



Universidad de las Ciencias Informáticas



Instituto Superior Politécnico "José Antonio Echