curso Optativo Impartido”</b>	
2.1.c Secretaria Docente selecciona el curso optativo impartido a eliminar.	2.1. c.1 El sistema elimina el curso optativo impartido, y se muestra un mensaje de confirmación y finaliza el caso de uso.
<b>Curso alternativo</b>	
<b>Prototipo</b>	<b>Ver Anexo #4</b>

Tabla 6: CU Gestionar cursos optativos.

**Caso de uso “Gestionar Líneas de Investigación”**

<b>Nombre del Caso de Uso</b>	<b>Gestionar Líneas de Investigación</b>
<b>Actores</b>	Secretaria Docente
<b>Propósito</b>	Registrar, modificar y eliminar líneas de investigación de los estudiantes.
<b>Resumen</b>	El caso de uso se inicia cuando la Secretaria Docente accede al sistema y solicita registrar las líneas de investigación, a partir de aquí el sistema brinda la posibilidad de realizar las operaciones de registrar las líneas de investigación por los estudiantes. Termina cuando se realiza esta operación.

<b>Referencias</b>	<b>R2.3</b>	
<b>Precondiciones</b>	Para registrar una línea de investigación, el estudiante debe tener al menos una línea de investigación.	
<b>Poscondiciones</b>		
<b>Curso normal de los eventos</b>		
<b>Acciones del actor</b>	<b>Respuestas del Sistema</b>	
1. La Secretaria docente decide Registrar las Líneas de Investigación del estudiante.	1.1 El sistema muestra una interfaz donde se pueden registrar, modificar o eliminar las Líneas de Investigación del estudiante.	
2. La Secretaria docente selecciona una de las operaciones a realizar.	2.1 El sistema ejecuta una de las siguientes acciones: a) Si decide registrar las Líneas de Investigación ir la sección “Registrar las Líneas de Investigación”. b) Si decide modificar las Líneas de Investigación ir a la sección “Modificar las Líneas de Investigación”. c) Si decide eliminar las Líneas de Investigación ir la sección “Eliminar las Líneas de Investigación”.	
<b>2.1a) Sección “Registrar las Líneas de Investigación”</b>		
2.1.a La Secretaria Docente introduce las Líneas de Investigación.	2.1. a.1 El sistema verifica que los datos introducidos sean válidos. 2.1. a.2 Se registran las Líneas de Investigación en el sistema y finaliza así el caso de uso.	
<b>Curso alternativo</b>		
	2.1. a.1.1 Se emite un mensaje de error para que se introduzcan	

	correctamente los datos.
<b>2.1b) Sección “Modificar las Líneas de Investigación”</b>	
2.1.b La Secretaria Docente introduce la modificación de las Líneas de Investigación.	2.1.b.1 El sistema verifica que los datos introducidos sean válidos. 2.1.b.2 Se actualiza la Línea de Investigación en el sistema y finaliza así el caso de uso.
<b>Curso alternativo</b>	
	2.1.b1.1 Se emite un mensaje de error para que se introduzcan correctamente los datos.
<b>2.1c) Sección “Eliminar las Líneas de Investigación”</b>	
2.1.c Secretaria Docente selecciona la Línea de Investigación a eliminar.	2.1. c.1 El sistema elimina la Línea de Investigación, y se muestra un mensaje de confirmación y finaliza el caso de uso.
<b>Curso alternativo</b>	
<b>Prototipo</b>	<b>Ver Anexo #5</b>

Tabla 7: CU Gestionar líneas de investigación

**Caso de uso “Gestionar Incidencias”**

<b>Nombre del Caso de Uso</b>	<b>Gestionar Incidencias</b>
<b>Actores</b>	Secretaria Docente
<b>Propósito</b>	Registrar, modificar y eliminar incidencias de los estudiantes.
<b>Resumen</b>	El caso de uso se inicia cuando la Secretaria Docente accede al sistema y solicita registrar incidencia, a partir de aquí el sistema brinda la posibilidad de realizar la



	operación de registrar incidencia. Una vez registrada la incidencia el sistema brinda la oportunidad de registrar otra incidencia. Termina cuando se realiza esta operación.	
<b>Referencias</b>	<b>R2.4</b>	
<b>Precondiciones</b>	Para registrar las incidencias, debe existir el estudiante.	
<b>Poscondiciones</b>		
<b>Curso normal de los eventos</b>		
<b>Acciones del actor</b>	<b>Respuestas del Sistema</b>	
1. La Secretaria docente decide Registrar las incidencias del estudiante.	1.1 El sistema muestra una interfaz donde se pueden registrar, modificar o eliminar las incidencias del estudiante.	
2. La Secretaria docente selecciona una de las operaciones a realizar.	2.1 El sistema ejecuta una de las siguientes acciones: a) Si decide registrar las incidencias ir a la sección "Registrar las incidencias". b) Si decide modificar las incidencias ir a la sección "Modificar las incidencias". c) Si decide eliminar las incidencias ir a la sección "Eliminar las incidencias".	
<b>2.1a) Sección "Registrar las incidencias"</b>		
2.1.a La Secretaria Docente introduce las incidencias.	2.1. a.1 El sistema verifica que los datos introducidos sean válidos. 2.1. a.2 Se registran las incidencias en el sistema y finaliza así el caso de uso.	
<b>Curso alternativo</b>		
	2.1. a.1.1 Se emite un mensaje de error para que se introduzcan correctamente los datos.	

<b>2.1b) Sección “Modificar las Líneas de Investigación”</b>	
2.1.b La Secretaria Docente introduce la modificación de las incidencias.	2.1.b.1 El sistema verifica que los datos introducidos sean válidos. 2.1.b.2 Se actualiza las incidencias en el sistema y finaliza así el caso de uso.
<b>Curso alternativo</b>	
	2.1.b1.1 Se emite un mensaje de error para que se introduzcan correctamente los datos.
<b>2.1c) Sección “Eliminar las Líneas de Investigación”</b>	
2.1.c Secretaria Docente selecciona las incidencias a eliminar.	2.1. c.1 El sistema elimina las incidencias, y se muestra un mensaje de confirmación y finaliza el caso de uso.
<b>Curso alternativo</b>	
<b>Prototipo</b>	<b>Ver Anexo #6</b>

Tabla 8: CU Gestionar incidencia

**Caso de uso “Gestionar Evaluación del Proyecto”**

<b>Nombre del Caso de Uso</b>	<b>Gestionar Evaluación del Proyecto</b>
<b>Actores</b>	Jefe de Proyecto
<b>Propósito</b>	Registrar, modificar y eliminar la evaluación de proyecto de los estudiantes.
<b>Resumen</b>	El caso de uso se inicia cuando el Jefe de Proyecto accede al sistema y solicita registrar, modificar o eliminar la evaluación de proyecto, a partir de aquí el sistema brinda la posibilidad de realizar las operaciones de registrar o modificar la evaluación de los estudiantes del proyecto. Termina cuando se realiza esta operación.

<b>Referencias</b>	R2.7	
<b>Precondiciones</b>	Para registrar, modificar o eliminar la evaluación, debe existir el estudiante.	
<b>Poscondiciones</b>		
<b>Curso normal de los eventos</b>		
<b>Acciones del actor</b>	<b>Respuestas del Sistema</b>	
1. El Jefe de proyecto selecciona esta opción.	1.1 El sistema muestra una interfaz que da la posibilidad de registrar o modificar la evaluación de proyecto.	
2. El Jefe de proyecto selecciona una de las operaciones a realizar.	2.1 El sistema ejecuta una de las siguientes acciones: a) Si decide registrar la evaluación del proyecto ir a la sección "Registrar Evaluación del proyecto". b) Si decide modificar la evaluación del proyecto ir a la sección "Modificar evaluación del proyecto". c) Si decide eliminar la evaluación del proyecto ir a la sección "Eliminar evaluación del proyecto".	
<b>Curso alternativo</b>		
<b>2.1a) Sección "Registrar Evaluación del proyecto"</b>		
2.1.a El Jefe de proyecto introduce la evaluación del proyecto.	2.1. a.1 El sistema verifica que los datos introducidos sean válidos. 2.1. a.2 Se registra la evaluación del proyecto en el sistema y finaliza así el caso de uso.	
<b>Curso alternativo</b>		

	2.1. a.1.1 Se emite un mensaje de error para que se introduzcan correctamente los datos.
<b>2.1b) Sección “Modificar evaluación del proyecto”</b>	
2.1.b El Jefe del proyecto introduce la modificación de la evaluación del proyecto.	2.1.b.1 El sistema verifica que los datos introducidos sean válidos. 2.1.b.2 Se actualiza la evaluación del proyecto en el sistema y finaliza así el caso de uso.
<b>Curso alternativo</b>	
	2.1.b.1.1 Se emite un mensaje de error para que se introduzcan correctamente los datos.
<b>2.1c) Sección “Eliminar evaluación del proyecto”</b>	
2.1.c El Jefe del proyecto introduce la evaluación del proyecto a eliminar.	2.1. c.1 El sistema elimina la evaluación del proyecto y se muestra un mensaje de confirmación y finaliza el caso de uso.
<b>Curso alternativo</b>	
<b>Prototipo</b>	<b>Ver Anexo #7</b>

Tabla 9: CU Gestionar evaluación proyecto.

**Caso de uso “Generar la Nota de PP”**

<b>Nombre del Caso de Uso</b>	Generar la Nota de PP
<b>Actores</b>	Secretaria Docente
<b>Propósito</b>	Generar la Nota de PP
<b>Resumen</b>	El caso de uso se inicia cuando la Secretaria Docente accede al sistema y genera la Nota de PP, a partir de aquí el sistema brinda la posibilidad de realizar la

	operación de generar la Nota de PP. Termina cuando se realiza esta operación.	
<b>Referencias</b>	<b>R2.6</b>	
<b>Precondiciones</b>	Para generar la Nota de PP, debe existir el estudiante.	
<b>Poscondiciones</b>		
<b>Curso normal de los eventos</b>		
<b>Acciones del actor</b>	<b>Respuestas del Sistema</b>	
1. La secretaria docente decide generar la Nota de PP del estudiante.	1.1 El sistema muestra una interfaz que da la posibilidad de generar la Nota de PP y finaliza así el caso de uso.	
<b>Curso alternativo</b>		
<b>Prototipo</b>	<b>Ver Anexo #8</b>	

Tabla 10: CU Generar la nota PP

**Paquete Salida de Datos.**

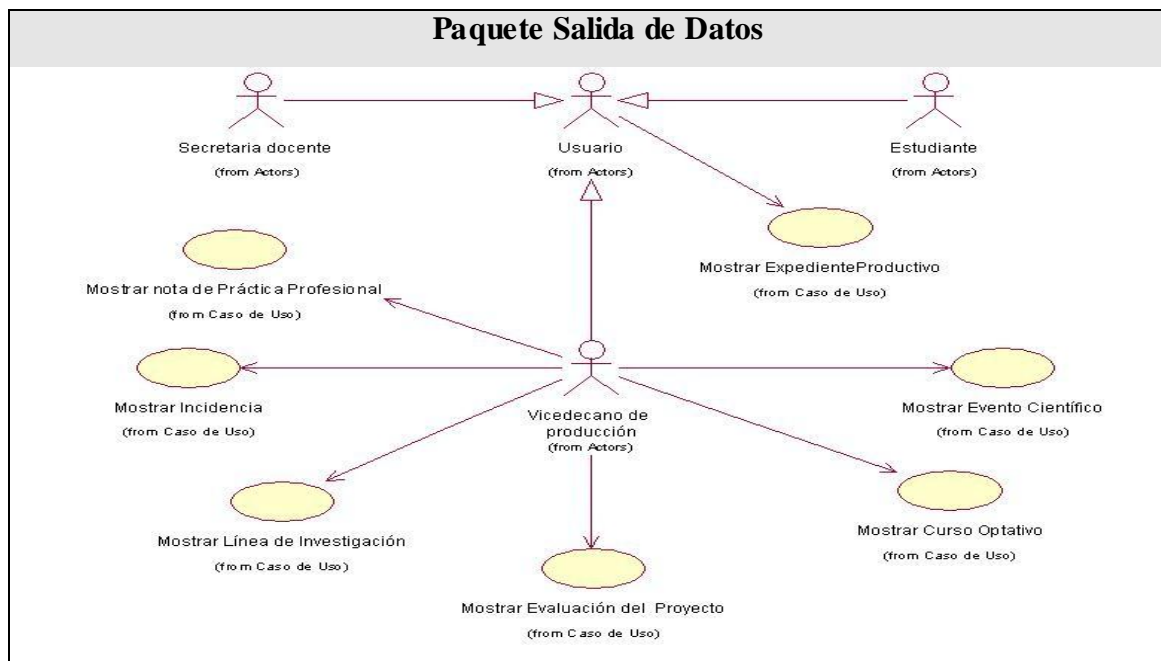


Imagen 5: Paquete salida de datos.

**Caso de uso “Mostrar Expediente Productivo”**

<b>Nombre del Caso de Uso</b>	<b>Mostrar Expediente Productivo</b>	
<b>Actores</b>	Vicedecano Producción, Secretaria y Estudiante	
<b>Propósito</b>	Mostrar el expediente productivo del estudiante.	
<b>Resumen</b>	El caso de uso se inicia cuando el Vicedecano Producción, Secretaria o Estudiante acceden al sistema y deciden ver el expediente productivo, el mismo tiene la opción de imprimir el expediente.	
<b>Referencias</b>	<b>R3.1</b>	
<b>Precondiciones</b>		
<b>Poscondiciones</b>		
<b>Curso normal de los eventos</b>		
<b>Acciones del actor</b>	<b>Respuestas del Sistema</b>	
1. El Vicedecano Producción, Secretaria o Estudiante quieren ver el expediente productivo.	1.1 El sistema recopila los datos necesarios.  1.2 El sistema muestra el expediente productivo y finaliza así el caso de uso.	
<b>Curso alternativo</b>		
<b>Prototipo</b>	<b>Ver Anexo #9</b>	

Tabla 11: CU Mostrar expediente productivo.

**Caso de uso “Mostrar Eventos Científicos”**

<b>Nombre del Caso de Uso</b>	<b>Mostrar Eventos Científicos</b>	
<b>Actores</b>	Vicedecano de Producción	
<b>Propósito</b>	Mostrar los eventos científicos del estudiante.	
<b>Resumen</b>	El caso de uso se inicia cuando el Vicedecano de Producción accede al sistema y decide ver los eventos científicos de algún estudiante.	
<b>Referencias</b>	<b>R3.2</b>	
<b>Precondiciones</b>	Para mostrar los eventos científicos del estudiante, el mismo debe tener registrado al menos un evento científico.	
<b>Poscondiciones</b>	Se muestra él o los eventos científicos registrados.	
<b>Curso normal de los eventos</b>		
<b>Acciones del actor</b>	<b>Respuestas del Sistema</b>	
1. El Vicedecano de Producción selecciona el estudiante del cual quiere ver sus eventos científicos.	1.1 El sistema recopila los datos necesarios. 1.2 El sistema muestra él o los eventos científicos registrados.	
<b>Curso alternativo</b>		
<b>Prototipo</b>	<b>Ver Anexo #10</b>	

Tabla 12: CU Mostrar evento científico

**Caso de uso “Mostrar Cursos Optativos”**

<b>Nombre del Caso de Uso</b>	<b>Mostrar Cursos Optativos</b>	
<b>Actores</b>	Vicedecano de Producción	
<b>Propósito</b>	Mostrar los cursos optativos impartidos por el estudiante.	
<b>Resumen</b>	El caso de uso se inicia cuando el Vicedecano de Producción accede al sistema y decide ver los cursos optativos impartidos por algún estudiante.	
<b>Referencias</b>	<b>R3.3</b>	
<b>Precondiciones</b>	Para mostrar los cursos optativos impartidos por el estudiante, debe tener registrado al menos un curso optativo impartido.	
<b>Poscondiciones</b>		
<b>Curso normal de los eventos</b>		
<b>Acciones del actor</b>	<b>Respuestas del Sistema</b>	
1. El Vicedecano de Producción selecciona el estudiante del cual quiere ver sus cursos optativos impartidos.	1.1 El sistema recopila los datos necesarios. 1.2 El sistema muestra él o los cursos optativos impartidos y finaliza el caso de uso.	
<b>Curso alternativo</b>		
<b>Prototipo</b>	<b>Ver Anexo #11</b>	

Tabla 13: CU Mostrar curso optativo



**Caso de uso “Mostrar Líneas de Investigación”**

<b>Nombre del Caso de Uso</b>		<b>Mostrar Líneas de Investigación</b>
<b>Actores</b>	Vicedecano de Producción	
<b>Propósito</b>	Mostrar las líneas de investigación del estudiante.	
<b>Resumen</b>	El caso de uso se inicia cuando el Vicedecano de Producción accede al sistema y decide ver las líneas de investigación de algún estudiante.	
<b>Referencias</b>	<b>R3.4</b>	
<b>Precondiciones</b>	Para mostrar las líneas de investigación del estudiante, el mismo debe tener registrado al menos una línea de investigación.	
<b>Poscondiciones</b>	Se muestran las líneas de investigación que tenga el estudiante.	
<b>Curso normal de los eventos</b>		
<b>Acciones del actor</b>	<b>Respuestas del Sistema</b>	
1. El Vicedecano de Producción selecciona el estudiante del cual quiere ver sus líneas de investigación.	1.1 El sistema recopila los datos necesarios.  1.2 El sistema muestra las líneas de investigación que tenga registrada el estudiante y finaliza el caso de uso.	
<b>Curso alternativo</b>		
<b>Prototipo</b>	<b>Ver Anexo #12</b>	

Tabla 14: CU Mostrar líneas de investigación.

**Caso de uso “Mostrar Incidencias”**

<b>Nombre del Caso de Uso</b>		<b>Mostrar Incidencias</b>
<b>Actores</b>	Vicedecano de Producción	
<b>Propósito</b>	Mostrar las incidencias del estudiante.	
<b>Resumen</b>	El caso de uso se inicia cuando el Vicedecano de Producción accede al sistema y decide ver las incidencias de algún estudiante.	
<b>Referencias</b>	<b>R3.5</b>	
<b>Precondiciones</b>	Para mostrar las incidencias del estudiante, el mismo debe tener registrado al menos una incidencia.	
<b>Poscondiciones</b>	Se muestran las incidencias que tenga el estudiante.	
<b>Curso normal de los eventos</b>		
<b>Acciones del actor</b>		<b>Respuestas del Sistema</b>
1. El Vicedecano de Producción selecciona el estudiante del cual quiere ver sus incidencias.		1.1 El sistema recopila los datos necesarios. 1.2 El sistema muestra las incidencias que tenga registrada el estudiante y finaliza el caso de uso.
<b>Curso alternativo</b>		
<b>Prototipo</b>	<b>Ver Anexo #13</b>	

Tabla 15: CU Mostrar incidencias.

**Caso de uso “Mostrar Evaluación de Proyecto”**

<b>Nombre del Caso de Uso</b>	Mostrar Evaluación de Proyecto	
<b>Actores</b>	Vicedecano de Producción	
<b>Propósito</b>	Mostrar Evaluación de Proyecto	
<b>Resumen</b>	El caso de uso se inicia cuando el Vicedecano de Producción accede al sistema y decide ver la evaluación de proyecto de algún estudiante.	
<b>Referencias</b>	<b>R3.6</b>	
<b>Precondiciones</b>	Para mostrar evaluación de proyecto del estudiante, el mismo debe pertenecer a un proyecto.	
<b>Poscondiciones</b>	Se muestra la evaluación de proyecto que tenga el estudiante.	
<b>Curso normal de los eventos</b>		
<b>Acciones del actor</b>	<b>Respuestas del Sistema</b>	
1. El Vicedecano Producción selecciona el estudiante del cual quiere ver su evaluación de proyecto.	1.1 El sistema recopila los datos necesarios. 1.2 El sistema muestra evaluación de proyecto que tenga registrada el estudiante y finaliza el caso de uso.	
<b>Curso alternativo</b>		
<b>Prototipo</b>	<b>Ver Anexo #14</b>	

Tabla 16: CU Mostrar evaluación de proyecto.

**Caso de uso “Mostrar nota de Práctica Profesional”**

<b>Nombre del Caso de Uso</b>		<b>Mostrar nota de Práctica Profesional</b>
<b>Actores</b>	Vicedecano de Producción	
<b>Propósito</b>	Mostrar las notas de Práctica Profesional	
<b>Resumen</b>	El caso de uso se inicia cuando el Vicedecano de Producción accede al sistema y decide ver la nota de práctica profesional de algún estudiante.	
<b>Referencias</b>	<b>R3.7</b>	
<b>Precondiciones</b>	Para mostrar la nota de práctica profesional del estudiante, al mismo se le debe haber generado la nota.	
<b>Poscondiciones</b>	Se muestra la nota de práctica profesional que tenga el estudiante.	
<b>Curso normal de los eventos</b>		
<b>Acciones del actor</b>		<b>Respuestas del Sistema</b>
1. El Vicedecano de Producción selecciona el estudiante del cual quiere ver su nota de práctica profesional.		1.1 El sistema recopila los datos necesarios.  1.2 El sistema muestra la nota de práctica profesional que tenga registrada el estudiante y finaliza el caso de uso.
<b>Curso alternativo</b>		
<b>Prototipo</b>	<b>Ver Anexo #15</b>	

Tabla 17: CU Mostrar nota de PP

### **Conclusiones**

Este capítulo sirvió como iniciador para el desarrollo del sistema propuesto. Se realizó el modelo del dominio, agrupando los conceptos que intervienen en el contexto del sistema, se determinaron los requerimientos del sistema y se determinaron los casos de usos correspondientes y sus concernientes descripciones para la realización de cada uno de los mismos.

### CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

#### Introducción

En este capítulo correspondiente al análisis y diseño del sistema, se profundizará en los casos de usos detallados, de manera que permitan reflejar una vista interna del sistema descrito, con el lenguaje de los desarrolladores. En esta vista se especifican mejor los casos de uso y se determinan las clases necesarias para llevar a cabo las funcionalidades contenidas en ellos. Se abordarán aspectos relacionados con este flujo de trabajo y se presentarán los artefactos vinculados al mismo.

#### 3.1 Análisis de sistema

En el modelo de análisis se refinan los casos de uso, otorgándole más detalle, permite asignar la funcionalidad a un grupo de objetos. Las clases que se utilizan para modelar el análisis, se centran en el tratamiento de los requerimientos funcionales, y se estereotipan de la siguiente manera:

- ✓ Clase interfaz: Modelan la interacción entre el sistema y sus actores.
- ✓ Clases controladoras: Coordinan la realización de uno o unos pocos casos de uso, relacionando las actividades de los objetos que implementan sus funcionalidades.
- ✓ Clases entidad: Modelan información que posee larga vida y que es a menudo persistente.

#### 3.2 Diagrama de clases de análisis

Son un artefacto en el que se representan los conceptos en un dominio del problema. Representa las cosas del mundo real, no de la implementación. A continuación se representan los diagramas de clases del análisis, por cada realización de caso de uso del sistema, para su mayor comprensión. A continuación se mostrarán los diagramas de las clases del análisis del paquete de administración y el resto se podrán ver desde el anexo 16 al 26.

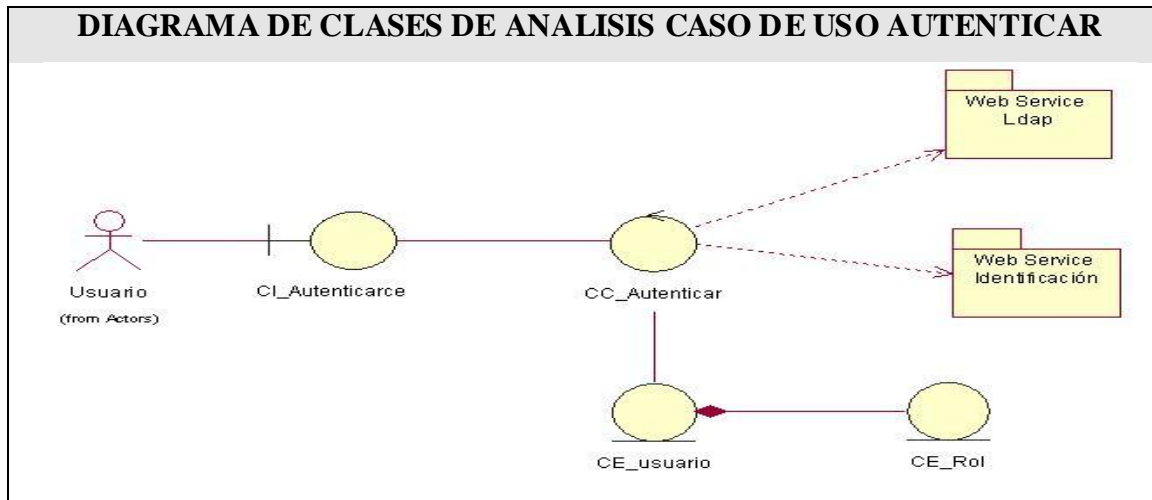


Imagen 6: Diagrama de clases de análisis CU autenticar

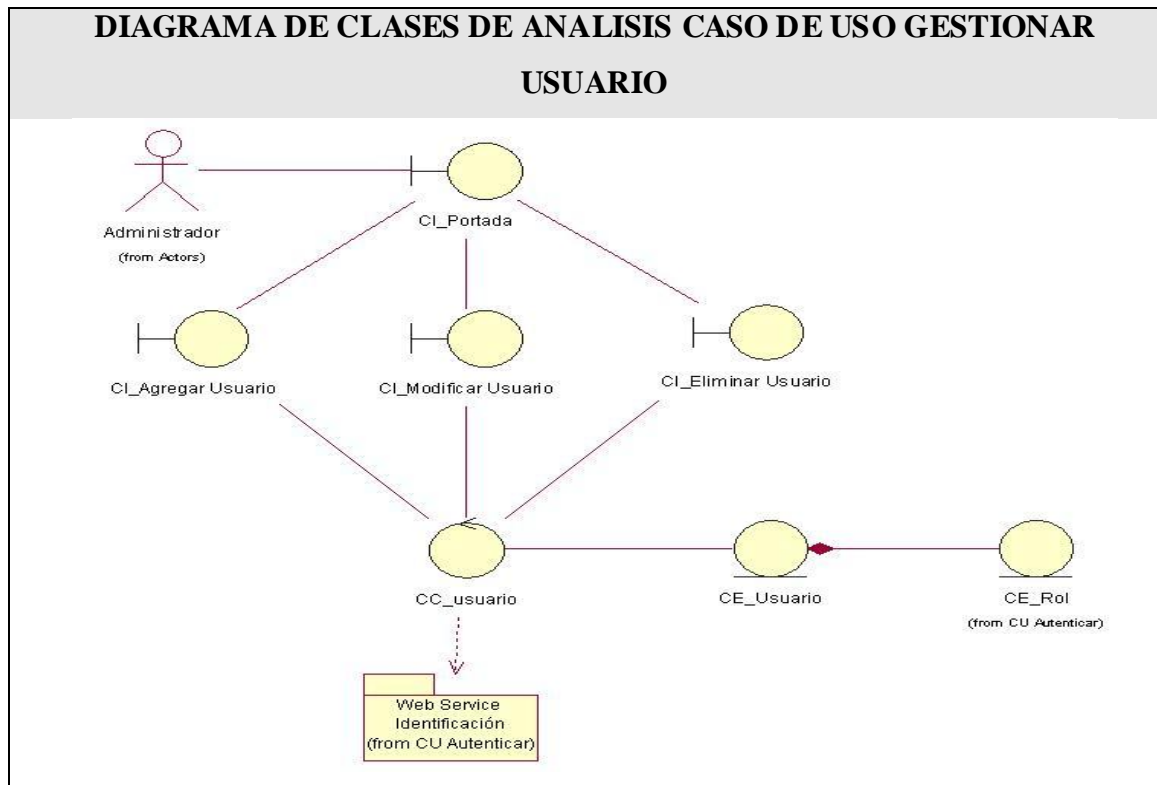


Imagen 7: Diagrama de clases de análisis CU gestionar usuario.

### 3.2.1 Diagramas de interacción

Muestran una interacción concreta: un conjunto de objetos y sus relaciones, junto con los mensajes que se envían entre ellos. Modelan el comportamiento dinámico del sistema; el flujo de control en una operación. Describen la interacción entre objetos; los objetos interactúan a

través de mensajes para cumplir ciertas tareas. Las interacciones proveen un “comportamiento” y típicamente implementan un Caso de Uso.

- Existen dos tipos de diagramas de interacción en UML:
  - Diagramas de Secuencia (*dimensión temporal*)
  - Diagramas de Colaboración (*dimensión estructural*)

### 3.2.2 Diagramas de Colaboración

Un diagrama de colaboración es muy similar a un diagrama de secuencia en el propósito que alcanza; es decir demuestra la interacción dinámica de los objetos en un sistema. Una característica que distingue un diagrama de colaboración es que demuestra los objetos y su asociación con otros objetos en el sistema. La asociación entre los objetos no se representa en un diagrama de secuencia.

A continuación se mostrarán los diagramas de colaboración del análisis del paquete de administración y el resto se podrán ver desde el anexo 27 al 73

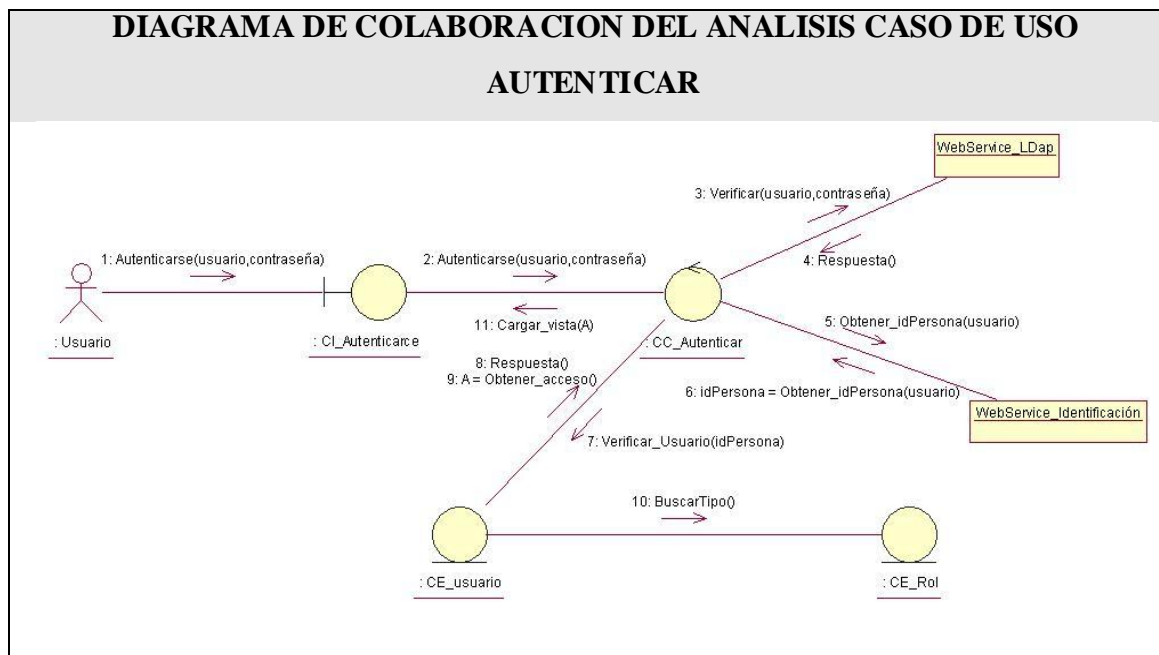


Imagen 8: Diagrama de colaboración CU autenticar.



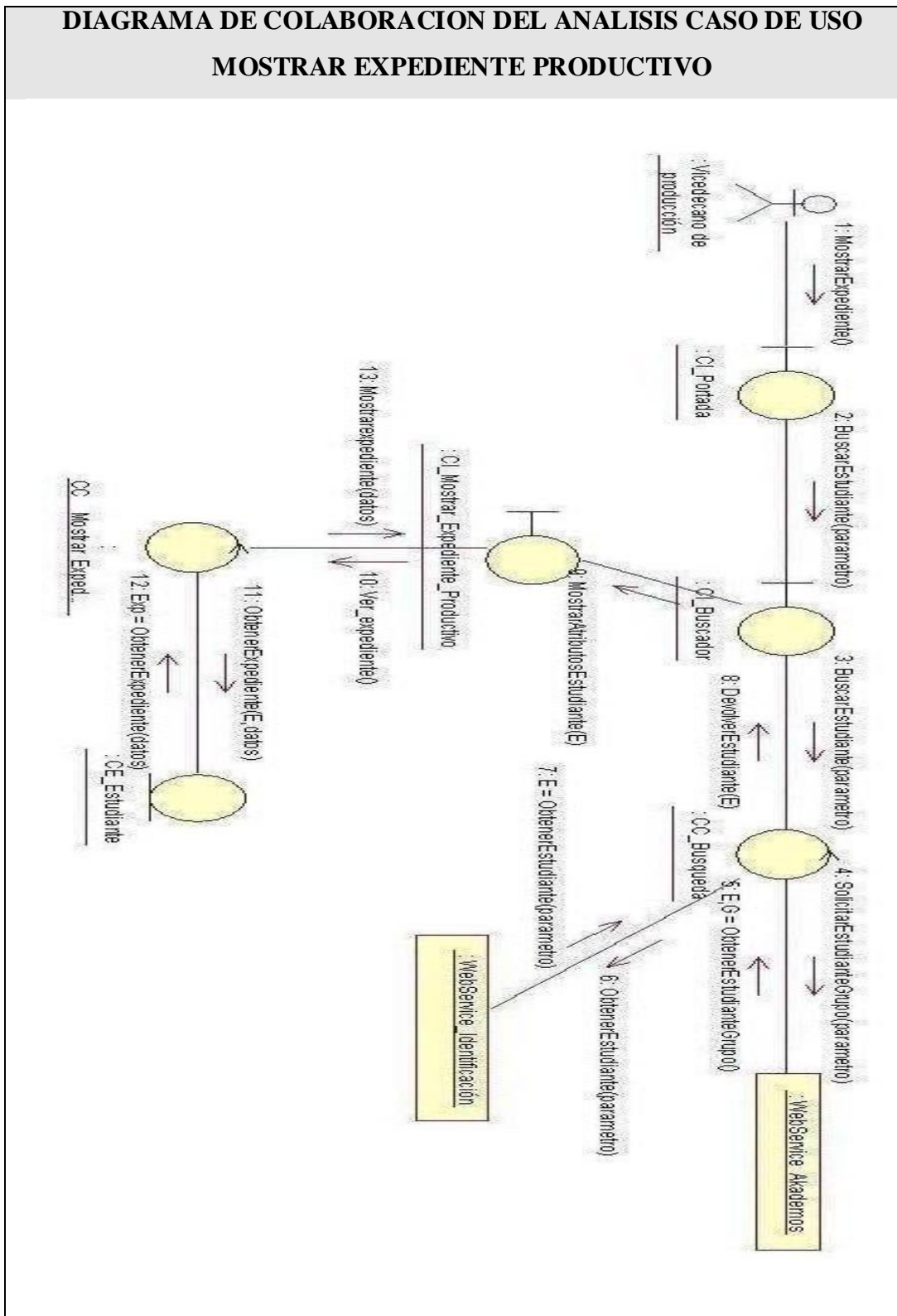


Imagen 9: Diagrama de colaboración del análisis CU Mostrar Expediente Productivo.

### 3.3 Clases bases

Se consideran las posibilidades de reutilización de las partes del sistema parecidas o de productos software general. Los subsistemas, interfaces u otros elementos del diseño se añaden consecutivamente. Se decide definir el sistema de clases base de la aplicación, pues se persigue facilidad de mantenimiento, escalabilidad, rapidez de desarrollo e independencia entre las distintas capas del sistema para facilitar su futura evolución.

#### Diagrama de Clases bases.

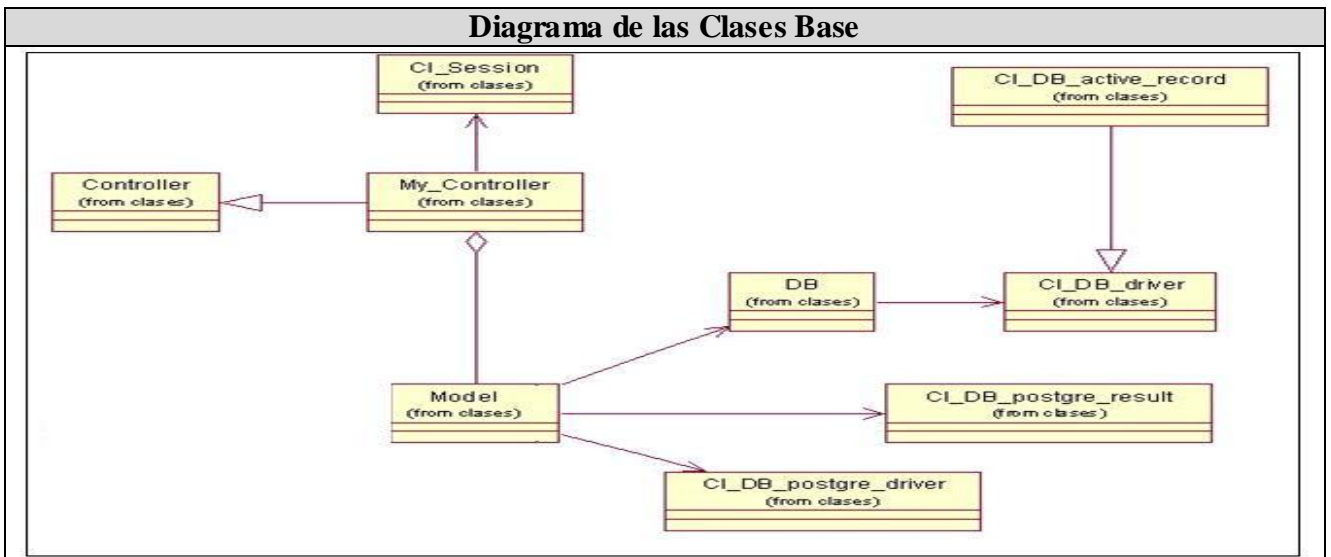


Imagen 10: Diagrama de clases bases.

#### Descripción de las clases.

<b>Controller</b>	<b>Súper clase que se encarga de servir de base funcional de las clases controladoras.</b>
Métodos	
Controller	Constructor de la clase que se encarga de inicializar los elementos.
_ci_initialize	Asigna todas las clases bases cargadas por el inicio de la Controller por variables en esta clase. También se encarga de llamar al autoload de la aplicación.

<b>MY_Controller</b>	<b>(estos son Clase que se encarga de servir de base</b>
----------------------	----------------------------------------------------------

métodos)	añadiendo nuevos métodos funcionales.
Métodos	
MY_Controller	Constructor de la clase que se encarga de inicializar los elementos.
Akademos	Hace la conexión con el servicio web de akademos.
Identificacion	Hace la conexión con el servicio web de identificación.
BuscarGrupos	Devuelve todos los grupos de la Facultad 2 utilizando el método Akademos, pasándole el identificador de la facultad.
Obtener_Cursos_Optativos	Devuelve todas las asignaturas cursadas de un estudiante utilizando el método Akademos, pasándole por parámetros el identificador del estudiante (su identificador de persona).
Busqueda_por_Grupo	Devuelve todos los estudiantes de un grupo utilizando el método Akademos, pasándole por parámetros el grupo y el identificador de la facultad 2.
Busqueda_persona	Devuelve la persona con todos sus métodos utilizando el servicio web de identificación, pasándole por parámetros el identificador de la persona.
Obtener_est_akademos	Devuelve el estudiante con todos sus métodos utilizando el servicio web de akademos, pasándole por parámetros el identificador del estudiante.
Busqueda_persona_user	Devuelve la persona con todos sus métodos utilizando el servicio web de identificación, pasándole por parámetros el usuario de la persona.
Busqueda_persona_solapin	Devuelve la persona con todos sus métodos utilizando el servicio web de identificación,

	pasándole por parámetros el solapín de la persona.
Busqueda_persona_CI	Devuelve la persona con todos sus métodos utilizando el servicio web de identificación, pasándole por parámetros el carnet de identidad de la persona.
Busqueda_foto	Hace la conexión con el repositorio de fotos utilizando el servicio web de identificación.
Busqueda_Avanzada	Devuelve la persona con todos sus métodos utilizando los métodos Busqueda_persona_user, Busqueda_persona_solapin, Busqueda_persona_CI, pasándole por parámetros el usuario o el solapín o el carnet de identidad de la persona.

<b>Session</b>	<b>Clase que se encarga de tratar todos los métodos y funcionalidades para el trabajo con sesión</b>
Métodos	
Session	Constructor de la clase que se encarga de inicializar los elementos.
regenerate_id	Genera el id de la sesión
destroy	Destruye la sesión y borra todo lo esta guardado en la sesión
get	Devuelve lo guardado en la sesión
exist	Verifica si existe un atributo en la sesión
set	Guarda en atributos en la sesión
delete	Borra los atributos de las sesiones
_session_id_expired	Chequea si la sesión ha expirado
<b>CI_DB_postgre_driver</b>	<b>Clase que se encarga de proveer métodos y funcionalidades importante a la hora del manejo de</b>

	<b>datos</b>
Métodos	
CI_DB_postgre_driver	Constructor de la clase que se encarga de inicializar los elementos.
db_connect	Conexión no persistente a la base de datos.
db_pconnect	Conexión persistente a la base de datos.
_version	Muestra la versión.
_execute	Ejecuta un query.
trans_begin	Inicia transacciones.
trans_commit	Termina transacciones.
trans_rollback	Vuelve a atrás las transacciones.
insert_id	Inserta un id.
count_all	Cuenta todas las consultas.
_list_tables	Muestra la consulta de una tabla.
_error_message	Muestra mensajes de error.
_insert	Dado los datos forma una consulta Insert.
_update	Dado los datos forma una consulta Update.
_delete	Dado los datos forma una consulta Delete.
_close	Cierra la conexión a la base de datos.

<b>CI_DB_postgre_result</b>	<b>Clase que se encarga de proveer métodos y funcionalidades importante a la hora del manejo de datos</b>
Métodos	
CI_DB_postgre_result	Constructor de la clase que se encarga de inicializar los elementos.
num_rows	Muestra el números de filas del resultado enviado
num_fields	Muestra el números de campos del resultado enviado
list_fields	Genera un arreglo de columnas con sus nombres
field_data	Obtiene los datos del campo

<b>DB</b>	<b>Clase de gestionar el acceso a métodos de clases para el acceso a datos.</b>
Métodos	
DB	Constructor de la clase que se encarga de inicializar los elementos. Además carga todas las clase instanciadas a el.

<b>CI_DB_driver</b>	<b>Clase que se encarga de proveer métodos y funcionalidades importante a la hora del manejo de datos</b>
Métodos	
CI_DB_driver	Constructor de la clase que se encarga de inicializar los elementos.
initialize	Inicializa archivos de la base de datos.
platform	Devuelve el nombre de la plataforma de base de datos que esta en uso.
version	Devuelve la versión de la base de datos
query	Ejecuta una consulta dada.
trans_off	Deshabilita transacciones
trans_start	Habilita transacciones
trans_complete	Transacciones completadas
trans_status	Estado de las transacciones
is_write_type	Comprueba si la consulta es del tipo escritura
total_queries	Retorna el numero total de consulta
last_query	Retorna el ultimo query que ha sido ejecutado
list_tables	Retorna un arreglo con nombres de las tablas
table_exists	Determina si una tabla dada existe
field_exists	Determina si un campo dada existe
field_data	Retorna un objeto con los datos de los campos
insert_string	Genera una consulta Insert

update_string	Genera una consulta Update
display_error	Muestra un error en pantalla
<b>CI_DB_active_record</b>	<b>Clase que se encarga de proveer métodos y funcionalidades importante a la hora del manejo de datos</b>
Métodos	
CI_DB_active_record	Constructor de la clase que se encarga de inicializar los elementos.
select	Genera la porción del select de la consulta
from	Genera la porción del from de la consulta
join	Genera la porción del join de la consulta
where	Genera la porción del where de la consulta
like	Genera la porción del %like% de la consulta
groupby	Genera la porción del groupby de la consulta
having	Genera la porción del having de la consulta
orderby	Genera la porción del orderby de la consulta
limit	Genera la porción del limit de la consulta
insert	Genera la porción del insert de la consulta
update	Genera la porción del update de la consulta
delete	Genera la porción del delete de la consulta
order_by	Genera la porción del order_by de la consulta

Tabla 18: Descripción de clases.

**Diagramas de secuencia de clases bases para el constructor.**

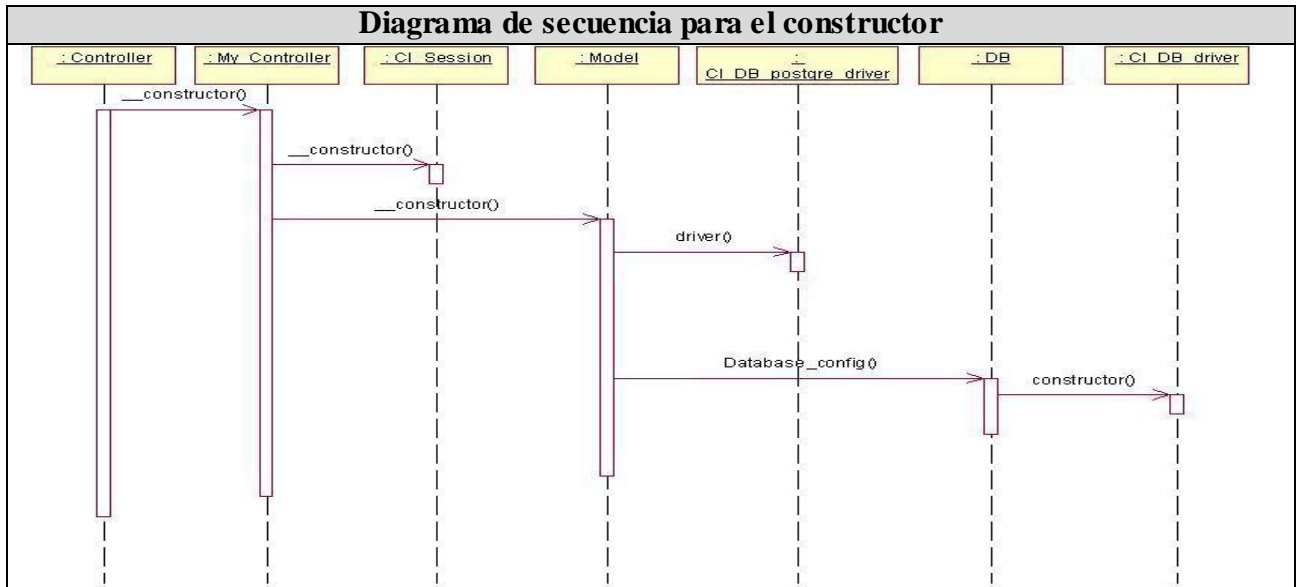


Imagen 11: Diagrama de secuencia para el constructor.

**3.4 Diseño del sistema**

El diseño de sistemas se ocupa de desarrollar las directrices propuestas durante el análisis, en términos de aquella configuración que tenga más posibilidades de satisfacer los objetivos planteados, tanto desde el punto de vista funcional como del no funcional.

**3.4.1 Diagramas de clases del diseño**

El Diagrama de Clase es uno de los principales artefactos generados en este flujo de trabajo. En él, al igual que en el análisis, la estructura de clases del sistema se especifica, con relaciones entre clases y estructuras de herencia, la única diferencia se centra en que en este nivel el principal objetivo es satisfacer los detalles de las implementaciones.

**3.4.1.1 Diagrama de clases Web**

Para modelar las aplicaciones Web no se usa la forma tradicional de modelar las clases, se usa varios mecanismos de extensión que fueron definidos para extender su semántica en aquellas situaciones en las que no es posible capturar todas las características de una determinada arquitectura o dominio en particular.



En esta tesis se elaboró un diagrama de clases Web para cada uno de los casos de uso del sistema. A continuación se mostrarán los diagramas de las clases del diseño del paquete de administración y Entrada de Datos, el resto se podrán ver desde el anexo 38 hasta el 43.

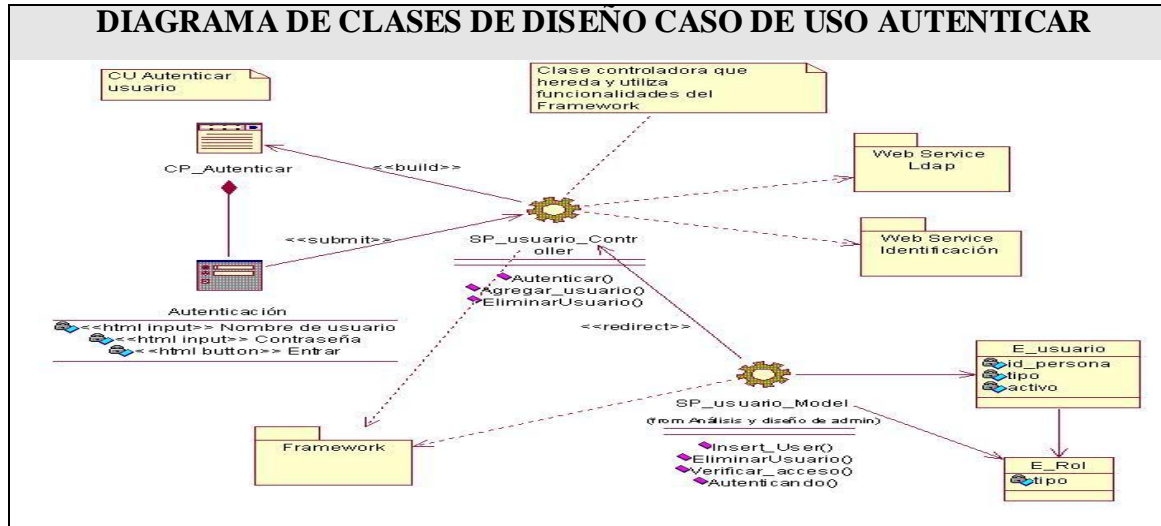


Imagen 12: Diagrama clases de diseño CU Autenticar

**DIAGRAMA DE CLASES DE DISEÑO CASO DE USO MOSTRAR  
EXPEDIENTE PRODUCTIVO**

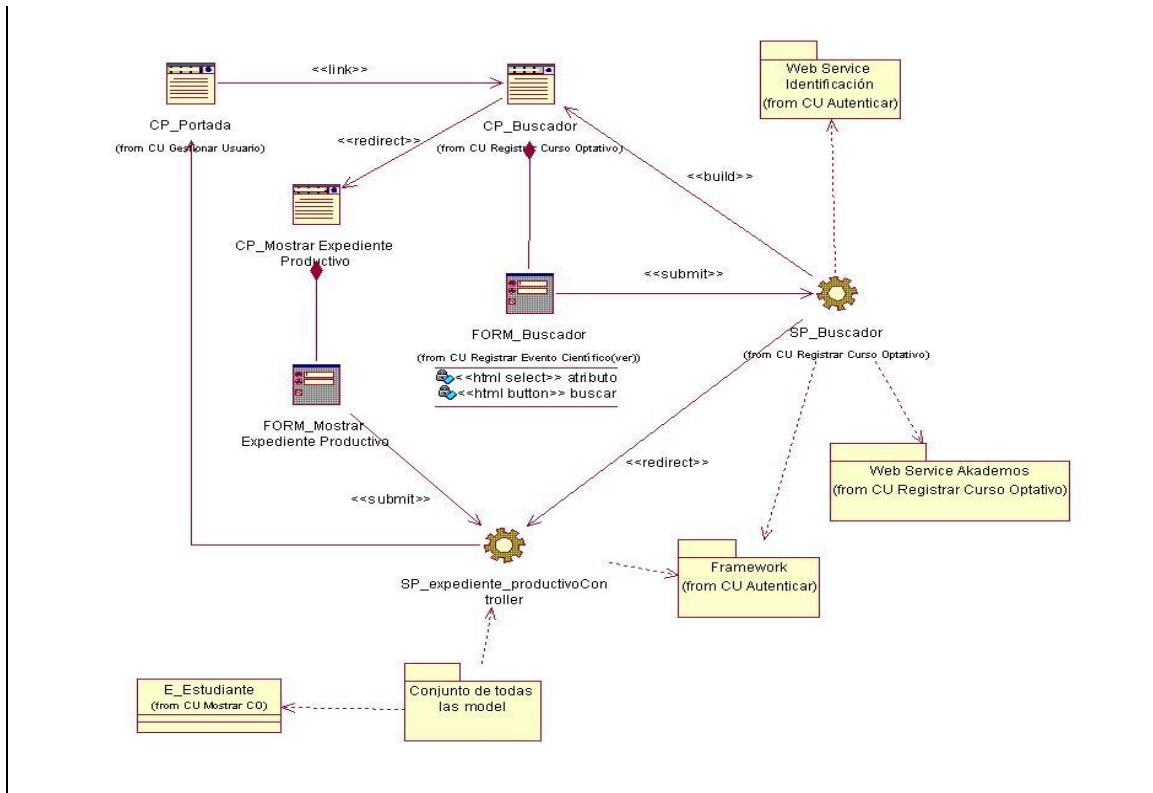


Imagen 13: Diagrama de clases de diseño CU mostrar expediente productivo.

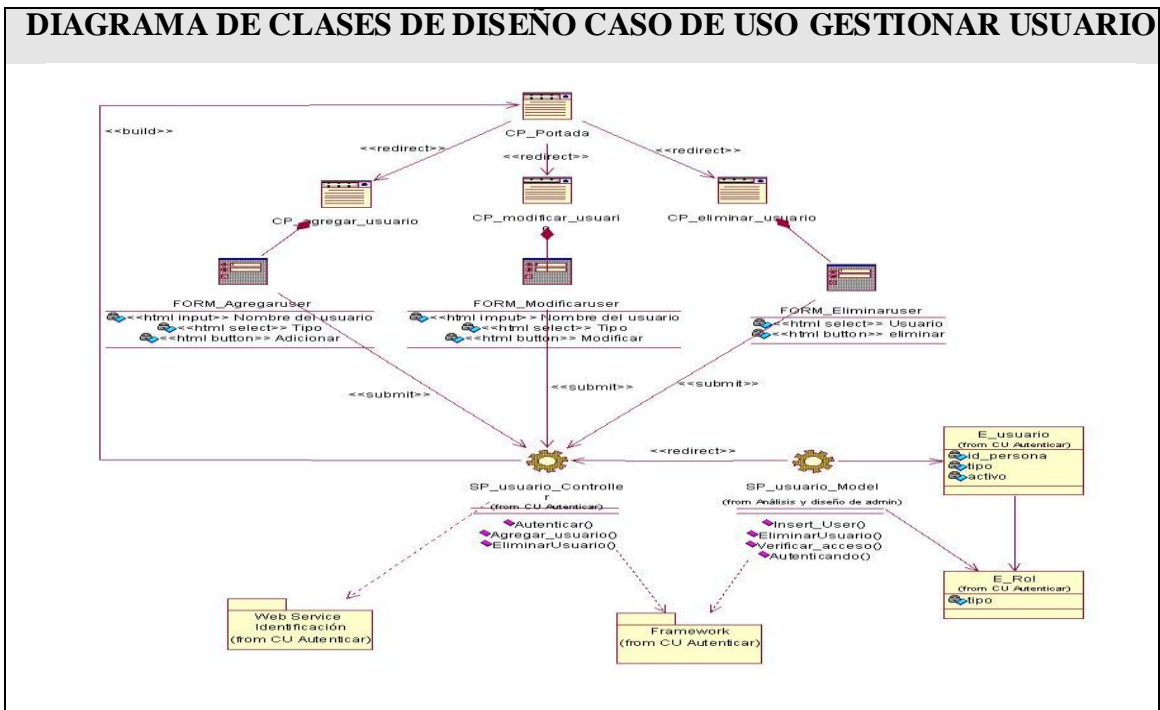


Imagen 14: Diagrama de clases de diseño CU Gestionar Usuario

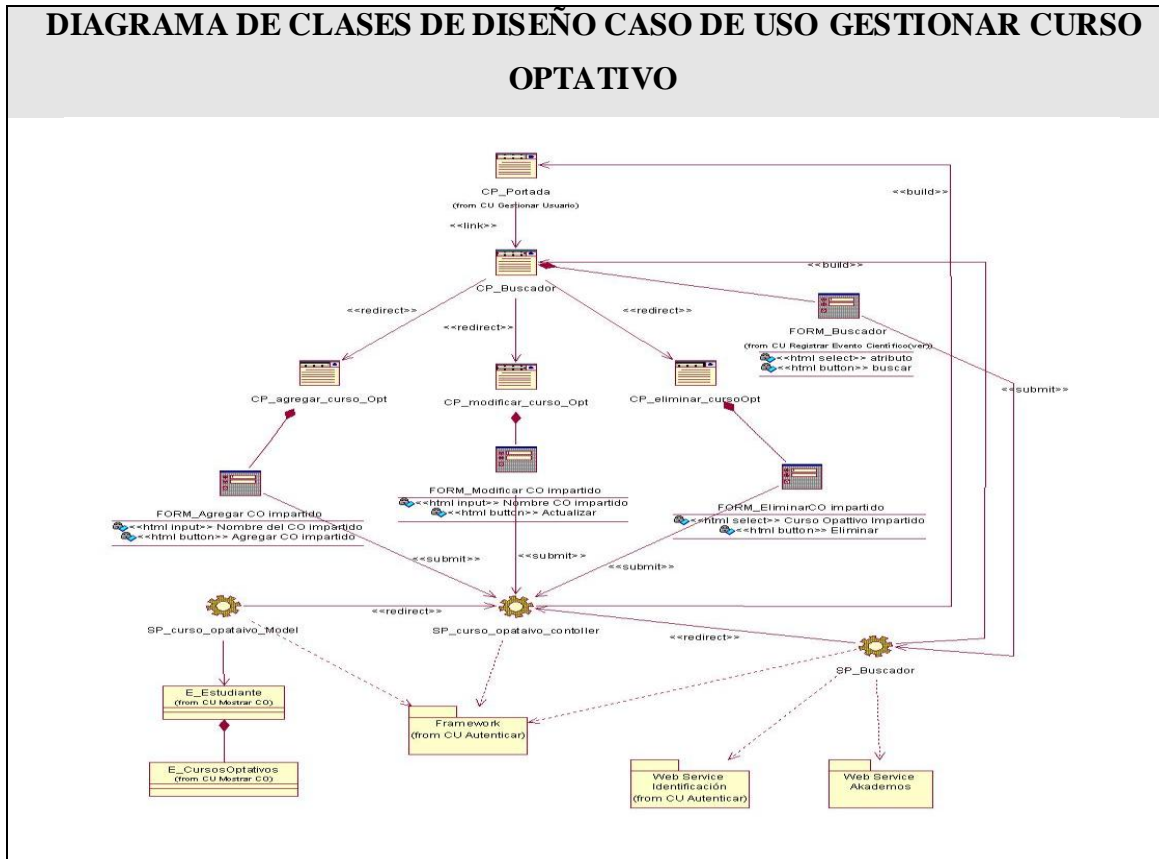


Imagen 15: Diagrama de clases de diseño de CU Gestionar Curso Optativo

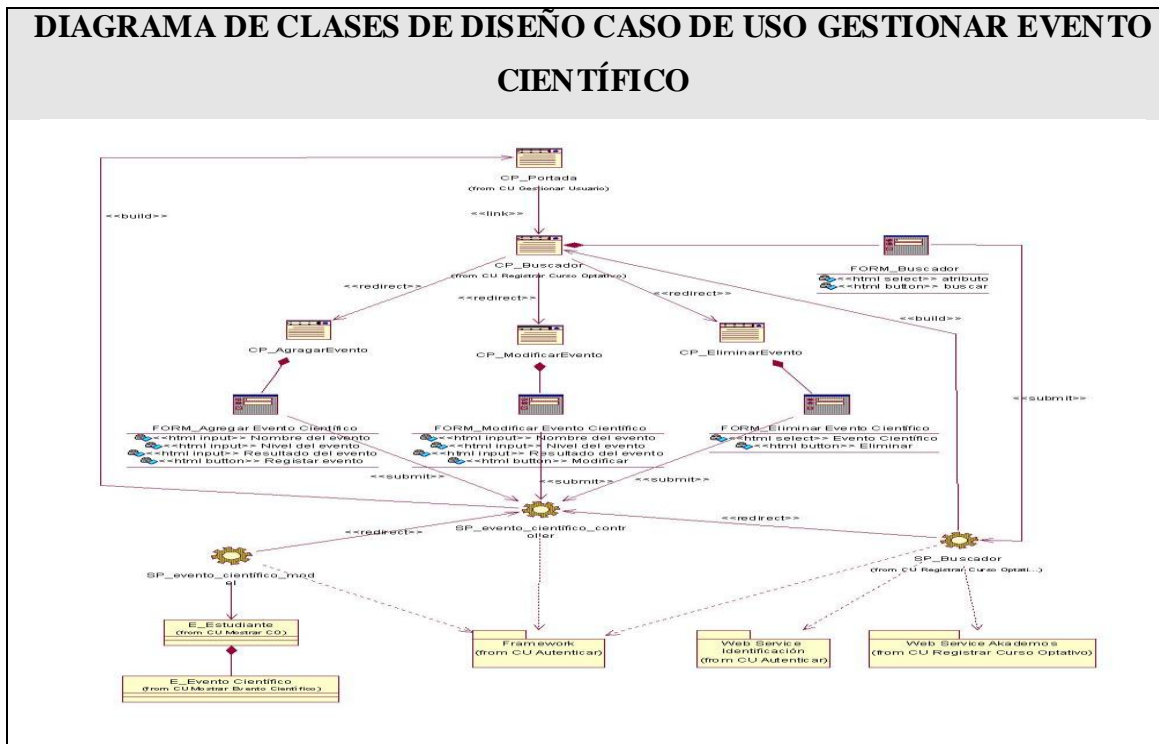


Imagen 16: Diagrama de clases de diseño CU Gestionar Evento Científico

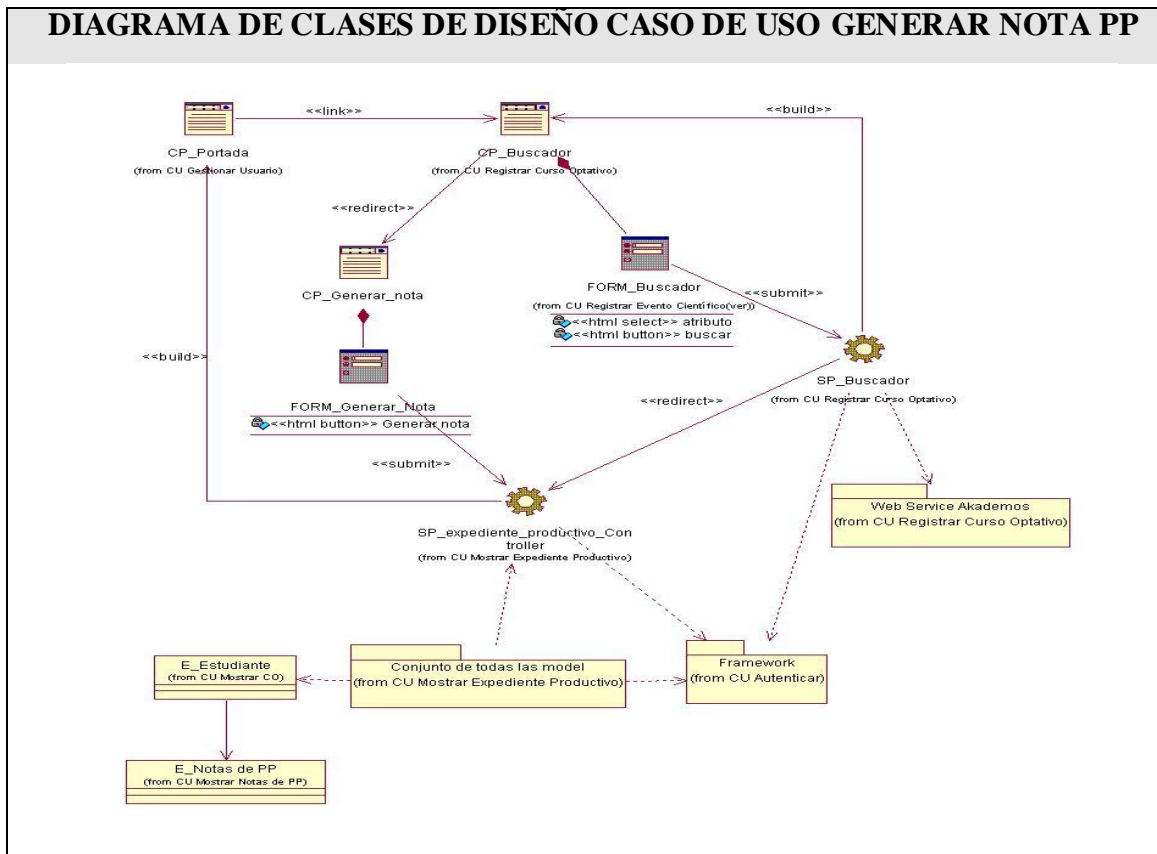


Imagen 17: Diagrama de clases de diseño CU Generar Nota PP

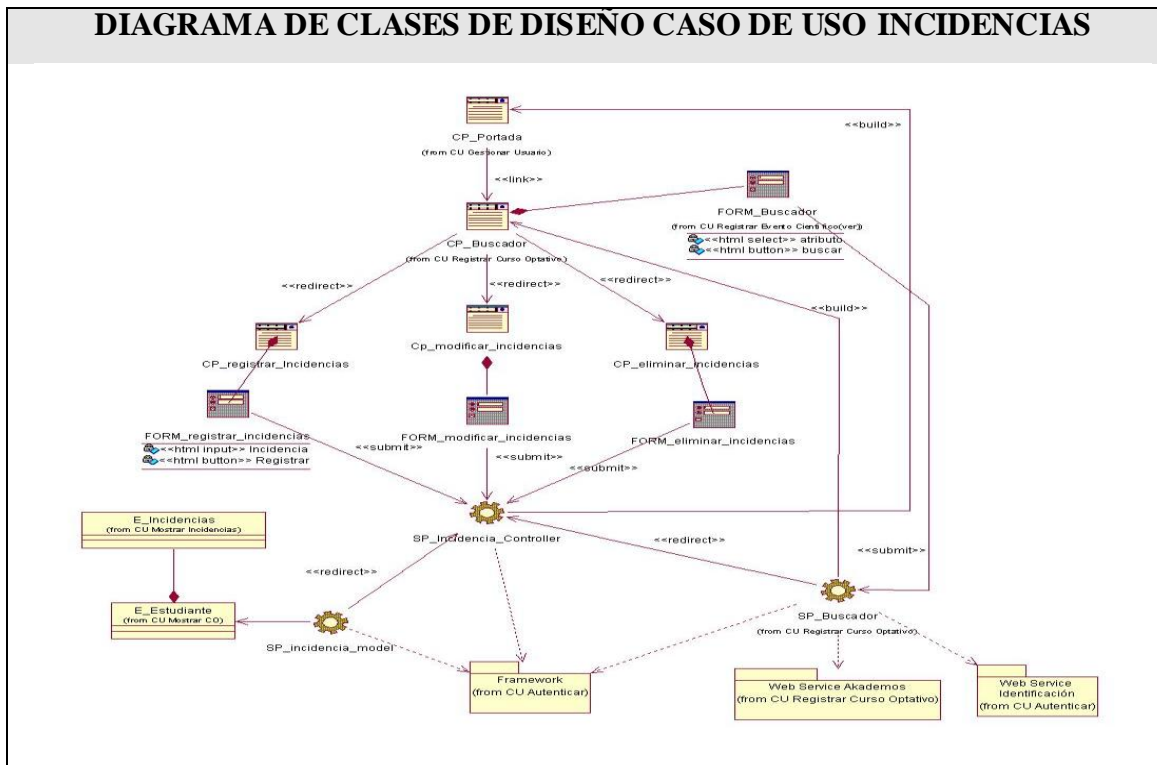


Imagen 18: Diagrama de clases de diseño CU Incidencias

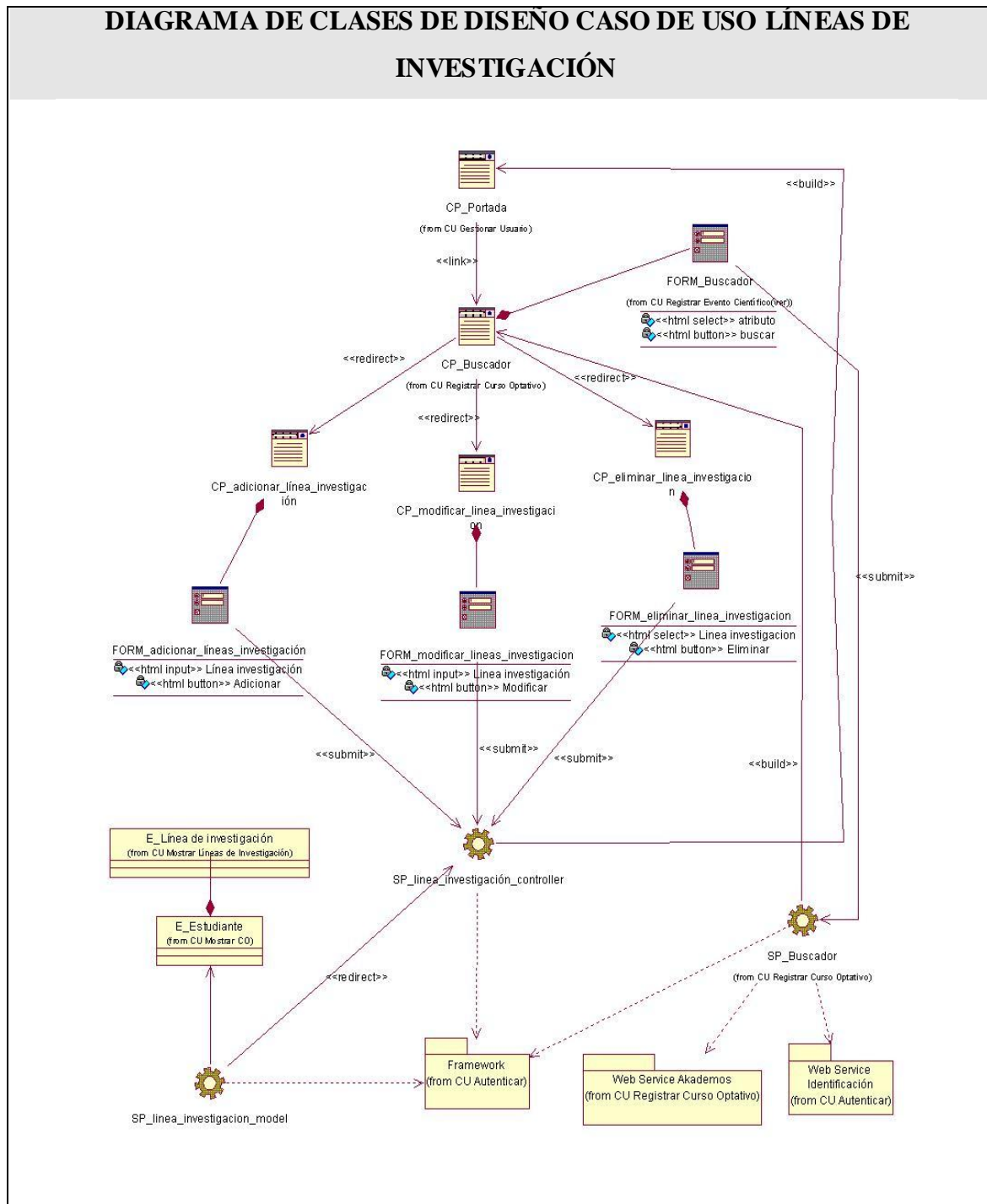


Imagen 19: Diagrama de clases de diseño CU Líneas de Investigación.

### 3.4.1.2 Principios de diseño

El diseño de la interfaz es uno de los puntos fundamentales a tratar a la hora de la presentación de la aplicación, teniendo en cuenta que es lo que ve el usuario y por lo tanto

debe ser lo más amigable y comprensible posible. Una aplicación con una interfaz bien diseñada debe tener, además de un buen diseño gráfico, una buena navegabilidad, usabilidad y distribución de los contenidos. Para el diseño de la interfaz de usuario de este sistema se han seguido los siguientes principios:

- ✓ Permitir su utilización por cualquier persona que tenga conocimientos básicos de informática.
- ✓ Permitir al usuario una interfaz atractiva y flexible facilitando exactitud y precisión.
- ✓ Los objetos que componen la interfaz den a entender al usuario de su existencia y su funcionalidad.
- ✓ Brindar a los usuarios una amplia legibilidad de la información esencial.
- ✓ Brindar a los usuarios avisos eficaces y métodos de respuesta durante y tras la finalización de la tarea, que elimine la complejidad innecesaria.

### **3.5 Arquitectura en capas**

El estilo de Arquitectura en Capas, define cómo organizar el modelo de diseño en capas, que pueden estar físicamente distribuidas, lo que significa que los componentes de una capa sólo pueden hacer referencia a componentes en capas inmediatamente inferiores.

#### **3.5.1 Modelo – Vista- Controlador (MVC)**

Utilizando la Arquitectura en capas es preciso mencionar el patrón de arquitectura de software MVC. El patrón MVC se ve frecuentemente en aplicaciones Web. Este patrón separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos.

El Modelo representa el proceso de abstracción para acceder al medio de persistencia de datos, en este se programan las clases que interactúan con la base de datos. La Vista no es más que la presentación del sitio, o sea, la parte exterior del mismo, en la práctica se recomiendan que sean ficheros HTML, al que se les puede colocar el código en el lenguaje del servidor. El Controlador es el coordinador de los recursos de software (Vistas y Modelos) para responder a cada una de las peticiones HTTP realizadas a la aplicación.

El MVC maximiza el rendimiento, la adaptabilidad y la simplicidad del producto en la menor cantidad de código y sencillez del mismo.





### **Conclusiones**

Con la ayuda de los diagramas propuestos por UML se ha logrado un mayor acercamiento a la programación del sistema propuesto, se modelaron los componentes como clases, utilizando las extensiones del UML, y se presentaron a través de diagramas de clases Web. Se ha analizado la arquitectura a utilizar y la implementación del patrón modelo vista controlador, a partir del refinamiento del diagrama conceptual se obtuvo un modelo físico de datos en correspondencia con los requerimientos planteados en el capítulo anterior. Todos estos elementos obtenidos son claves para la correcta implementación del sistema propuesto. Con esto se puede empezar a construir el sistema tratando de que se cumplan todos los requerimientos y las funciones que han sido consideradas necesarias.



### CAPÍTULO 4: IMPLEMENTACIÓN DEL SISTEMA

#### Introducción

La implementación, por su parte, toma los resultados del diseño e implementa el sistema en términos de componentes. En este capítulo se presentará el modelo de implementación, mediante los diagramas de componentes por paquetes y el de despliegue, que resultaron del diseño realizado de cada uno de los casos de uso del sistema planteado.

#### 4.1 Implementación

El Modelo de Implementación describe cómo se organizan y se relacionan los elementos del modelo del diseño, así cómo se implementan en términos de componentes, definiendo un componente como la parte modular de un sistema, desplegable y reemplazable que encapsula implementación y un conjunto de interfaces y proporciona la realización de los mismos. Un componente típicamente contiene clases y puede ser implementado por uno o más artefactos (ficheros ejecutables, binarios, etc.).

#### 4.2 Diagrama de componente

Un Diagrama de Componentes representa la separación de un sistema de software en componentes físicos (por ejemplo archivos, cabeceras, módulos, paquetes, etc.). El diagrama de componentes se usa para modelar la estructura del software, incluyendo las dependencias entre los componentes de software, los componentes de código binario, y los componentes ejecutables. En el diagrama de componentes se modelan componentes del sistema, a veces agrupados por paquetes, las dependencias que existen entre componentes y paquetes de componentes.

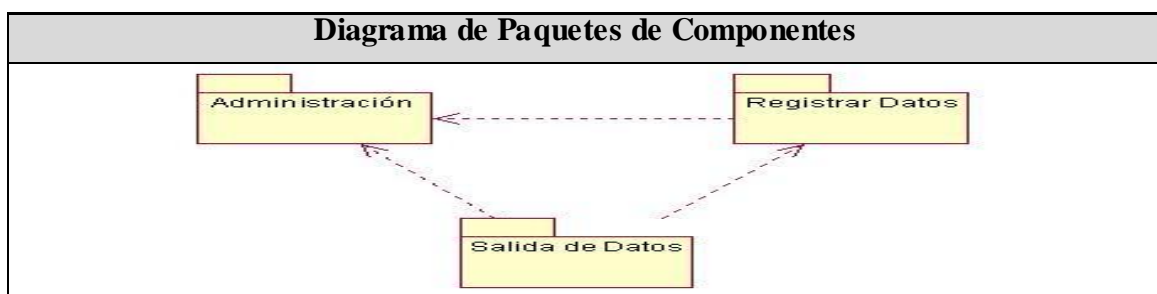


Imagen 21: Diagrama de paquetes de Componentes.

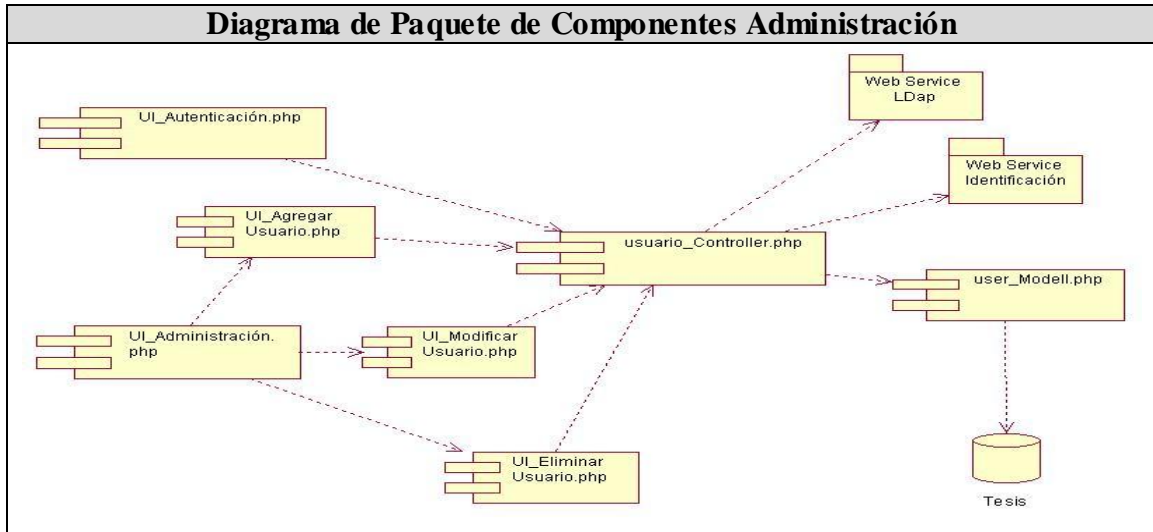


Imagen 22: Diagrama de Paquetes de Componentes Administración

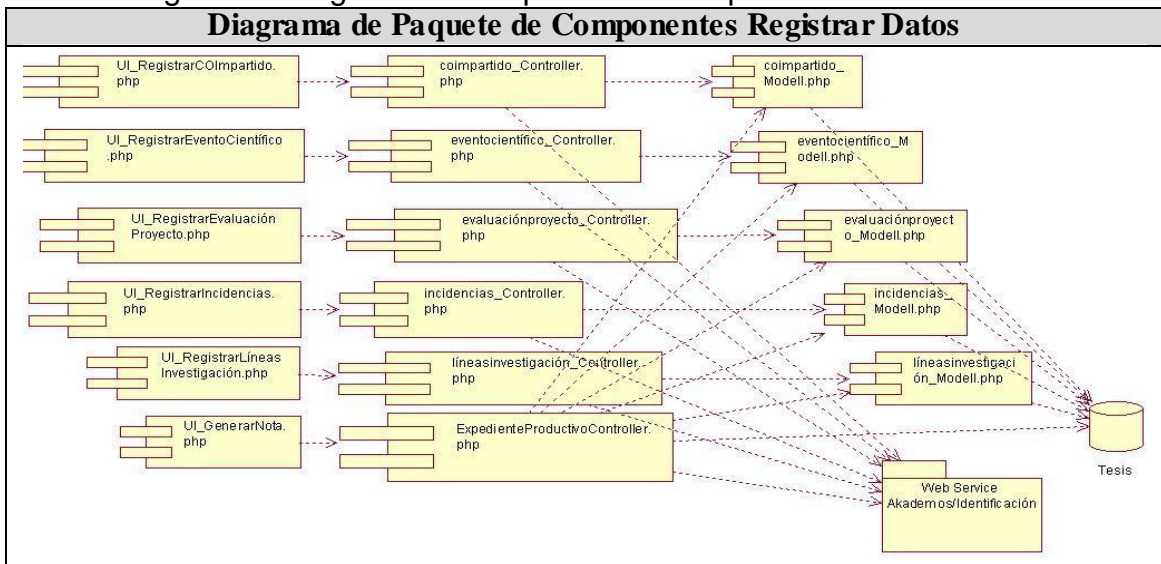


Imagen 23: Diagrama de Paquete de Componentes Registrar Datos

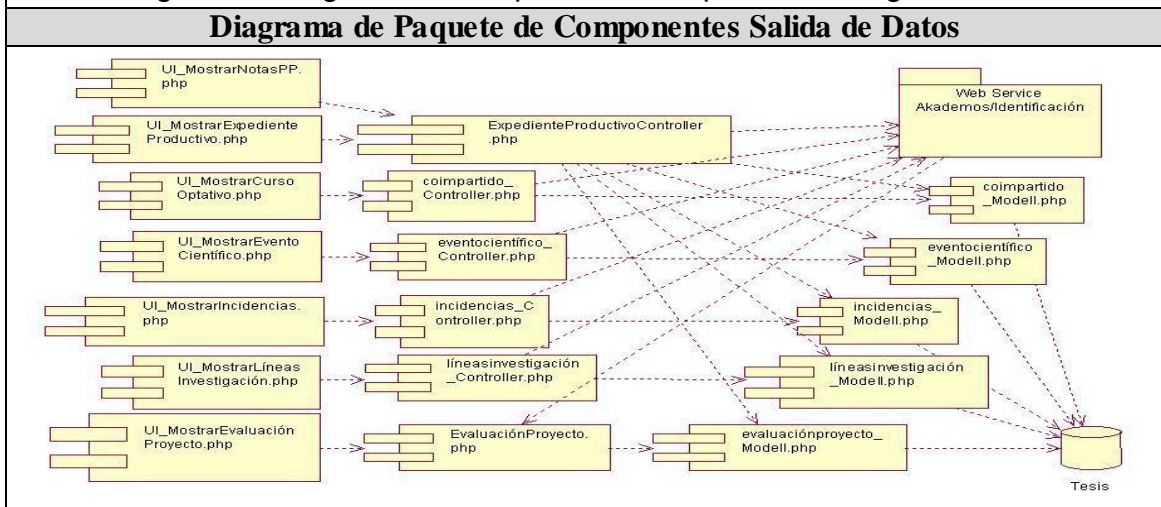


Imagen 24: Diagrama de Paquete de Componentes Salidas de Datos

### 4.3 Modelo de Despliegue

El modelo de despliegue describe la distribución física del sistema, muestra como están distribuidos los componentes de software entre los distintos nodos de cómputo. Permite comprender la correspondencia entre la arquitectura software y la arquitectura hardware.

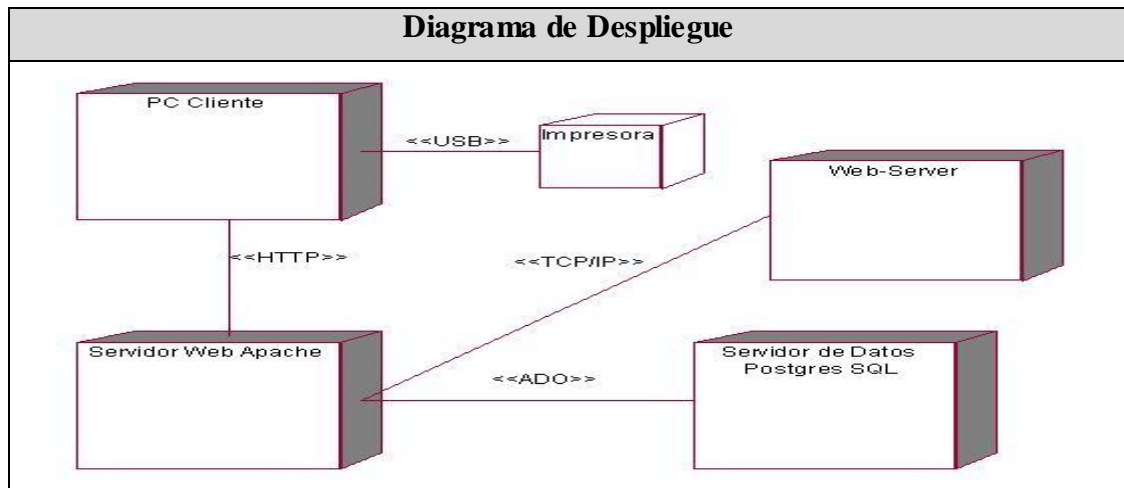


Imagen 25: Diagrama de Despliegue

### Conclusiones

Luego de haber pasado por casi todos los flujos que propone RUP, se ha logrado modelar el presente sistema, donde cualquier programador o estudiante con los mínimos conocimientos de ingeniería de software puede utilizar estos diagramas como referencia para continuar con la mejora del software, o simplemente como material de estudio, ahora solo queda implementar todo lo expuesto en el transcurso del documento para obtener el producto final.

**CAPÍTULO 5. ESTUDIO DE FACTIBILIDAD**

**Introducción**

Para la realización de un proyecto es de suma importancia el análisis del costo y los beneficios que reportará. Como resultado de este análisis se obtiene el tiempo de desarrollo en meses, costo y la cantidad de personas que se necesitan para desarrollar el proyecto. En este capítulo se describe la estimación de costos del sistema propuesto y sus beneficios.

**5.1 Planificación basada en casos de uso**

Pasos para la estimación:

**Paso 1. Calcular el Punto de Casos de Usos sin ajustar.**

$$UUCP = UAW + UUCW$$

Donde:

UUCP: Puntos de casos de uso sin ajustar.

UAW: Factor de peso de los actores sin ajustar.

UUCW: Factor de peso de los casos de uso sin ajustar.

Tipo de actor	Descripción	Factor de peso	Actores	Total
Simple	Sistema con sistema a través de interfaz de programación.	1	0	0
Medio	Sistema con sistema mediante protocolo de interfaz basada en texto.	2	0	0
Complejo	Persona que interactúa con el sistema mediante interfaz gráfica.	3	5	15
	Total		5	15

Tabla 19: Factor de peso de los actores sin ajustar

$$UAW = \sum cant \ actores * peso$$

$$UAW = 15$$

Tipo de CU	Descripción	Peso	Cantidad de CU	Total
Simple	El caso de uso tiene de 1 a 3 transacciones.	5	9	45
Medio	El caso de uso tiene de 4 a 7 transacciones.	10	6	60
Complejo	El caso de uso tiene más de 8 transacciones.	15	0	0
	Total		15	105

Tabla 20: Factor de peso de los CU sin ajustar.

$$UUCW = \sum cant\ CU * Pesc$$

$$UUCW = 105$$

$$UUCP = 15 + 105$$

$$UUCP = 120$$

**Paso 2. Calcular los Puntos de casos de uso ajustados.**

$$UCP = UUCP * TCF * EF$$

Donde:

UCP: Puntos de casos de uso ajustados.

UUCP: Puntos de casos de uso sin ajustar.

TCF: Factor de complejidad técnica.

EF: Factor de ambiente.

El factor de complejidad técnica (TCF) se calcula mediante la cuantificación de un conjunto de factores que determinan la complejidad técnica del sistema. Cada uno de los factores se cuantifica con un valor de 0 a 5, donde 0 significa un aporte irrelevante y 5 un aporte muy importante.

Factor	Descripción	Peso	Valor asignado	Total
T1	Sistema distribuido	2	0	0
T2	Tiempo de respuesta	1	4	4
T3	Eficiencia del usuario final	1	1	1
T4	Funcionamiento Interno complejo	1	3	3
T5	El código debe ser reutilizable	1	3	3
T6	Facilidad de instalación	0,5	4	2
T7	Facilidad de uso	0,5	5	2,5
T8	Portabilidad	2	5	10
T9	Facilidad de cambio	1	3	3
T10	Concurrencia	1	3	3
T11	Incluye objetivos especiales de seguridad	1	1	1
T12	Provee acceso directo a terceras partes	1	0	0
T13	Se requieren facilidades especiales de entrenamiento de usuarios	1	0	0
Total				32.5

Tabla 21: Factor de Complejidad Técnica

$$TCF = 0.6 + 0.01 * \sum (peso * valor \text{ asignado})$$

$$TCF = 0.6 + 0.01 * 32.5$$

$$TCF = 0.6 + 0.325$$

$$TCF = 0.925$$

El factor de ambiente (EF) está relacionado con las habilidades y el entrenamiento del grupo involucrado en el desarrollo del sistema. El cálculo del mismo es similar al cálculo del Factor

de complejidad técnica, es decir, se trata de un conjunto de factores que se cuantifican con valores de 0 a 5.

Factor	Descripción	Peso	Valor asignado	Total
E1	Familiaridad con el modelo de proyecto utilizado	1,5	2	3
E2	Experiencia en la aplicación	0,5	1	0,5
E3	Experiencia en la orientación a objetivos.	1	4	4
E4	Capacidad del analista líder.	0,5	3	1,5
E5	Motivación.	1	5	5
E6	Estabilidad de requerimientos	2	5	10
E7	Personal Part-Time	-1	0	0
E8	Dificultad del lenguaje de programación	-1	3	-3
Total				21

Tabla 22: Factor de Ambiente

$$EF = 1.4 - 0.03 * \sum (\text{peso} * \text{valor asignado})$$

$$EF = 1.4 - 0.03 * 21$$

$$EF = 1.4 - 0.63$$

$$EF = 0.77$$

$$UCP = UUCP * TCF * EF$$

$$UCP = 120 * 0.925 * 0.77$$

$$UCP = 85.47$$

**Paso 3. Estimación de esfuerzo a través de los puntos de casos de uso.**

$$E = UCP * CF$$

Donde:

E: Esfuerzo estimado en horas hombres.

UCP: Punto de casos de usos ajustados.

CF: Factor de conversión.

Para obtener el factor de conversión (CF) se cuentan cuantos valores de los que afectan el factor ambiente (E1...E6) están por debajo de la media (3), y los que están por arriba de la media para los restantes (E7, E8). Si el total es 2 o menos se utiliza el factor de conversión 20 Horas-Hombre / Punto de Casos de uso. Si el total es 3 o 4 se utiliza el factor de conversión 28 Horas-Hombre / Punto de Casos de uso. Si el total es mayor o igual que 5 se recomienda efectuar cambios en el proyecto ya que se considera que el riesgo de fracaso del mismo es demasiado alto.

En este caso se puede decir que:

CF = 20 Horas-Hombre / Punto de Casos de uso.

$$E = 85.47 * 20$$

$$E = 1709.4 \text{ Horas Hombre}$$

#### Paso 4. Calcular esfuerzo de todo el proyecto.

Actividad	Porcentaje %	Horas-Hombres
Análisis	10	427.35
Diseño	20	854.7
Implementación	40	1709.4
Pruebas	15	641.025
Sobrecarga (otras actividades)	15	641.025
Total	100	4273.5

Tabla 23: Esfuerzo del proyecto

Si  $E_T = 4273.5$  horas-hombre y se estima que cada mes tiene como promedio 192 horas laborables, eso daría un  $E_T = 22.2578125$  mes-hombre. Esto quiere decir que 1 persona puede realizar el problema analizado en 20 meses y medio aproximadamente.



### **-Costo del Proyecto.**

Se asume como salario promedio mensual \$100.00

CH: Cantidad de hombres.

Tiempo: Tiempo total del proyecto.

CH = 2 hombres

CHM = 2 \* Salario Promedio

CHM = 200.00 \$/mes

Costo = CHM \*  $E_T$  / CH

Costo = 200.00 \* 22.2578125 / 2

**Costo = \$ 2225.78125 ≈ \$2225.78**

Tiempo =  $E_T$  / CH

Tiempo = 22.2578125 / 2

**Tiempo = 11.12890625 ≈ 11.12 meses**

De los resultados obtenidos se interpreta que con 2 hombres trabajando en el proyecto el mismo se desarrolla en 11.12 meses y su costo total se estima que sea \$2225.78

### **5.2 Beneficios tangibles e intangibles**

El Sistema para la informatización del Expediente Productivo e Investigativo del Estudiante, no es un producto con fines comerciales, su principal objetivo es resolver los problemas que existen en la gestión de la información productiva e investigativa de cada uno de los estudiantes que pertenecen a la Facultad 2.

Contar con una aplicación Web flexible, dinámica y de interfaz agradable que le permita registrar, actualizar y conocer de una forma más precisa y en el menor tiempo posible la información, referente a la producción e investigación de cada uno de los estudiantes, es el beneficio fundamental del sistema.

Por tanto, los beneficios inmediatos son generalmente intangibles:

- ✓ Disminución del tiempo y esfuerzo que se invierte en esta tarea que se realiza, hasta ahora, de forma manual.
- ✓ Disminución de la acumulación de grandes volúmenes de información, almacenados en formato duro.

- ✓ Disminución de los gastos al ser menos costoso crear y procesar información digital que copias duras.
- ✓ Fácil y rápido acceso y publicación de la información actualizada.
- ✓ Fácil procesamiento de la información.

### 5.3 Análisis de costos y beneficios

Desarrollar un producto informático cuesta. Justificar entonces su desarrollo depende de los beneficios que reportarían su implantación y utilización. Los beneficios pueden ser económicos y de orden social, estos últimos son de tanta importancia como los primeros. El sistema propuesto está dirigido a la Facultad 2, para la gestión de la actividad productiva e investigativa de los estudiantes, por tanto su mayor beneficio es de orden social.

Una vez implantado el sistema éste contribuirá a aumentar la eficiencia en la gestión de la información, al disminuir el tiempo necesario en el registro, consulta y actualización de la información.

La tecnología utilizada para el desarrollo del sistema es totalmente libre, por tanto no hay que incurrir en gastos en el pago de licencias de uso. El sistema es portable por lo que un cambio de plataforma para la implantación del mismo es viable y factible, y no hay que incurrir en muchos cambios; debido a la estructuración en capas de los procesos del negocio que se diseñaron.

Analizando el costo del proyecto, los numerosos beneficios que reporta, detallados con anterioridad, se puede concluir que su implementación es realmente factible.

### Conclusiones

En este capítulo se describió el estudio de factibilidad realizado correspondiente al sistema propuesto, teniendo en cuenta el costo estimado y los beneficios que reportará al ser implantado. La herramienta propuesta reportará beneficios significativos e importantes para la gestión del Expediente Productivo e Investigativo del Estudiante de la Facultad 2, al contribuir a mantener centralizada la información productiva e investigativa de cada estudiante, lo que indica que es factible implementar la herramienta propuesta.

### CONCLUSIONES

Con este trabajo se logró el objetivo propuesto: desarrollar una aplicación capaz de informatizar la gestión del Expediente Productivo e Investigativo del Estudiante, mediante el empleo adecuado de las TIC. Este sistema facilitará el flujo de información de la actividad productiva e investigativa del estudiante, asegurando su almacenamiento y haciendo su transmisión más rápida.

Se estudió y analizó cómo se realiza el seguimiento, control y asignación de las actividades de cada estudiante y la obtención de la nota de la asignatura de Práctica Profesional. Se estudiaron las posibles herramientas de Software Libre para el desarrollo de este sistema. Se desarrolló la aplicación siguiendo la metodología RUP, y se utilizaron representaciones UML para la modelación del proyecto.

Se diseñó e implementó una base de datos capaz de almacenar de manera organizada la información referente al Expediente Productivo e Investigativo del Estudiante. Se realizó el análisis y diseño e implementación del sistema automatizando los procesos necesarios para una correcta y eficiente gestión del Expediente Productivo e Investigativo del Estudiante.

Finalmente, se analizaron los beneficios y costos que generaría el sistema y se concluyó que desarrollar el proyecto es factible.

### RECOMENDACIONES

1. Someter a prueba el sistema durante un período de tiempo significativo, para confirmar su desempeño y que se correspondan las funcionalidades del sistema con lo que se espera.
2. Continuar añadiéndole nuevas funcionalidades al sistema.
3. Proponer la utilización y generalización de este sistema en la UCI para la gestión del Expediente Productivo e Investigativo del Estudiante, luego de obtener resultados satisfactorios en la Facultad 2.

### BIBLIOGRAFÍA

- [1]. El Proceso Unificado de Desarrollo de Software. Volumen 1.
- [2]. Larman, C. UML y Patrones. Introducción al análisis y diseño orientado a objetos. Ingeniería del Software 2.
- [3]. Lancillota Analía, Arquitectura Cliente/Servidor, Definición (Extraído el 25-nov)  
<http://www.mastermagazine.info/termino/4294.php>
- [4]. Empresa Consultora Externa, Arquitectura Cliente/Servidor Ventajas y Desventajas (Extraído el 25-nov-07) <http://www.csi.map.es/csi/silice/Global75.html>
- [5] El Sistema Operativo GNU (Consultado el 25-dic-07)  
[http://64.233.169.104/search?q=cache:tA-CfKGbpQAJ:www.gnu.org/home.es.html+software+libre&hl=es&ct=clnk&cd=2&gl=cu&lr=lang\\_es](http://64.233.169.104/search?q=cache:tA-CfKGbpQAJ:www.gnu.org/home.es.html+software+libre&hl=es&ct=clnk&cd=2&gl=cu&lr=lang_es)
- [6]. Del Castillo San Félix Alvaro, Desarrollador y admin de software libre, Barrapunto, [acs@barrapunto.com](mailto:acs@barrapunto.com), Servidor Web Apache (Extraído 24-nov-07),  
<http://acs.barrapunto.org/articulos/trunk/LinuxActual/Apache/apache.pdf>
- [7]. James Garrett [Jesse](#) Ajax: Un nuevo acercamiento a las aplicaciones Web (Extraído el 25-nov-07 ), <http://www.maestrosdelweb.com/editorial/ajax/>
- [8]. Alvarez [Miguel Angel](#), Director de DesarrolloWeb.com, Introducción a javascript (Extraído el 11-dic-07), <http://www.desarrolloweb.com/articulos/490.php>
- [9]. Van Der Henst S [Christian](#). Introducción al PHP, (publicación 23.05.2006) (Extraído el 256-nov-07) <http://www.maestrosdelweb.com/editorial/phpintro/>
- [10]. Framework codeIgniter, <http://codeigniter.com/> (Extraído 15-1-2008)
- [11]. [Alvarez](#) Sara, Equipo DesarrolloWeb.com, Sistema Gestor de Base de Datos, <http://www.desarrolloweb.com/articulos/sistemas-gestores-bases-datos.html>, (Extraído 20-2-2008)
- [12]. Carlos D. González, Base de Datos PostgreSQL, SQL avanzado y PHP, ( Extraído 2008)<http://www.usabilidadweb.com.ar/postgre.php>
- [13]. Van Der Henst S Christian, Las novedades de Dreamweaver 8,( publicación 08.08.2005), (Extraído 1-2-2008)

[http://64.233.169.104/search?q=cache:dJJPxMq7ZpoJ:www.maestrosdelweb.com/actualidad/2334/+Dreamweaver+8&hl=es&ct=clnk&cd=7&gl=cu&lr=lang\\_es](http://64.233.169.104/search?q=cache:dJJPxMq7ZpoJ:www.maestrosdelweb.com/actualidad/2334/+Dreamweaver+8&hl=es&ct=clnk&cd=7&gl=cu&lr=lang_es)

[14]. Yanover David , Director de MasterMagazine, Adobe Photoshop CS, (Extraído 1-2-2008)  
[http://64.233.169.104/search?q=cache:IYSw9pjo6fAJ:www.desarrolloweb.com/articulos/1905.php+Adobe+Photoshop+CS&hl=es&ct=clnk&cd=7&gl=cu&lr=lang\\_es](http://64.233.169.104/search?q=cache:IYSw9pjo6fAJ:www.desarrolloweb.com/articulos/1905.php+Adobe+Photoshop+CS&hl=es&ct=clnk&cd=7&gl=cu&lr=lang_es)

[15]. Yanover David , Director de MasterMagazine, Zend Studio, (Extraído 11-1-2008)  
[http://64.233.169.104/search?q=cache:aJIZJhDicoAJ:www.desarrolloweb.com/articulos/1178.php+zend+studio&hl=es&ct=clnk&cd=1&gl=cu&lr=lang\\_es](http://64.233.169.104/search?q=cache:aJIZJhDicoAJ:www.desarrolloweb.com/articulos/1178.php+zend+studio&hl=es&ct=clnk&cd=1&gl=cu&lr=lang_es)

[16]. <http://www.rational.com.ar/herramientas/roseenterprise.html>, (Extraído 21-3-2008) y ayuda de Rational.

[17]. Ayuda de PgAdmin, SQL Manager 2005 for PostgreSQL.

[18]. Ayuda del Case Studio, ucistore.

[19]. Sanchez, M. A. Metodologías De Desarrollo De Software (07 de 06 de 2004). Informatizate. Recuperado el 24 de 11 de 2007, de Informatizate:  
[http://www.informatizate.net/articulos/metodologias\\_de\\_desarrollo\\_de\\_software\\_07062004.html](http://www.informatizate.net/articulos/metodologias_de_desarrollo_de_software_07062004.html)

[20]. Herramientas CASE, <http://www.csi.map.es/csi/silice/Sgcase5.html>

[21]. Molpeceres Alberto, Procesos de desarrollo: RUP, XP y FDD (Visitado 24-nov-07)  
<http://www.willydev.net/descargas/articulos/general/cualxpfdrrup.PDF>

[22]. Navegador Firefox: Mucho mejor que IE (Extraído el 25-nov-07)  
<http://www.wilkinsonpc.com.co/free/firefox.html>

[23]. Pressman, R. Ingeniería del Software. Un enfoque práctico. Parte 1.

[24]. Pressman, R. Ingeniería del Software. Un enfoque práctico. Parte 2.

[25]. PINCIROLI, F. Diagramas de interacción Di tutto il Mondo, 2001: 25.

[pág 70 ]. FERRER, F. Diseño de Sistemas 2006. [2007]. Disponible en:  
<http://www.daedalus.es/AreasISDiseno-E.php>

[26]. BARRERA Presente y futuro de los SI ESTR@TEGIAMagazine, 2006, 1: 23.

[26]. RUP: Rational Unified Process (Proceso Unificado de desarrollo). Metodología para el desarrollo de Software.

[27]. Aguilar Vicente y Suau Pablo , PostgreSQL, (Extraído 15-1-2008) <http://www.fedora-es.com/node/189>

[28]. Pérez Valdés Damián, Los Frameworks de PHP agilizan tu trabajo II, (Extraído 5-2-2008) <http://www.maestrosdelweb.com/editorial/los-frameworks-de-php-agilizan-tu-trabajo-ii/>

[29]. EL PARADIGMA CUANTITATIVO DE LA INVESTIGACIÓN CIENTÍFICA, Rolando Alfredo Hernández León y Sayda Coello González, Universidad de las Ciencias Informáticas, Ciudad de la Habana, Noviembre del 2002

## ANEXOS

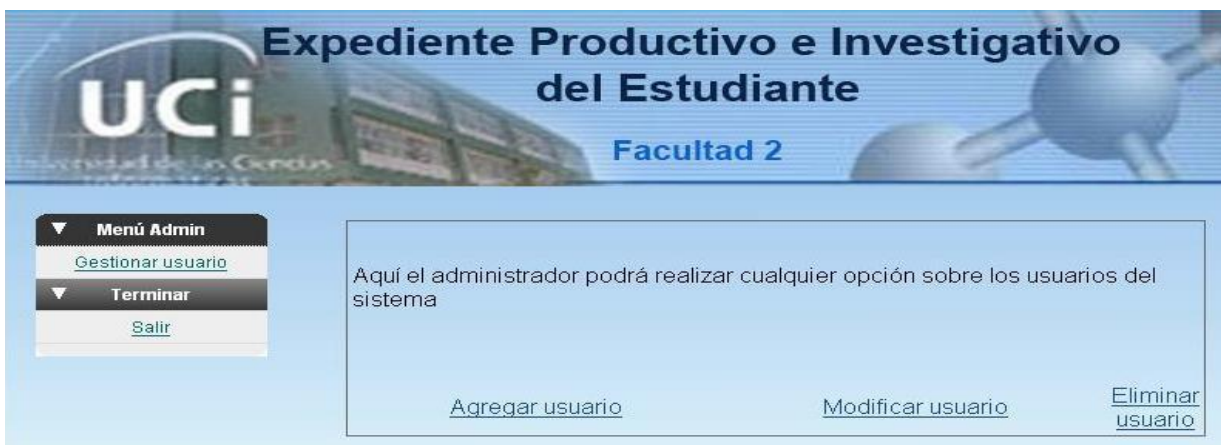
### Anexo 1: Autenticarse



Usuario:

Contraseña:

### Anexo 2: Gestionar Usuario



**UCi** Universidad de La Guayana Facultad 2

**Expediente Productivo e Investigativo del Estudiante**

**Menú Admin**

- Gestionar usuario

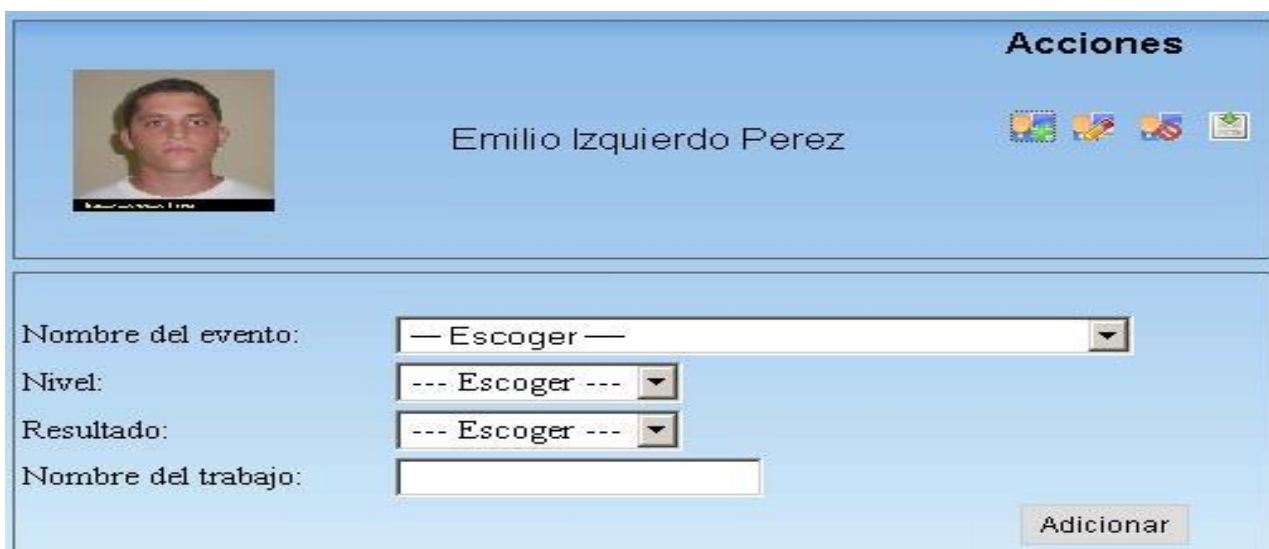
**Terminar**

- Salir


Aquí el administrador podrá realizar cualquier opción sobre los usuarios del sistema

[Agregar usuario](#) [Modificar usuario](#) [Eliminar usuario](#)

### Anexo 3: Gestionar Evento Científico



**Acciones**

 Emilio Izquierdo Perez

Nombre del evento:

Nivel:

Resultado:

Nombre del trabajo:



### Anexo 4: Gestionar Curso Optativo

**Acciones**

Luis Carlos Aguilar Panoque

Nombre del curso	Curso académico
<input type="checkbox"/> SQL	07-08
<input type="checkbox"/> php	07-08

**Eliminar**

### Anexo 5: Gestionar Línea Investigación

**Acciones**

Emilio Izquierdo Perez

**Nombre**

- Polo de Auditoría y Control
- Polo de Seguridad Informática y Redes
- Polo de Telecomunicaciones y Telemática

**Mostrar Líneas**

**Nombre**

- Redes de nueva generación
- Estudios de herramientas antivirus en GNU/Linux

**Adicionar**

### Anexo 6: Gestionar Incidencias

**Acciones**

Emilio Izquierdo Perez

Se ha insertado la Incidencia

**Incidencia**

No entregó las tareas programadas a tiempo.

**Adicionar**

**Anexo 7: Gestionar Evaluación del Proyecto**

<b>Rol</b>	<b>Evaluación</b>
<input type="text" value="- Escoger -"/>	<input type="text" value="- Escoger -"/>
<input type="button" value="Adicionar"/>	
<b>Otros Aspectos</b>	<b>Evaluación</b>
Disciplina tecnológica	<input type="text" value="- Escoger -"/>
Asistencia	<input type="text" value="- Escoger -"/>
Puntualidad	<input type="text" value="- Escoger -"/>
Responsabilidad laboral	<input type="text" value="- Escoger -"/>
Entrega de información en tiempo	<input type="text" value="- Escoger -"/>
Calidad de la información entregada	<input type="text" value="- Escoger -"/>
<input type="button" value="Adicionar"/>	

**Anexo 8: Generar Nota de PP**

<b>Grupo:</b> <input type="text" value="02501"/>
<input type="button" value="Buscar"/>
<b>Listado de Estudiantes</b>
<input checked="" type="checkbox"/> Luis Carlos Aguilar Paneque
<input checked="" type="checkbox"/> Carlos Guillen Chang
<input checked="" type="checkbox"/> Yunior Ernesto Guerrero Chacón
<input checked="" type="checkbox"/> Gerlin Vidal Rodriguez
<input checked="" type="checkbox"/> Yordan Ernesto Estrada Rodriguez
<input checked="" type="checkbox"/> Lisbet Bertoli Ramos
<input checked="" type="checkbox"/> Sheila Diaraye Sylla Martín
<input checked="" type="checkbox"/> Yailín Valdivia Linares
<input checked="" type="checkbox"/> Marilyn Amador Medero
<input checked="" type="checkbox"/> Yurima Cabrera Sarrión
<input checked="" type="checkbox"/> Milagro Leonor Leyva Fonseca
<input checked="" type="checkbox"/> Arianna Ortega Castillo
<input checked="" type="checkbox"/> Yusmilaidi Causse Ascanio
<input checked="" type="checkbox"/> Daydier Bernal Calvo
<input checked="" type="checkbox"/> Yojandy Batista Chillón
<input checked="" type="checkbox"/> Surima Gé Pérez
<input checked="" type="checkbox"/> Leonel Soto Rodríguez

## Anexo 9: Mostrar Expediente Productivo

<b>EXPEDIENTE PRODUCTIVO E INVESTIGATIVO DEL ESTUDIANTE</b> <b>Universidad de las Ciencias Informáticas</b> <b>Curso:07-08</b>			
<b>Nombre del estudiante:</b> Luis Carlos Aguilar Paneque <b>Solapín:</b> 50837			
<b>Eventos Científicos</b>			
<b>Nombre del trabajo</b>	<b>Evento realizado</b>	<b>Nivel</b>	<b>Resultado</b>
Martí en Nosotros	Seminario Juvenil Martiano	Internacional	Participante
Copa Pascal	Copas y Olimpiadas	Mención	Destacado
<b>Cursos Optativos Impartidos</b>			
<b>Nombre del curso impartido</b>			
SQL			
php			
<b>Proyecto</b>			
No tiene información registrada			
<b>Línea de Investigación</b>			
<b>Polos de la Facultad</b>		<b>Líneas de Investigación</b>	
Polo de Seguridad Informática y Redes		Estudios de herramientas antivirus en GNU/Linux	
<b>Incidencia</b>			
<b>Fecha de registro</b>			
2008-06-14 16:21:33			
NOTA DE PP:5			
<a href="#">Atrás</a> 			

## Anexo 10: Mostrar Evento Científico




Luis Carlos Aguilar Paneque

Acciones




Nombre del trabajo	Nombre del evento	Nivel	Resultado	Curso
Martí en Nosotros	Seminario Juvenil Martiano	Internacional	Participante	07-08
Copa Pascal	Copas y Olimpiadas	Mención	Destacado	07-08

## Anexo 11: Mostrar Cursos optativo Impartidos




Luis Carlos Aguilar Paneque

Acciones




Nombre del curso	Curso académico
SQL	07-08
php	07-08

## Anexo 12: Mostrar Línea de Investigación



Luis Carlos Aguilar Paneque


Acciones




Líneas de investigación	Polos de la facultad	Curso académico
Estudios de herramientas antivirus en GNU/Linux	Polo de Seguridad Informática y Redes	07-08

### Anexo 13: Mostrar Incidencias

Acciones



Luis Carlos Aguilar Paneque




Fecha	Curso académico
2008-06-14 16:21:33	07-08

### Anexo 14: Mostrar Evaluación de Proyecto


Nombre del Proyecto	Tipo de Proyecto
Rol	Evaluación
Otros Aspectos	
Disciplina tecnológica	
Asistencia	
Puntualidad	
Responsabilidad laboral	
Entrega de información en tiempo	
Calidad de la información entregada	

### Anexo 15: Mostrar Nota de PP

Acciones

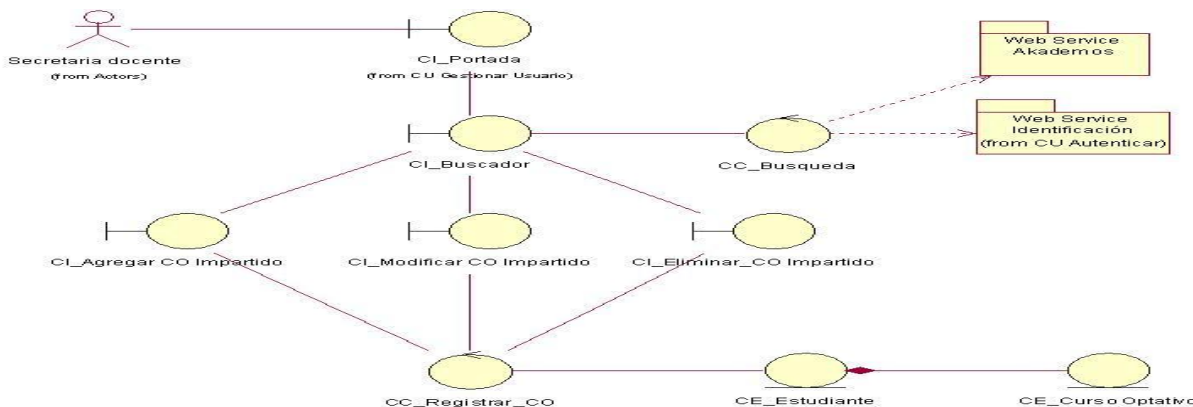


Luis Carlos Aguilar Paneque

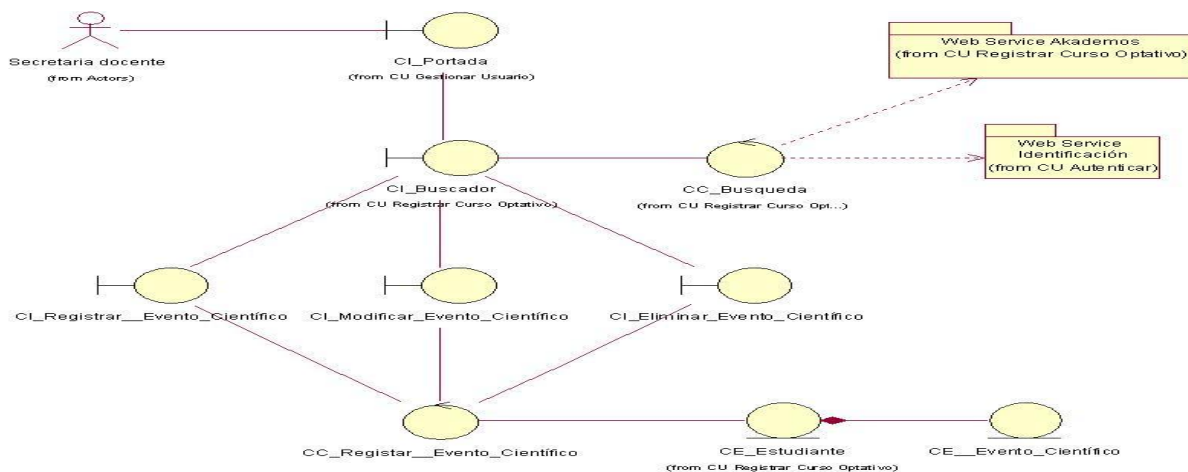


La nota de PP es: 5

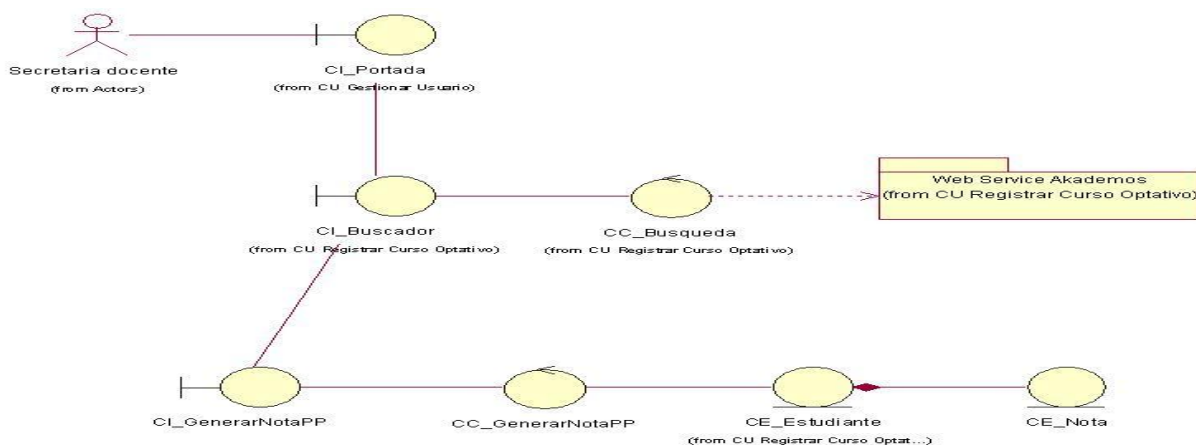
### Anexo 16: Clase de análisis del CU Registrar Cursos Optativo



### Anexo 17: Clase de análisis del CU Registrar Evento Científico

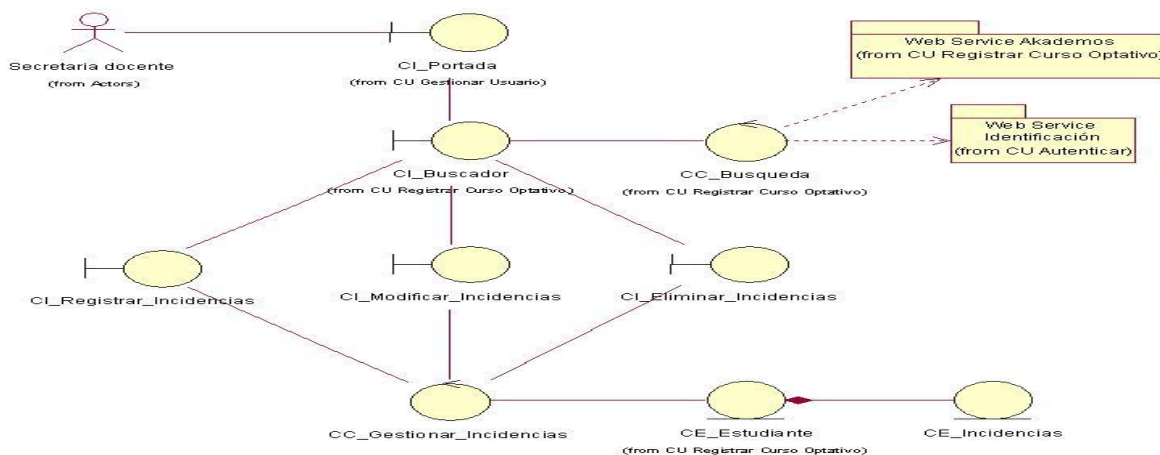


### Anexo 18: Clase de análisis del CU Generar Nota de PP

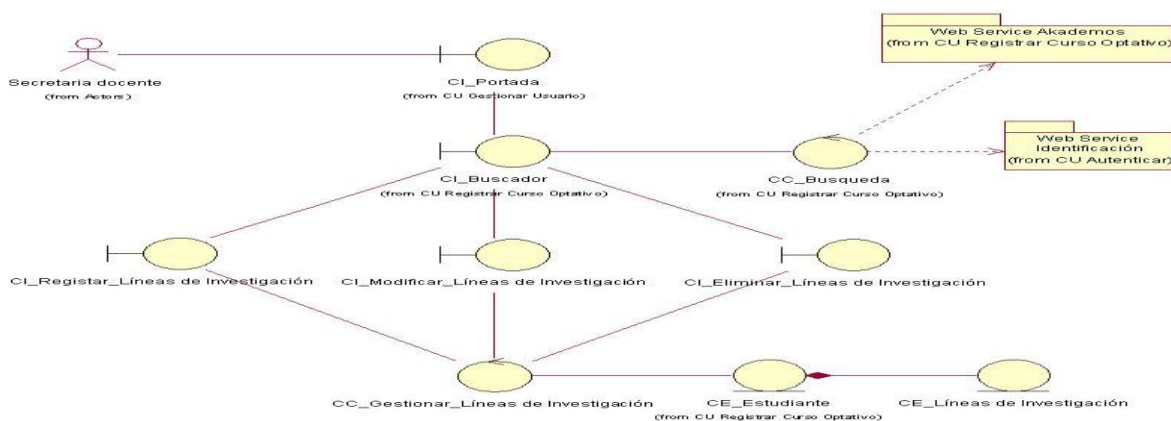




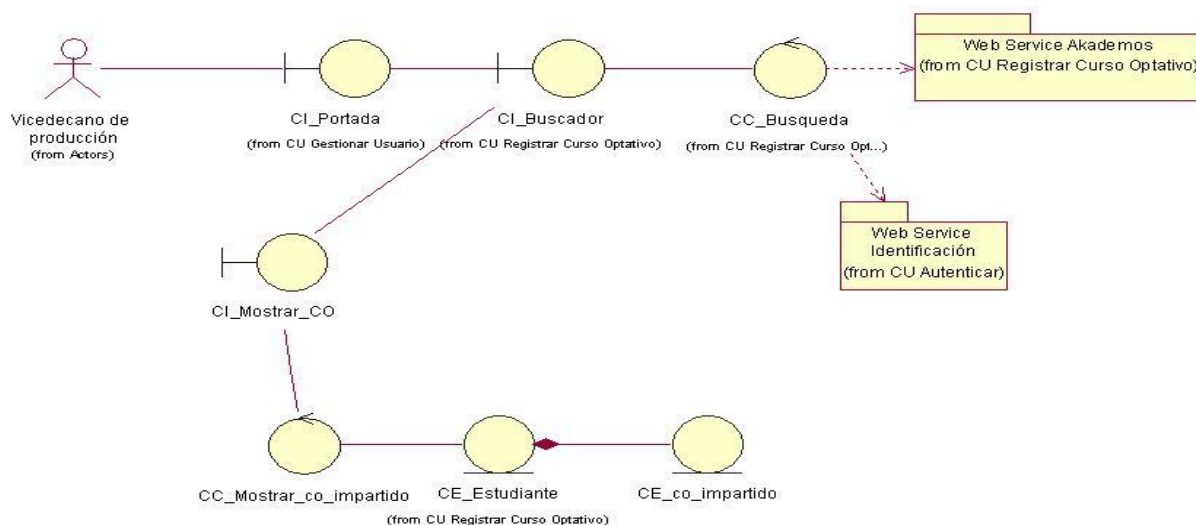
### Anexo 19: Clase de análisis del CU Registrar Incidencias



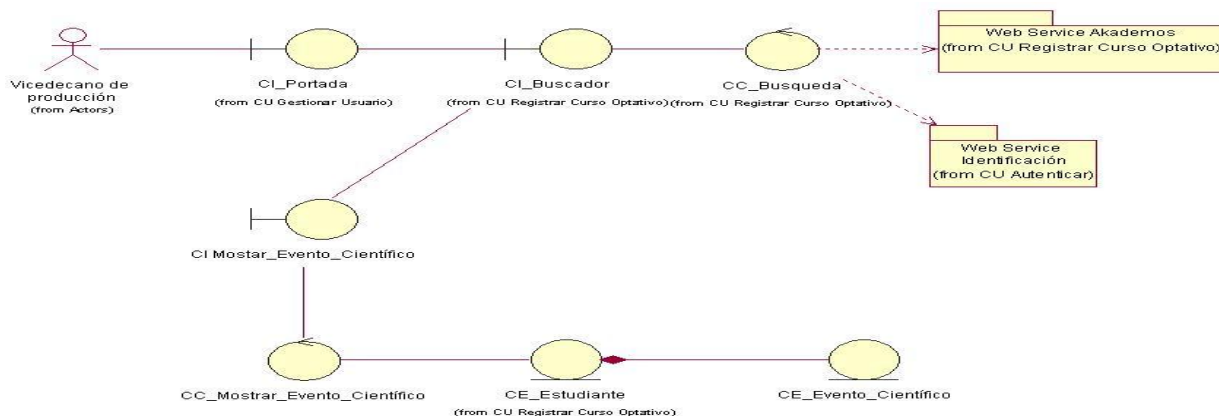
### Anexo 20: Clase de análisis del CU Registrar Líneas de Investigación



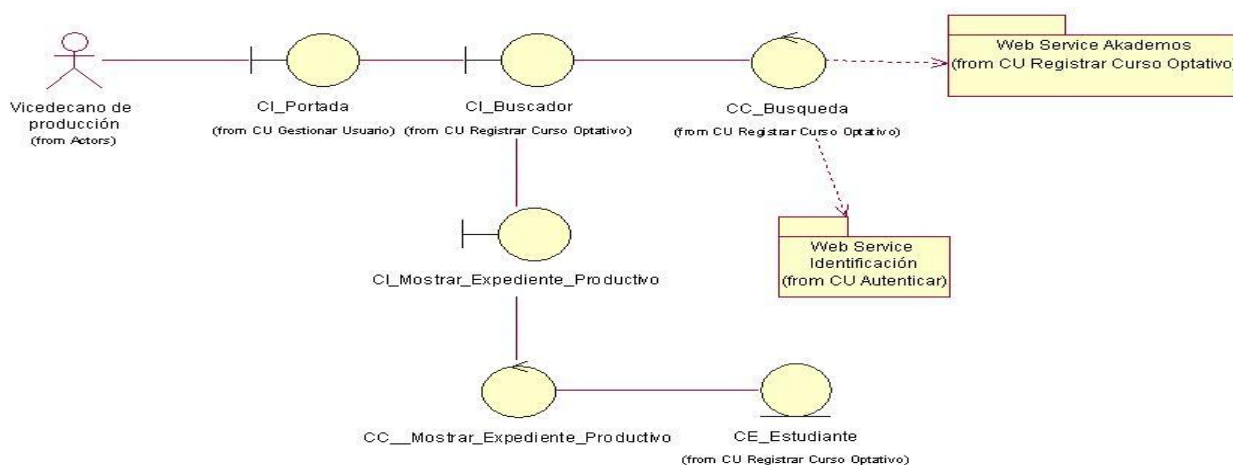
### Anexo 21: Clase de análisis del CU Mostrar Curso Optativo



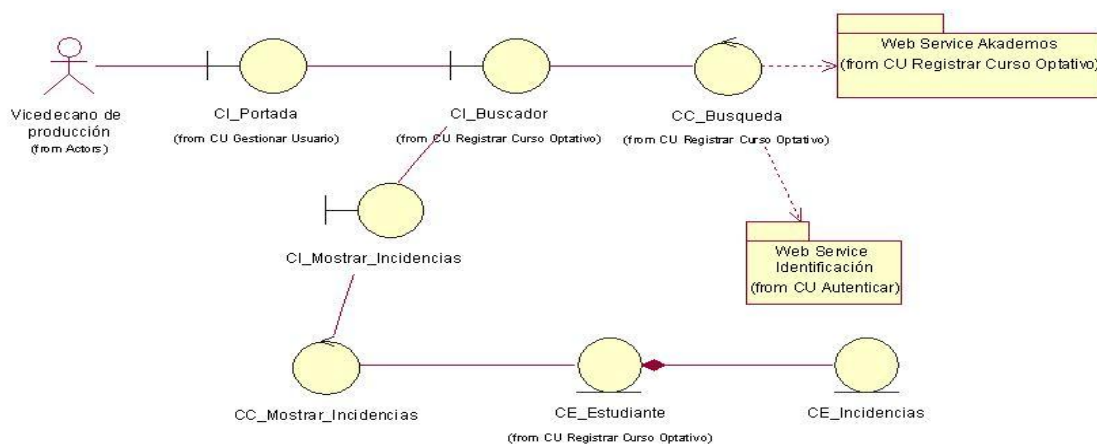
### Anexo 22: Clase de análisis del CU Mostrar Evento Científico



### Anexo 23: Clase de análisis del CU Mostrar Expediente Productivo

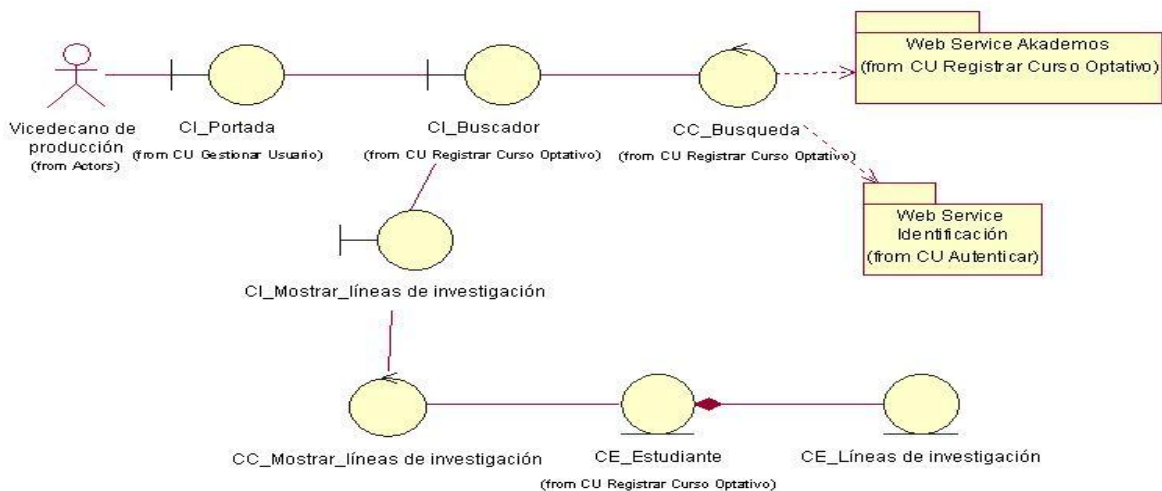


### Anexo 24: Clase de análisis del CU Mostrar Incidencias

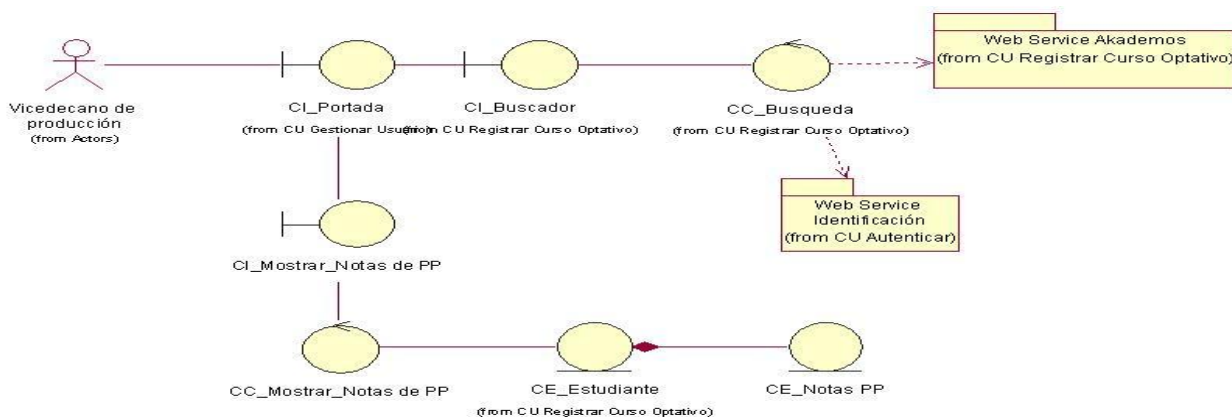




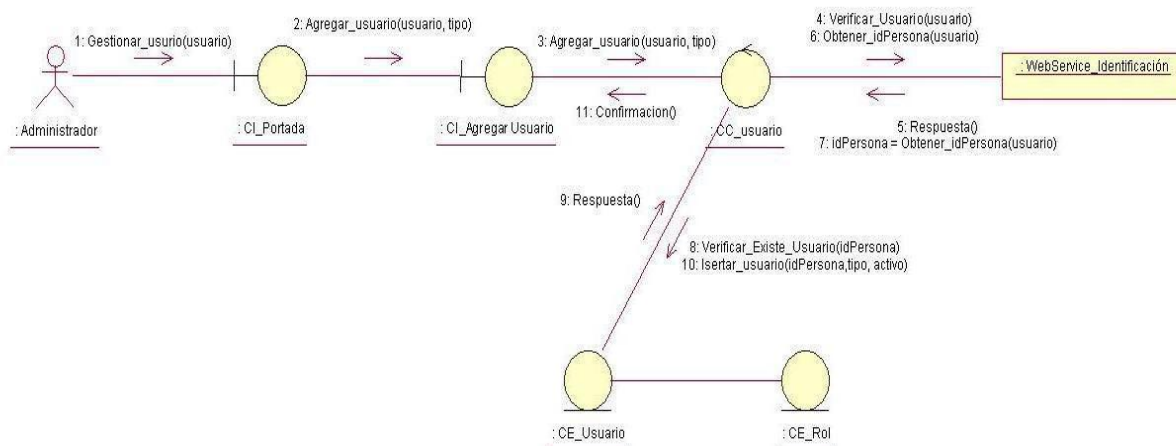
### Anexo 25: Clase de análisis del CU Mostrar Líneas de Investigación

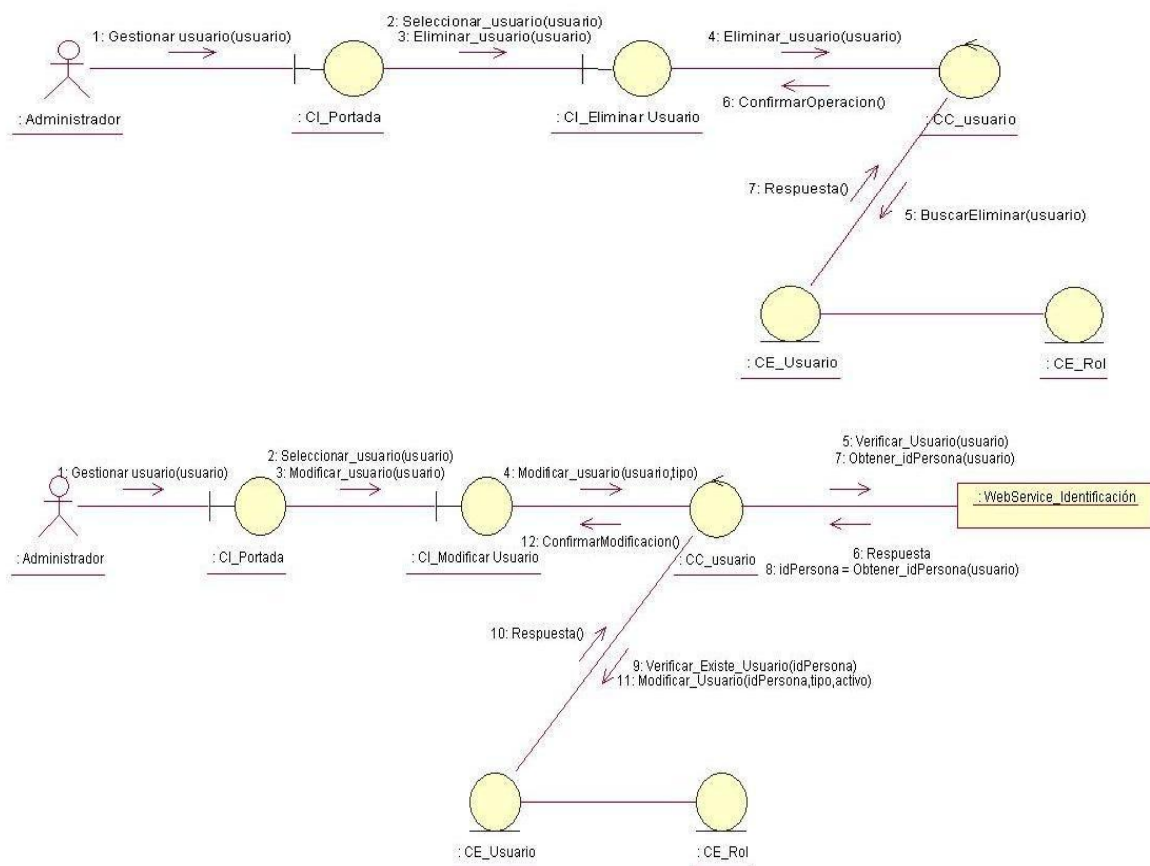


### Anexo 26: Clase de análisis del CU Mostrar Nota de PP

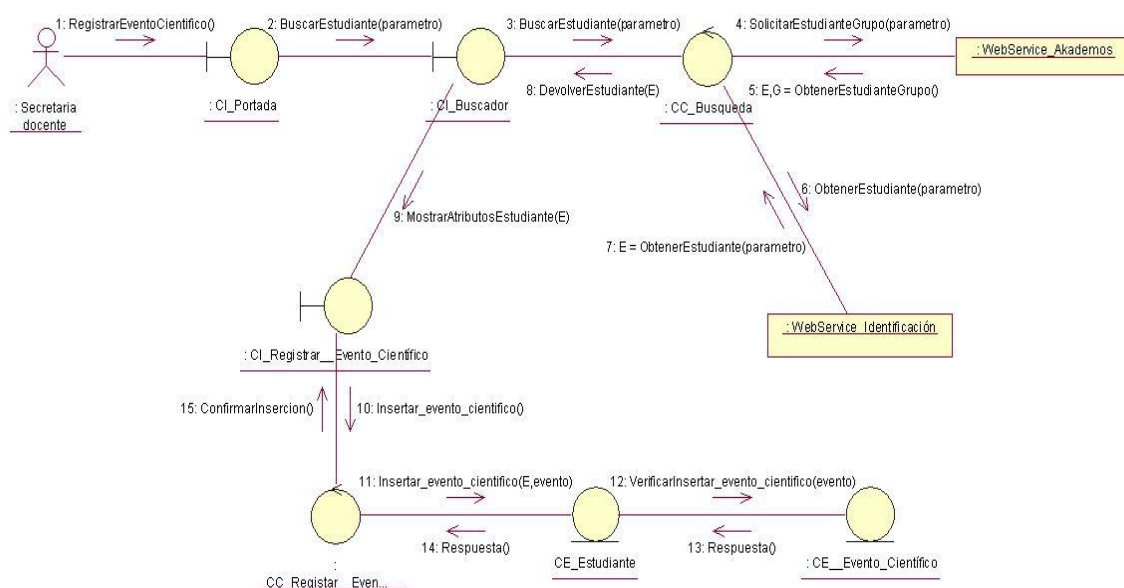


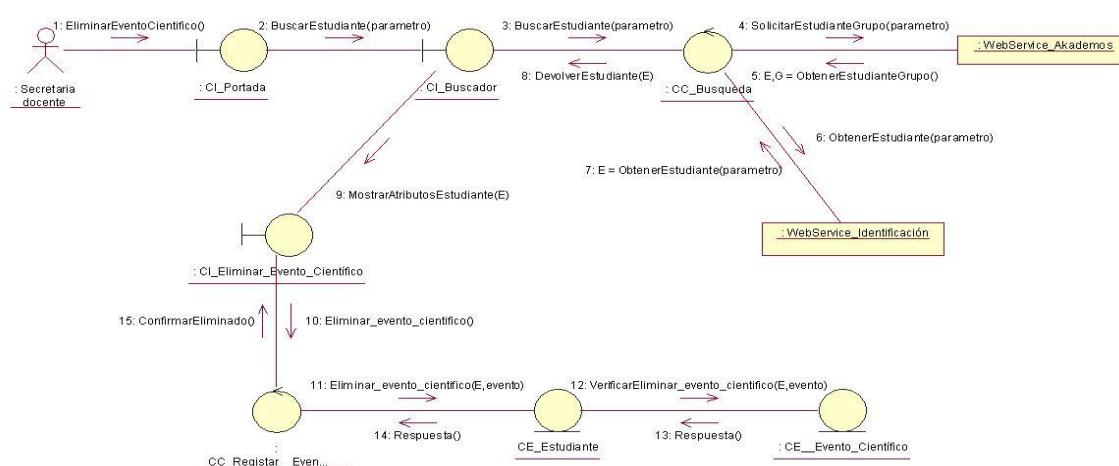
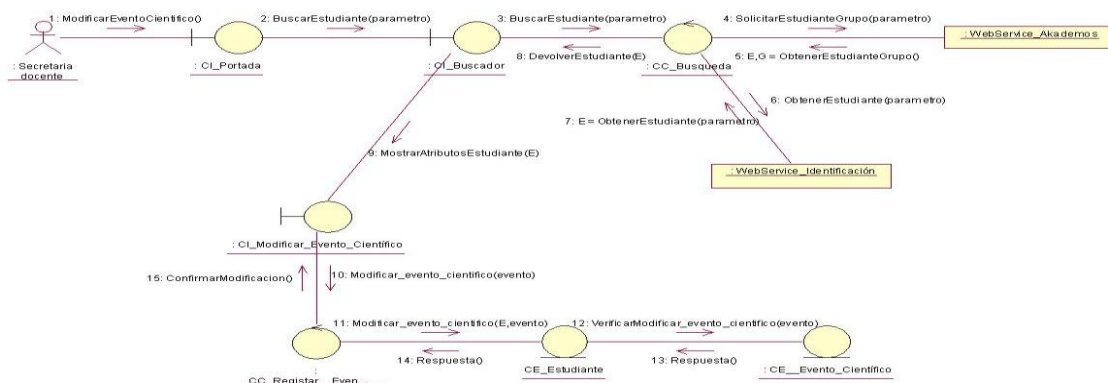
### Anexo 27: Clase de colaboración del análisis del CU Gestionar Usuario



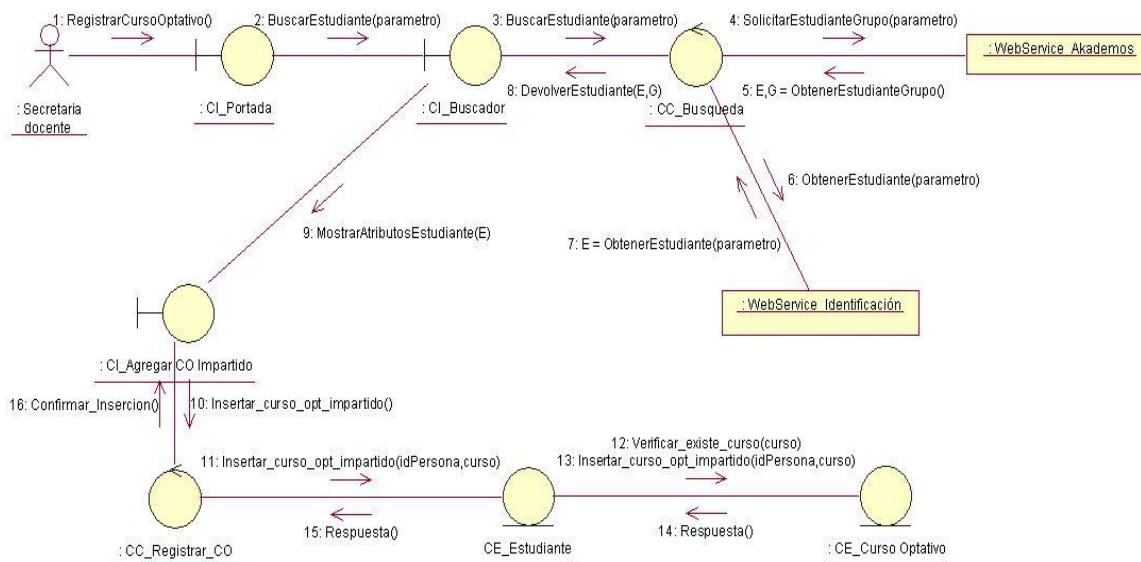


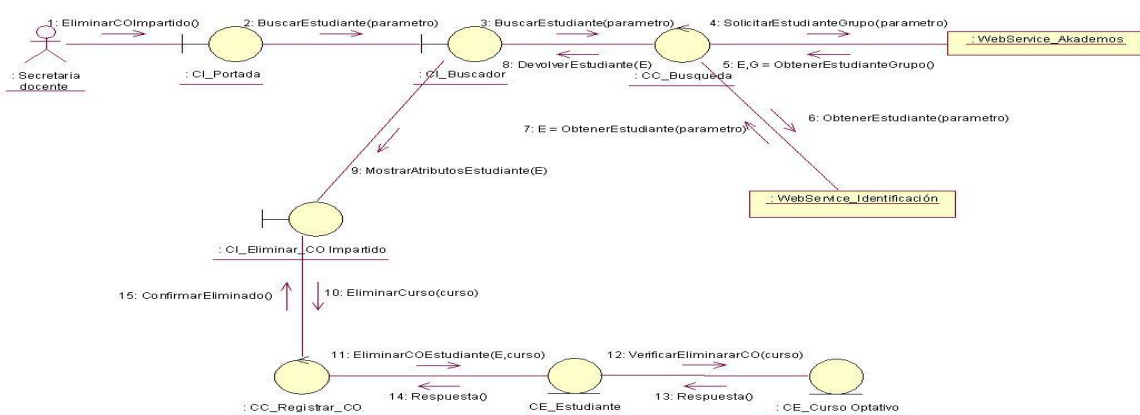
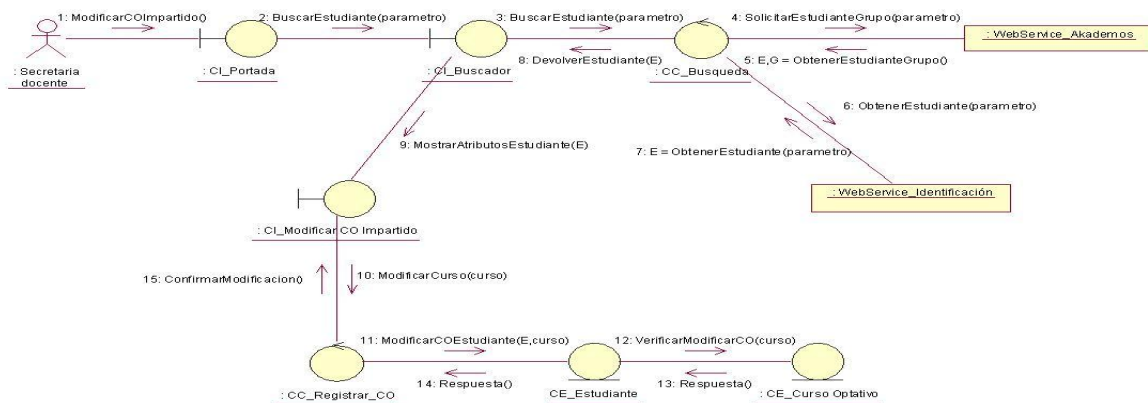
**Anexo 28: Clase de colaboración del análisis del CU Gestionar Evento Científico.**



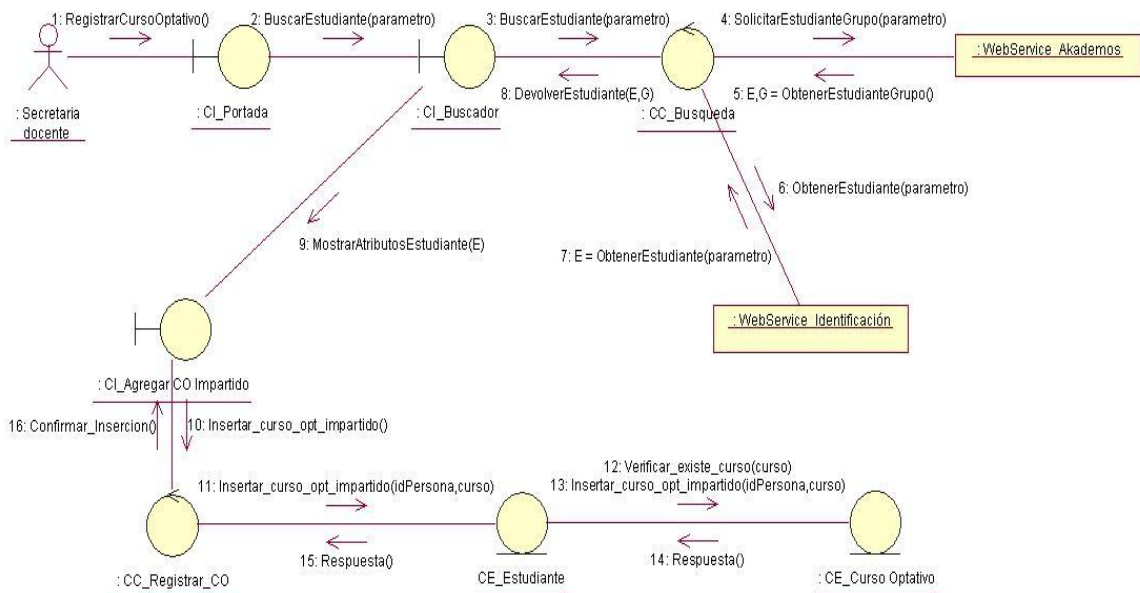


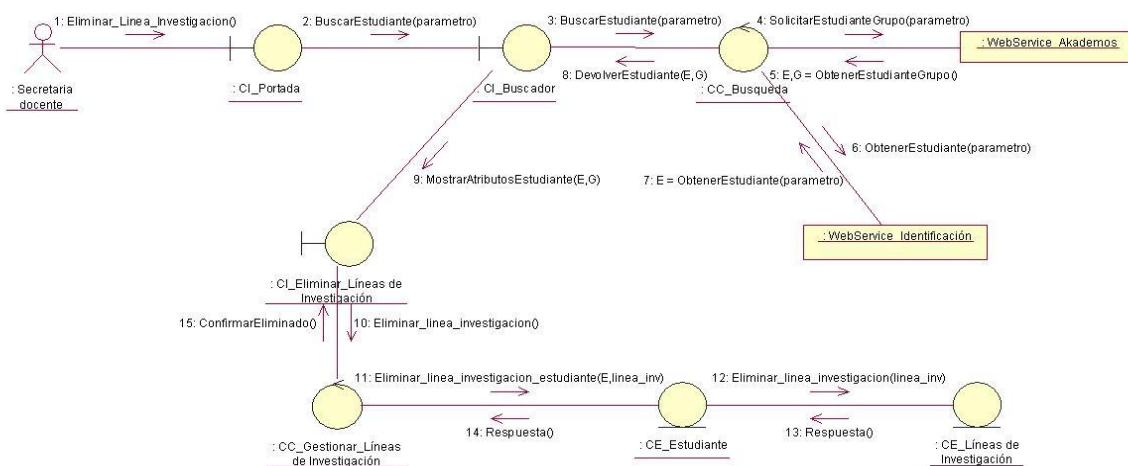
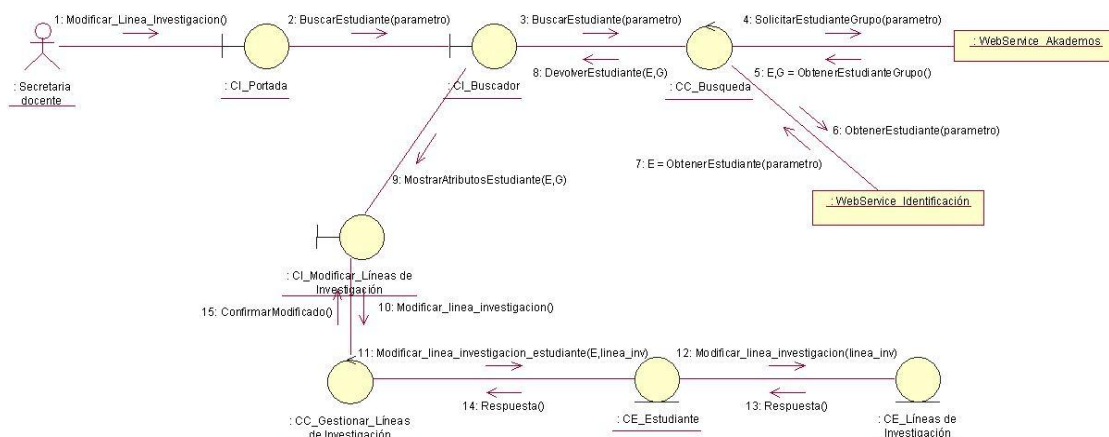
**Anexo 29: Clase de colaboración del análisis del CU Gestionar Curso Optativo.**



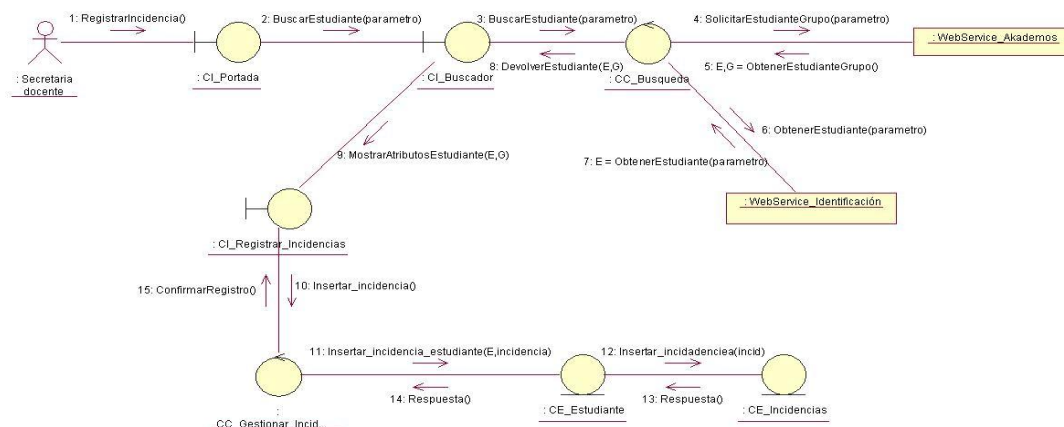


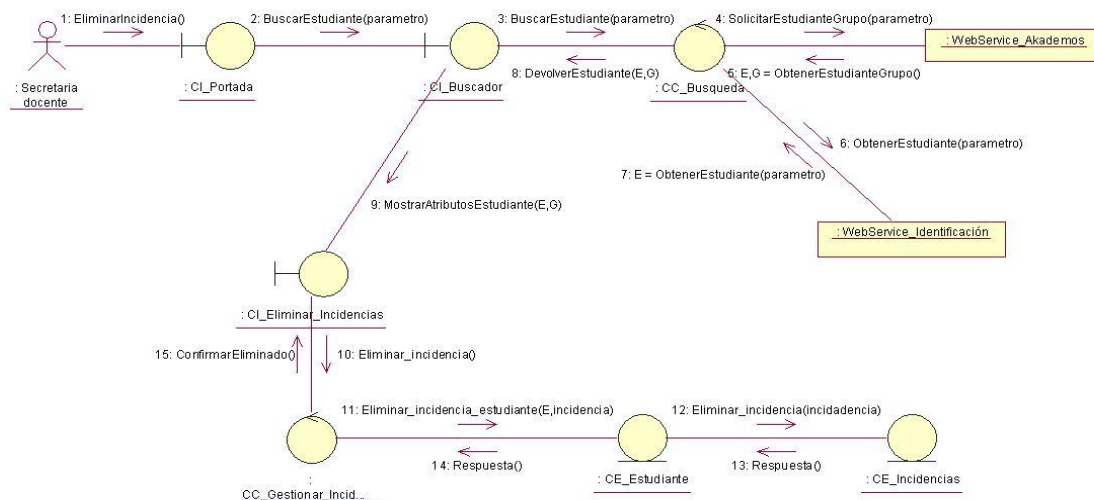
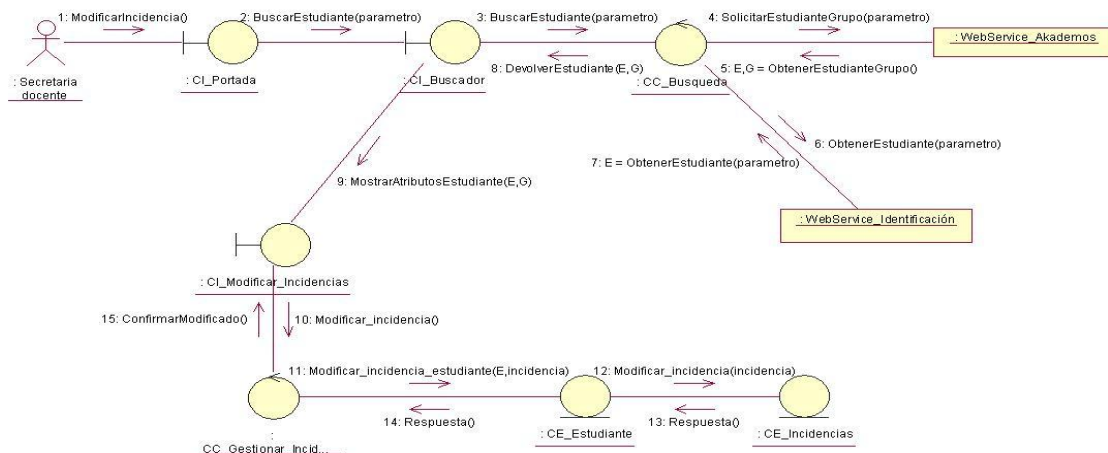
**Anexo 30: Clase de colaboración del análisis del CU Gestionar Líneas de Investigación.**



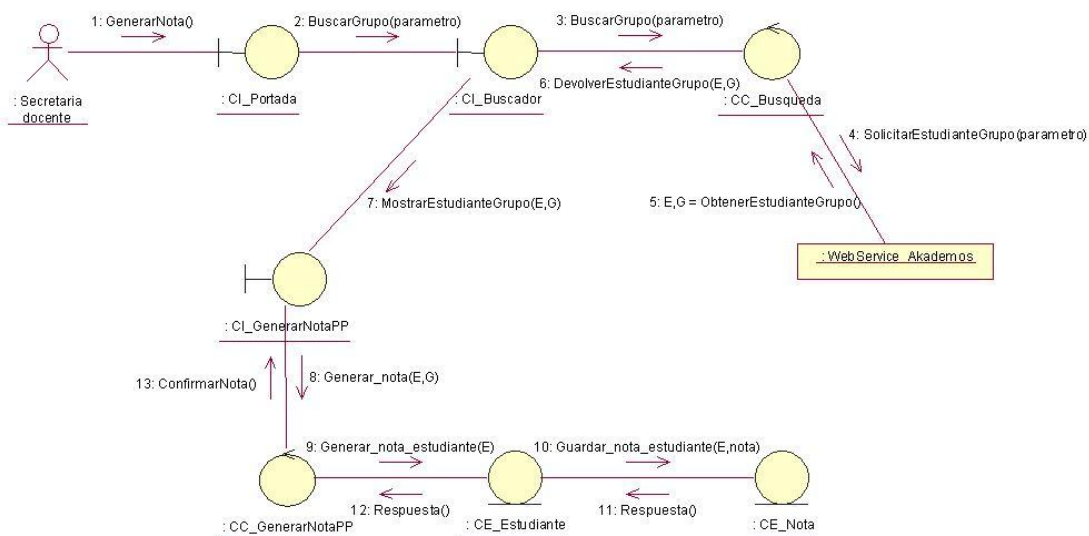


### Anexo 31: Clase de colaboración del análisis del CU Gestionar Incidencias.



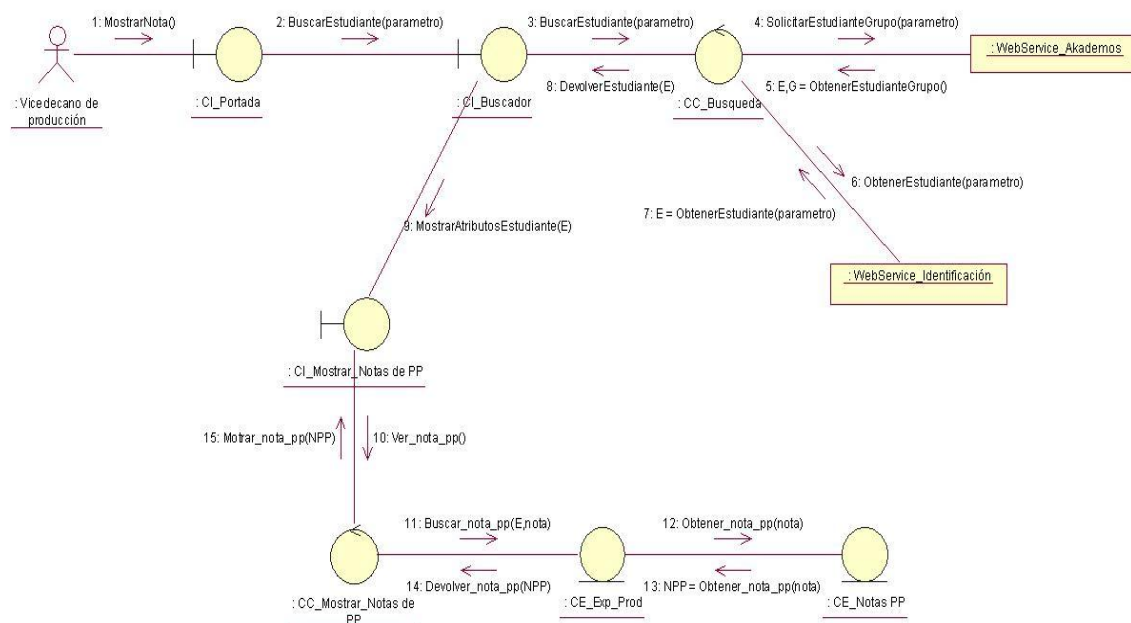


**Anexo 32: Clase de colaboración del análisis del CU Generar Nota PP.**

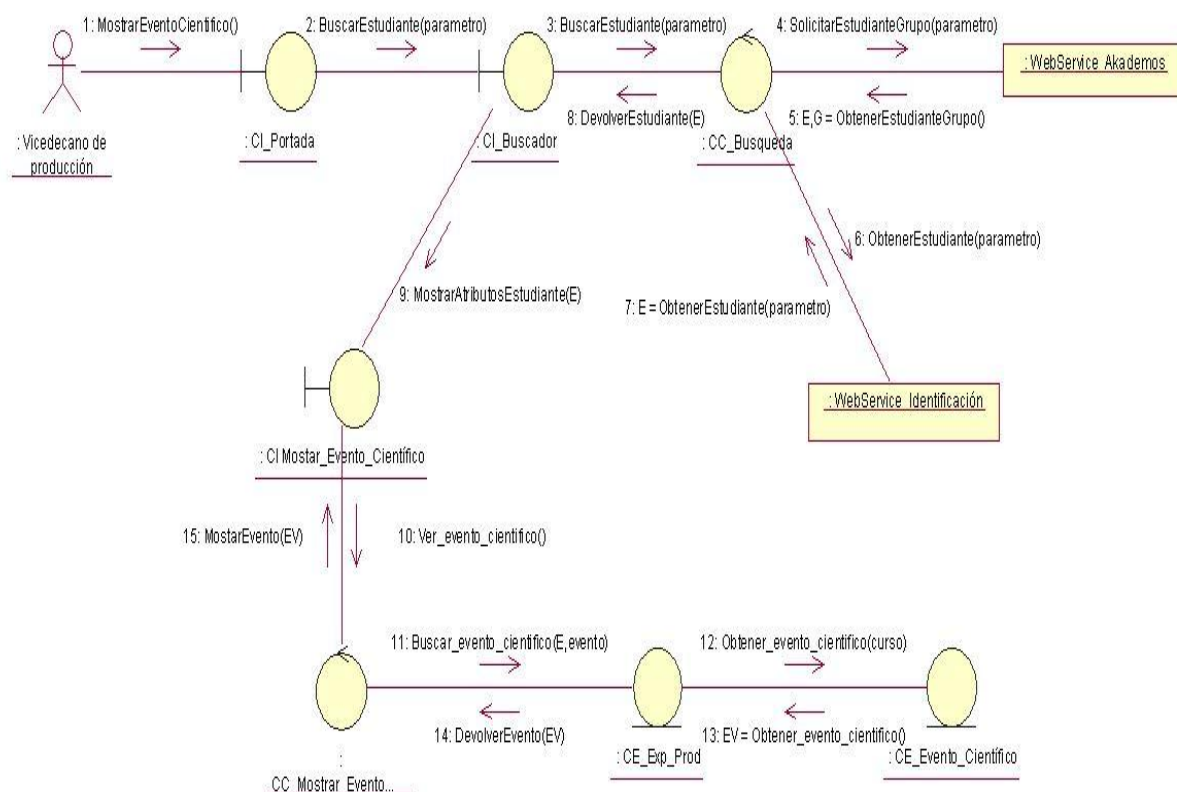




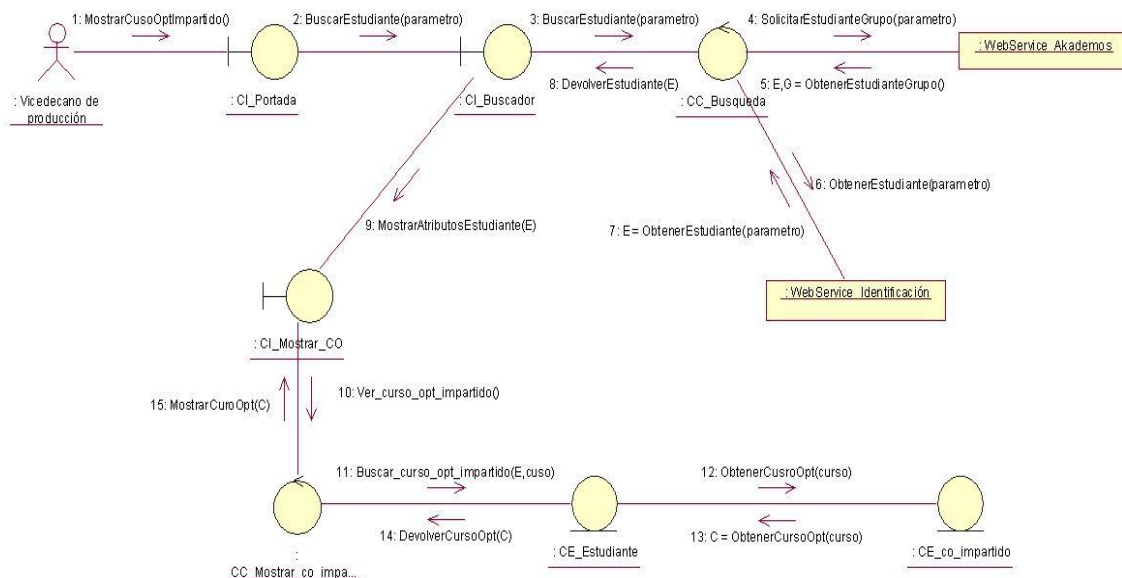
**Anexo 33: Clase de colaboración del análisis del CU Mostrar Nota de PP.**



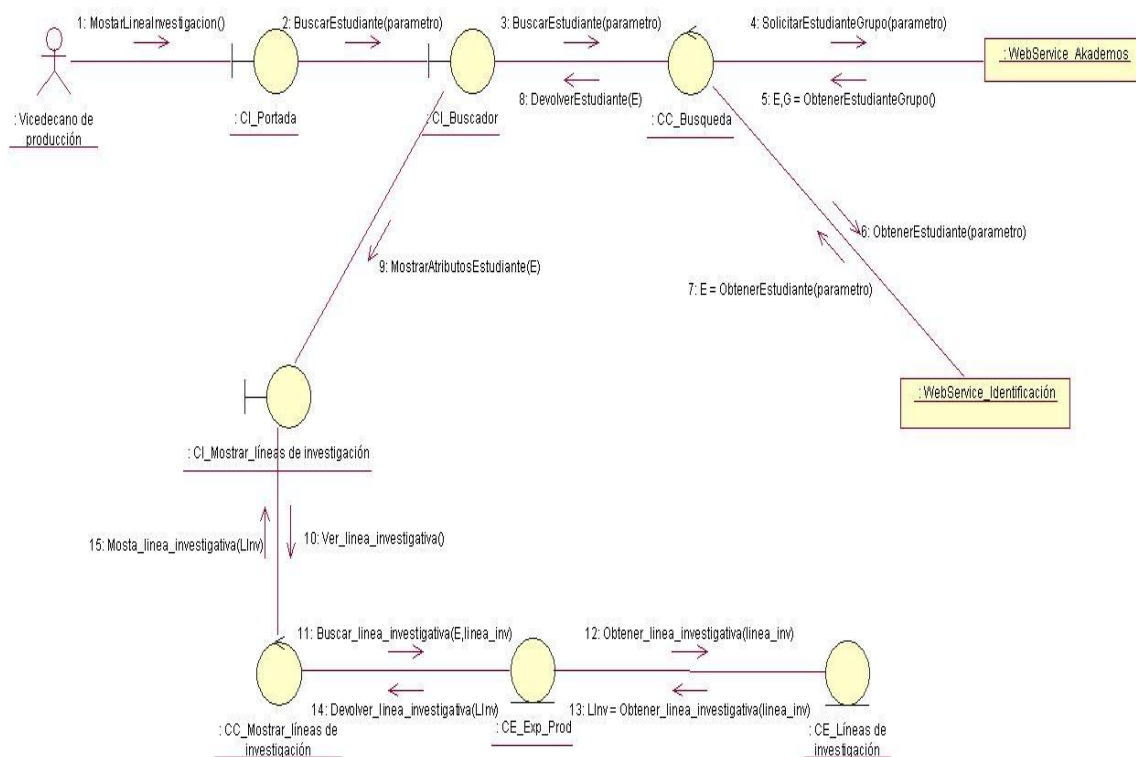
**Anexo 34: Clase de colaboración del análisis del CU Mostrar Eventos Científicos.**



### Anexo 35: Clase de colaboración del análisis del CU Mostrar Curso Optativo.

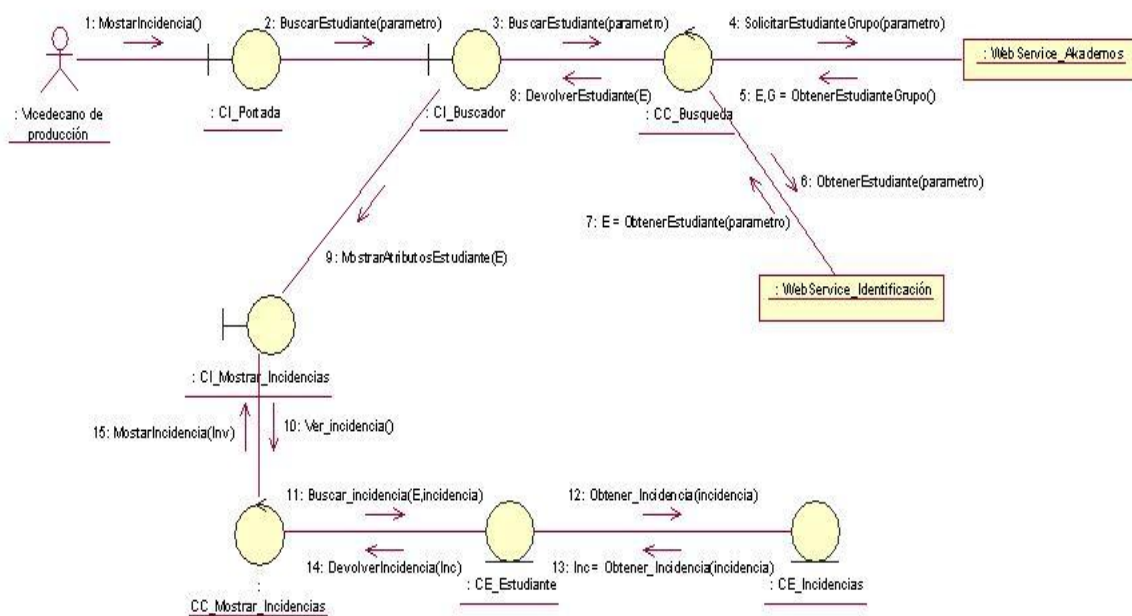


### Anexo 36: Clase de colaboración del análisis del CU Mostrar Líneas de Investigación.

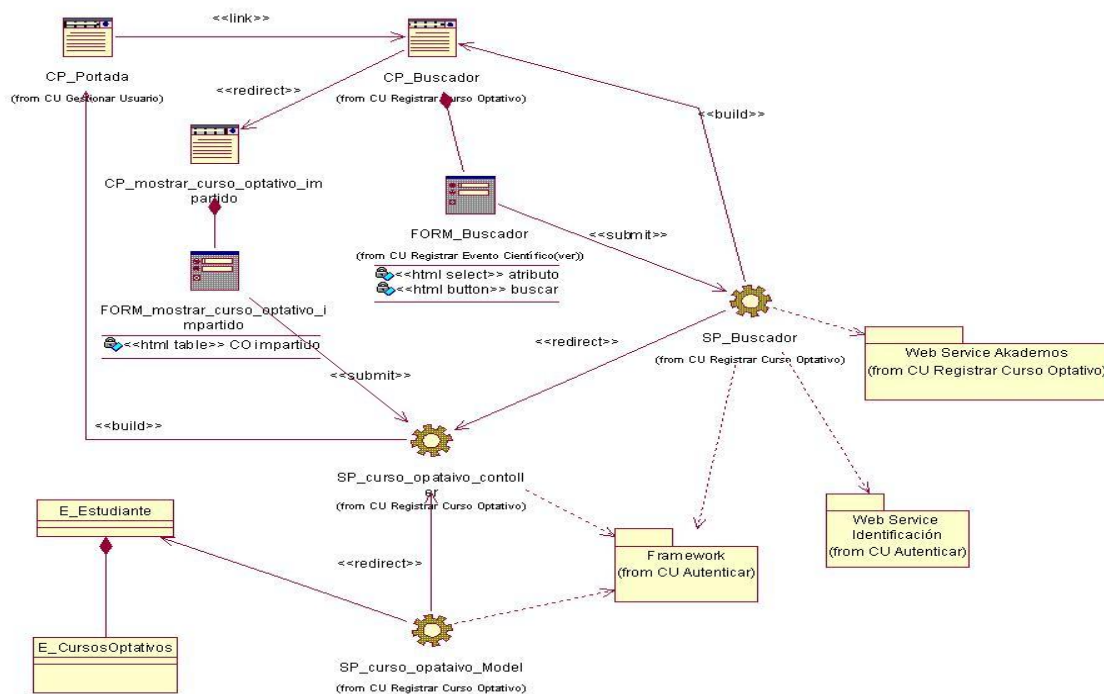




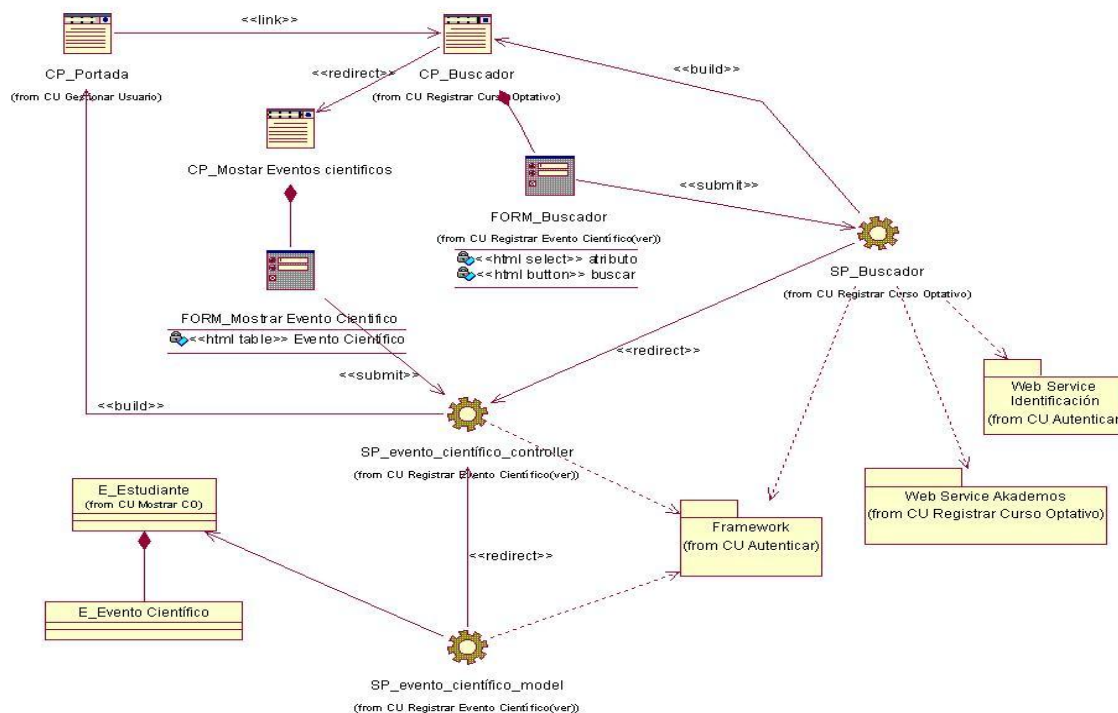
### Anexo 37: Clase de colaboración del análisis del CU Mostrar Incidencias.



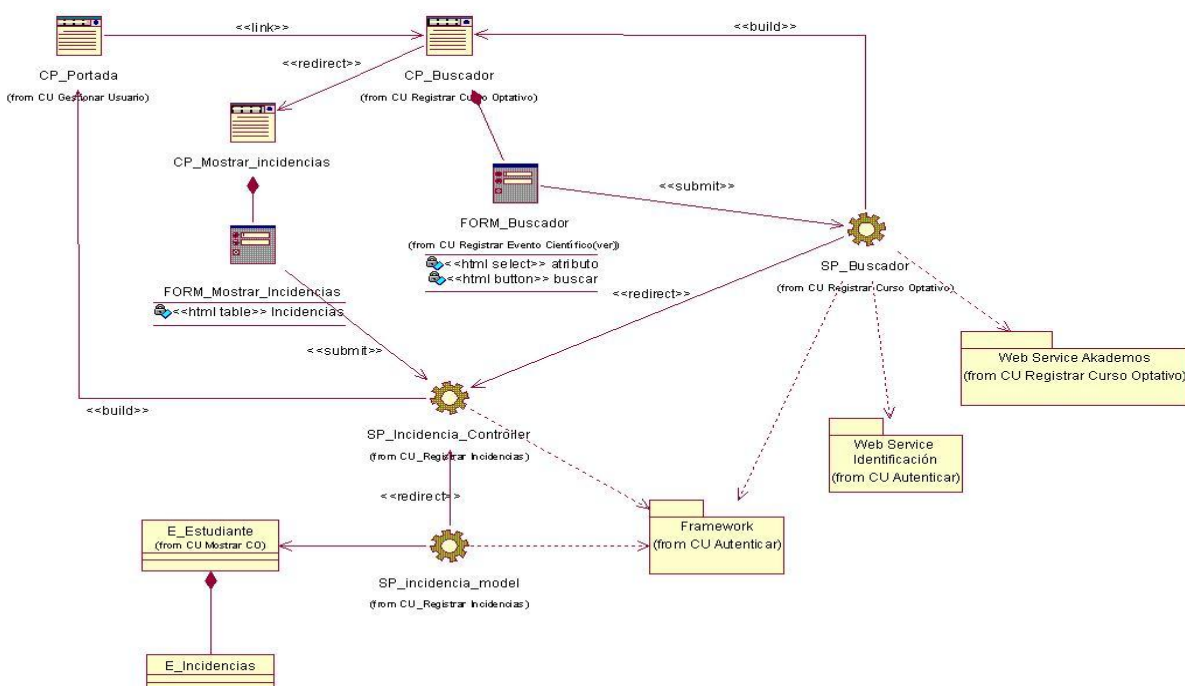
### Anexo 38: Clase de Diseño del CU Mostrar Curso Optativo.



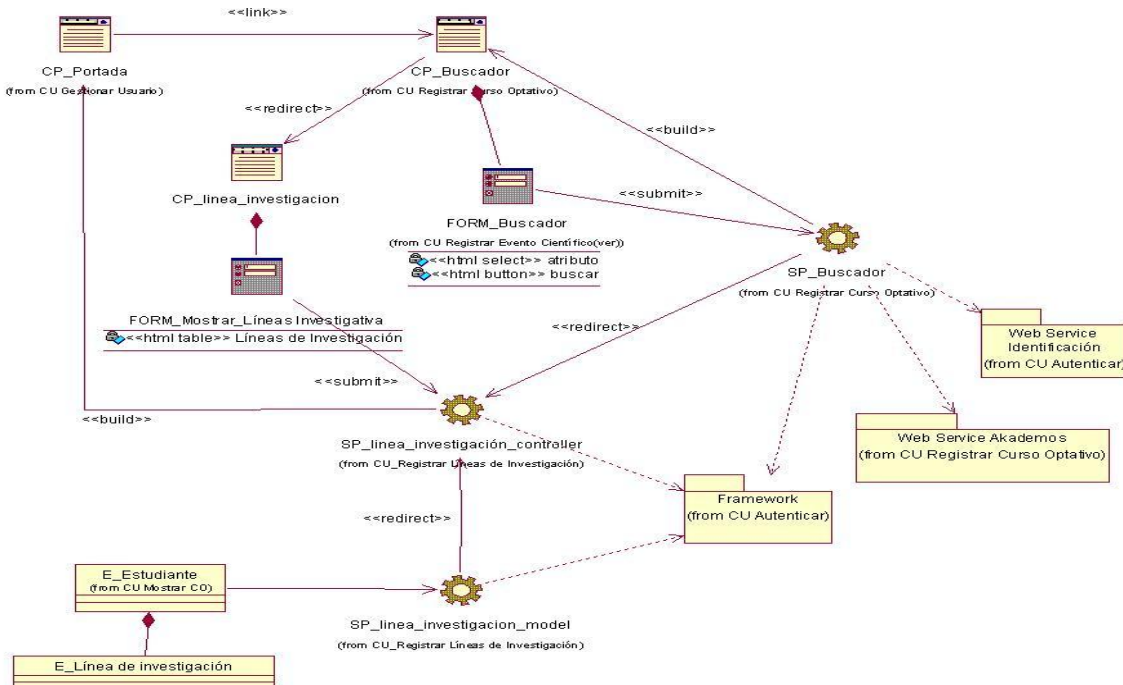
### Anexo 39: Clase de Diseño del CU Mostrar Evento Científico.



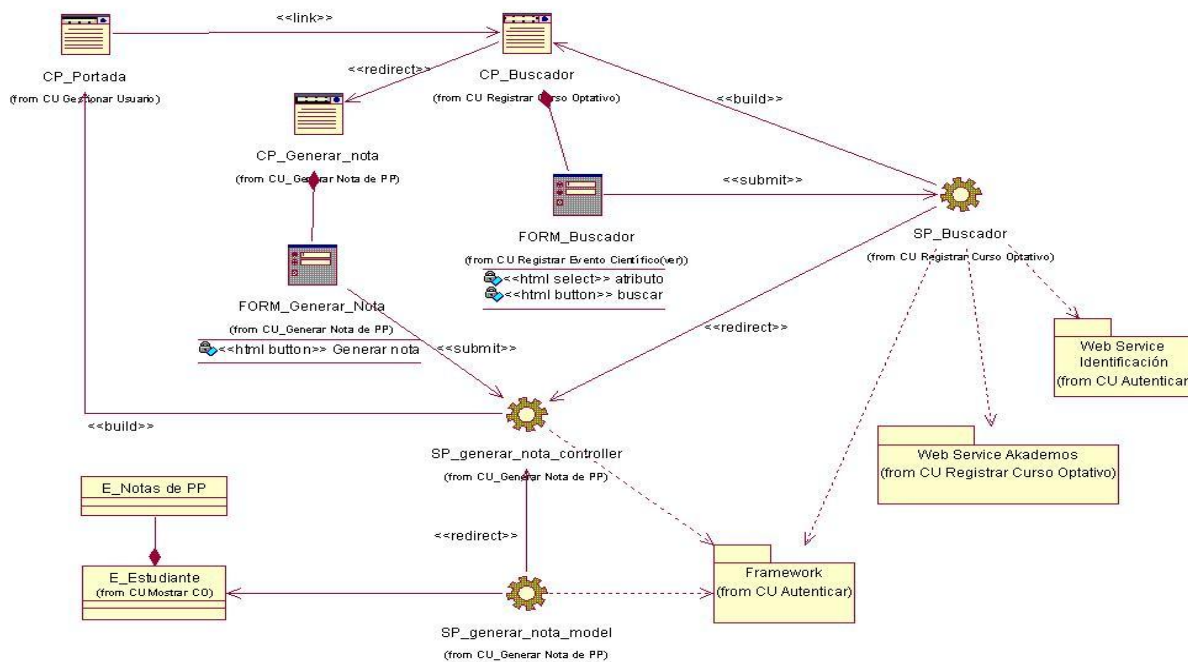
### Anexo 40: Clase de Diseño del CU Mostrar Incidencias.



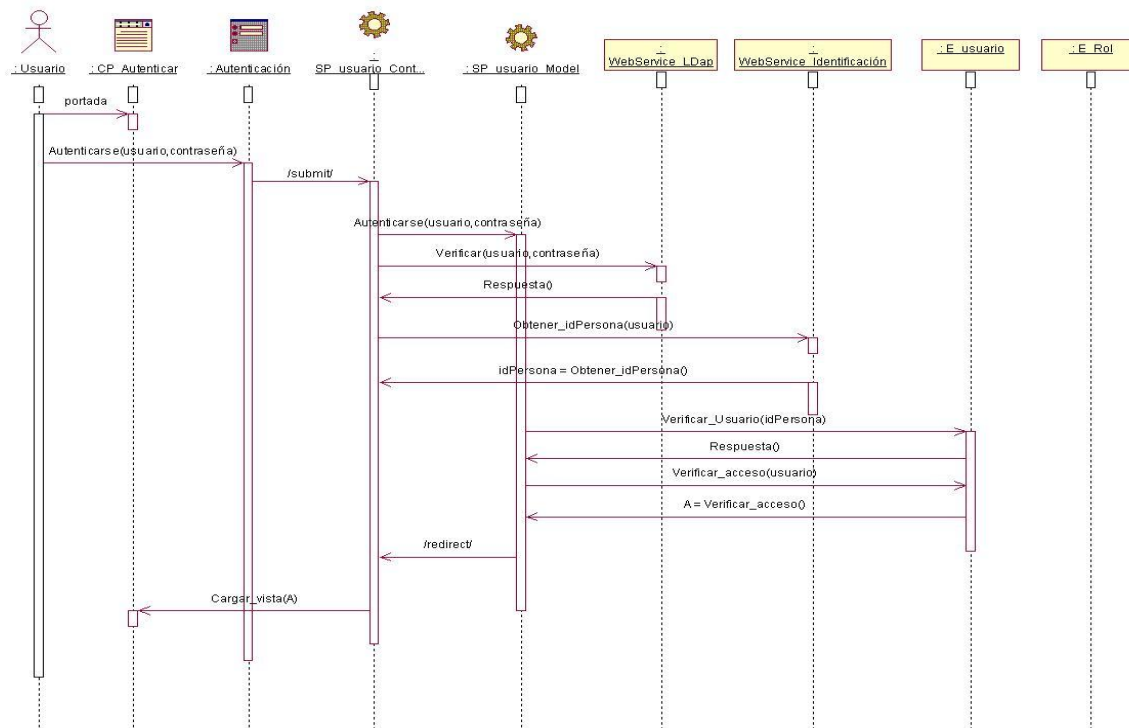
### Anexo 41: Clase de Diseño del CU Mostrar Líneas de Investigación.



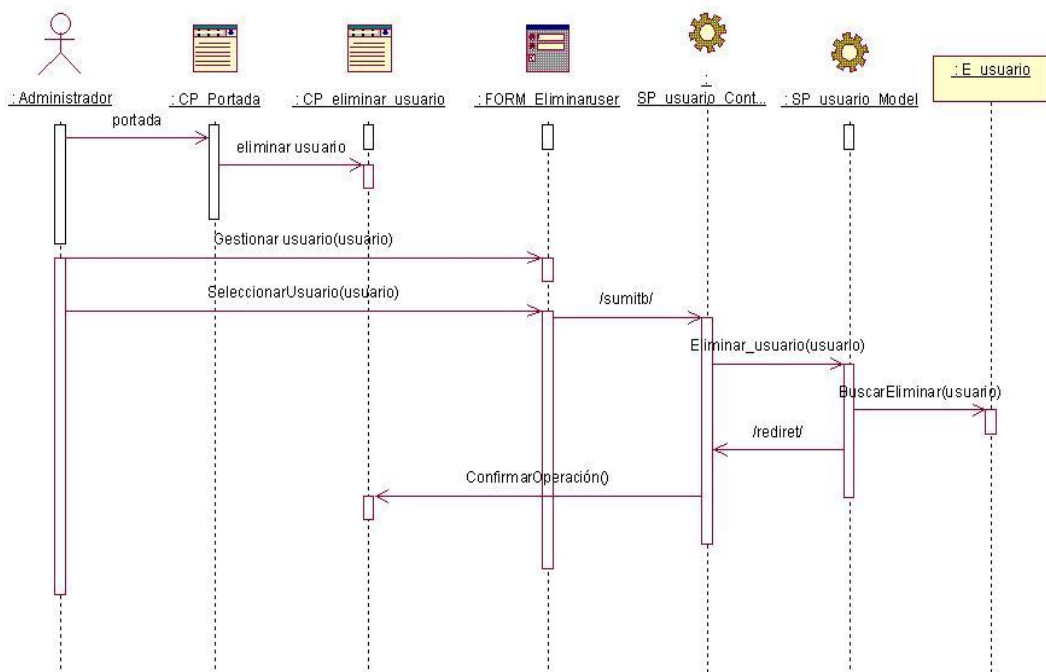
### Anexo 42: Clase de Diseño del CU Mostrar Nota de PP.

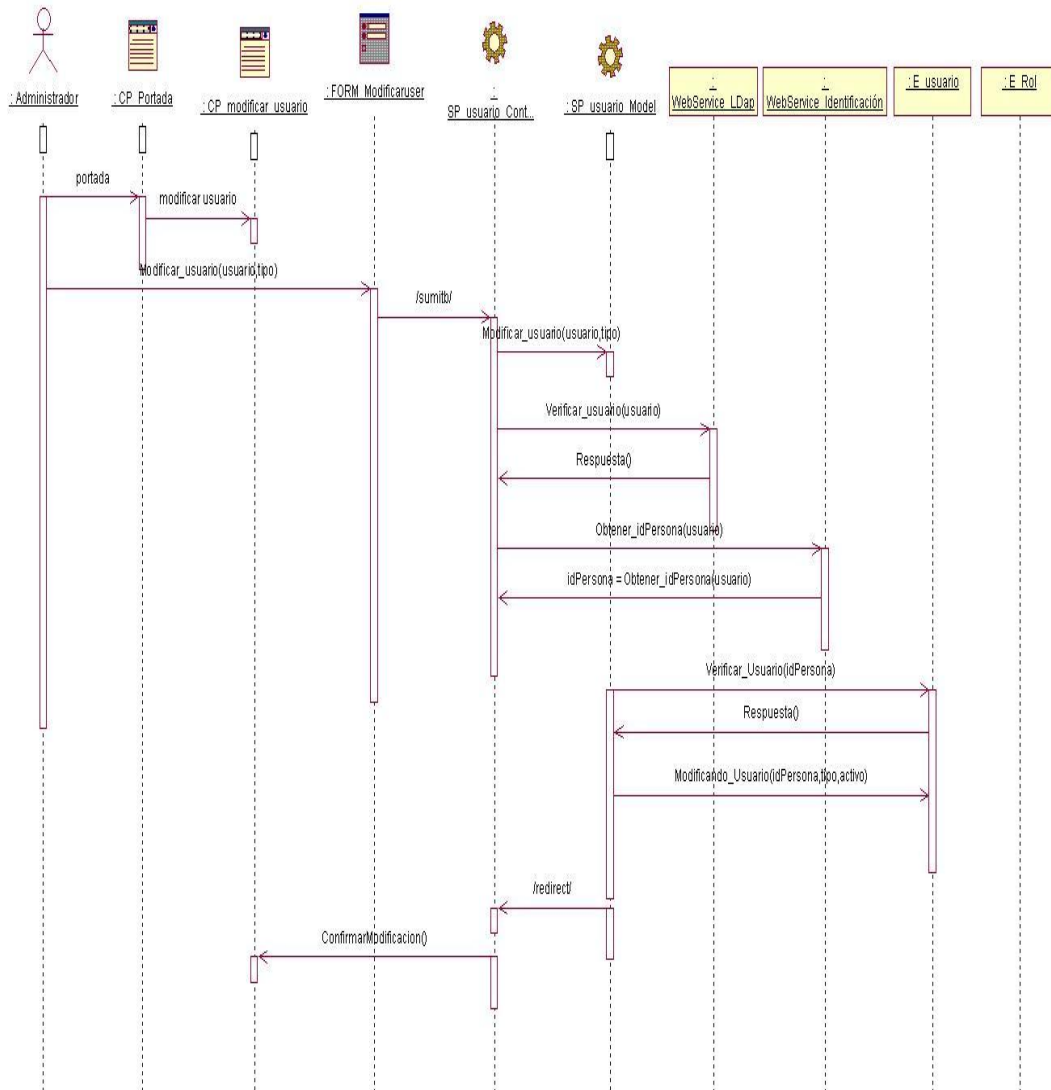


### Anexo 43: Clase de Secuencia del Diseño del CU Autenticar.

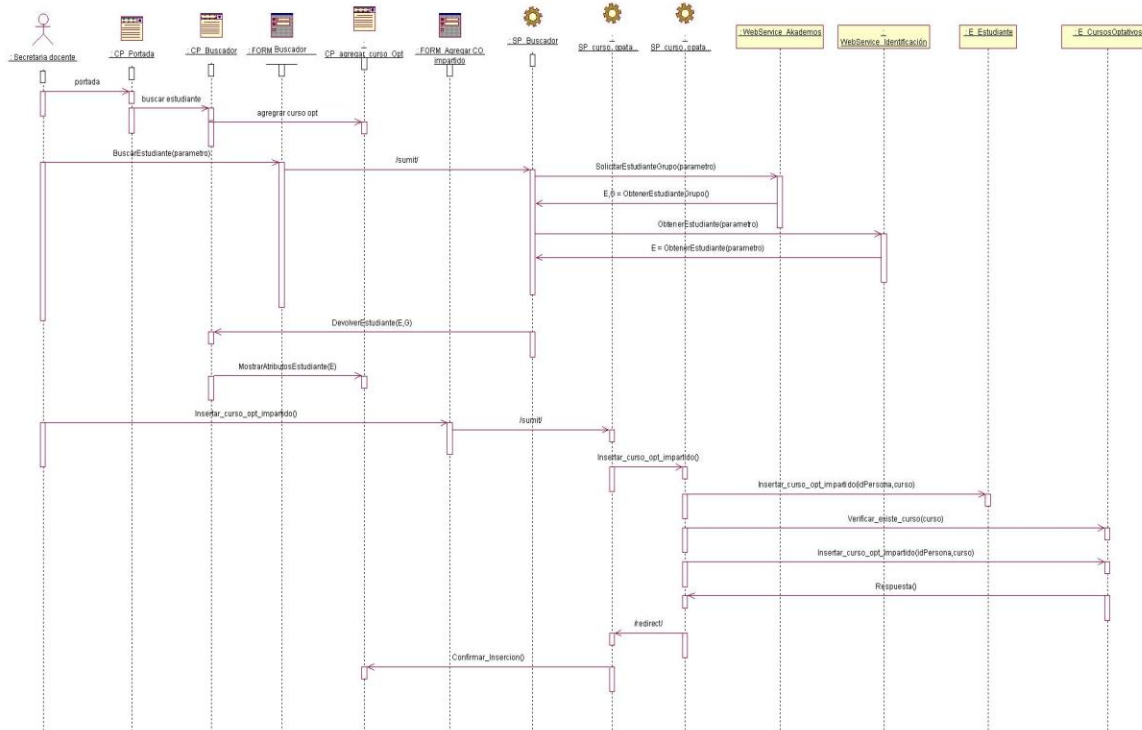


### Anexo 44: Clase de Diseño del CU Gestionar Usuario.

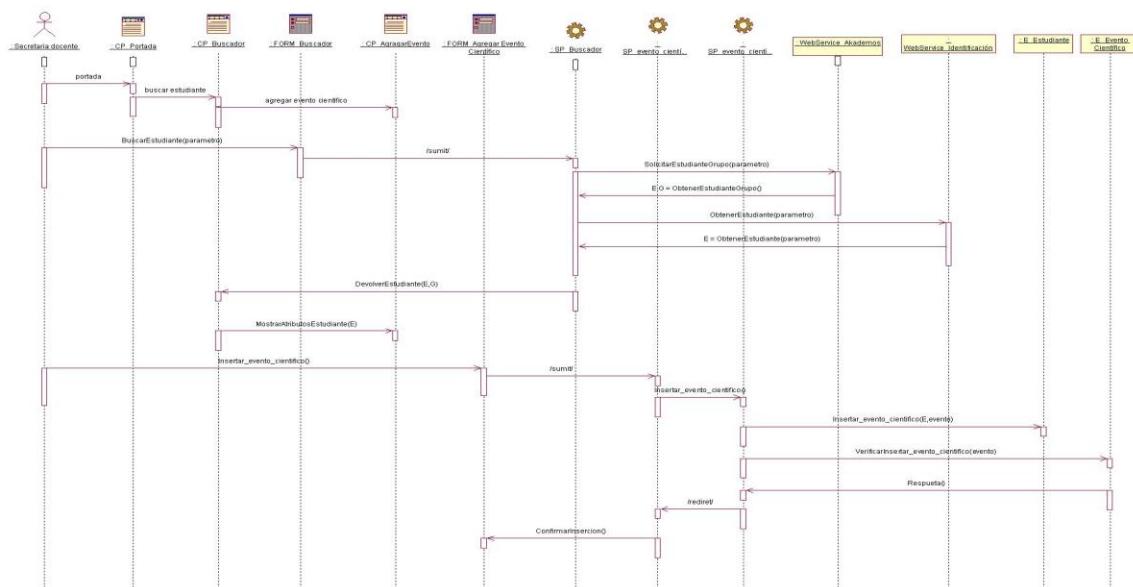




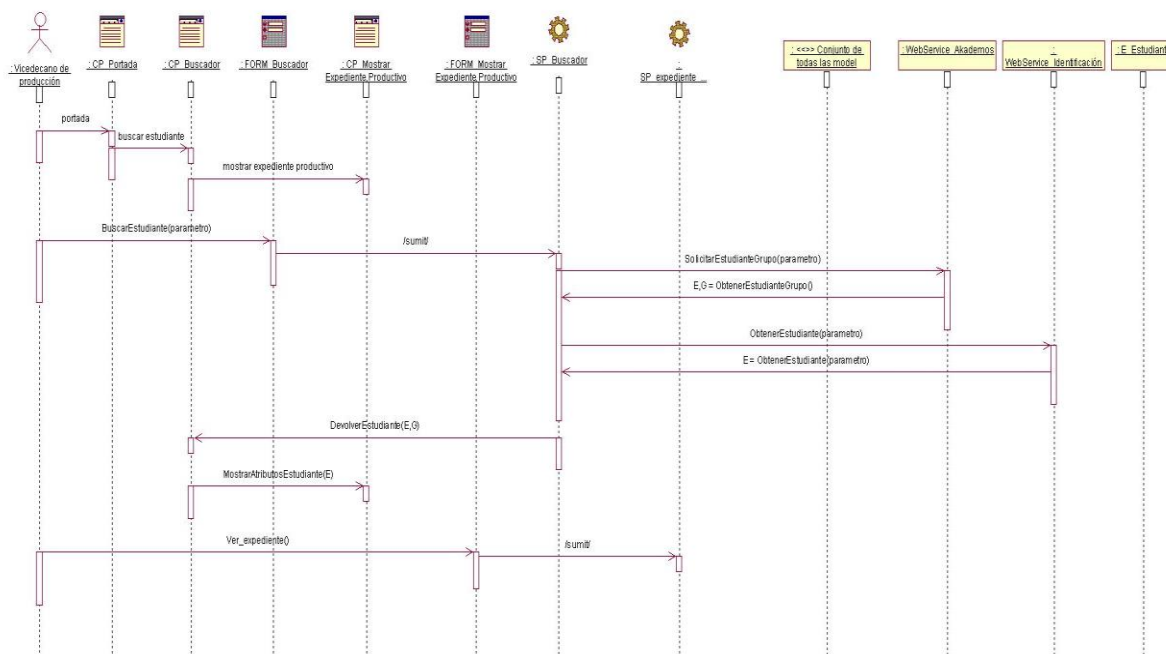
### Anexo 45: Clase de Diseño del CU Gestionar Curso Optativo.



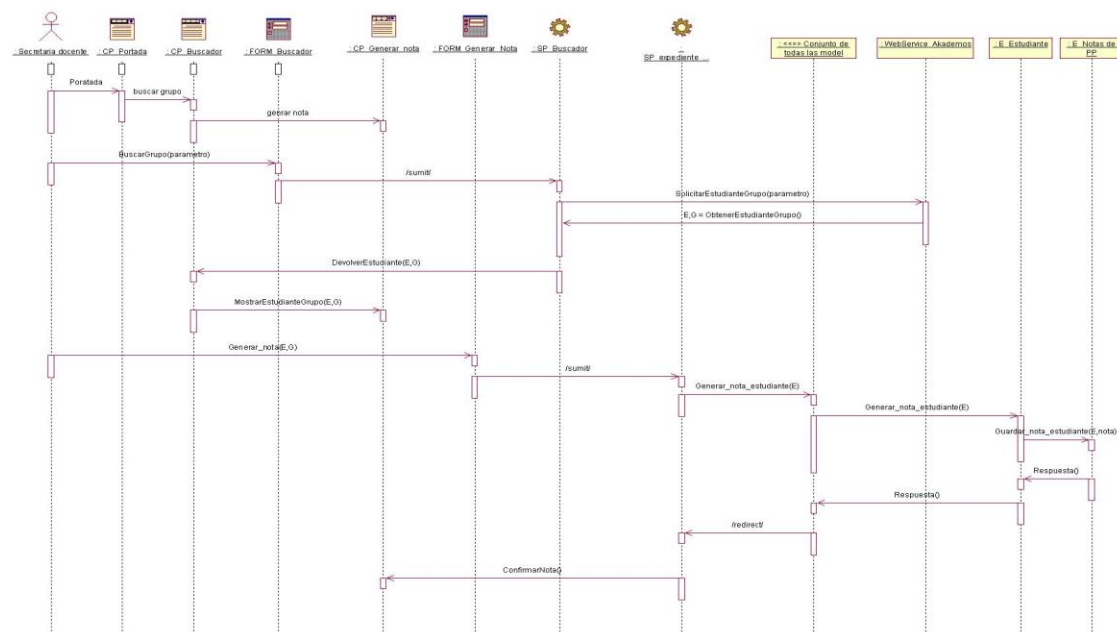
### Anexo 46: Clase de Diseño del CU Gestionar Evento Científico.



### Anexo 47: Clase de Diseño del CU Mostrar Expediente Productivo.



### Anexo 48: Clase de Diseño del CU Generar Nota de PP.





## GLOSARIO DE TÉRMINOS

1. **CASE (Computer-Aided Software Engineering):** Ingeniería de Software Asistida por Ordenador.
2. **RUP (Racional Unified Process):** El Proceso Unificado de Desarrollo Software o simplemente Proceso Unificado es un marco de desarrollo software iterativo e incremental. El refinamiento más conocido y documentado del Proceso Unificado es el Proceso Unificado de Rational o simplemente RUP.
3. **UML (Unified Modeling Language):** Es un lenguaje gráfico para visualizar, especificar y documentar cada una de las partes que comprende el desarrollo de software. Entrega una forma de modelar cosas conceptuales como lo son procesos del negocio y funciones de sistema, además de cosas concretas como lo son escribir clases en un lenguaje determinado, esquemas de base de datos y componentes de software reusables.
4. **CSS(Cascading Style Sheets):** Hojas de estilo en cascada, son un lenguaje formal usado para definir la presentación de un documento estructurado.
5. **HTML (HyperText Markup Language):** Lenguaje de Marcado de Hipertexto, predominante para la construcción de páginas web.
6. **XHTML (eXtensible Hypertext Markup Language):** Lenguaje extensible de marcado de hipertexto).
7. **XML (Extensible Markup Language):** Lenguaje de marcas extensibles, es un metalenguaje extensible de etiquetas. No es realmente un lenguaje en particular, sino una manera de definir lenguajes para diferentes necesidades.
8. **LAN (Local Area Network):** Red de área local.
9. **WAN (Wide Area Network):** Red de área amplia.
10. **MVC(Modelo-Vista-Controlador):** Patrón de arquitectura, este patrón separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos.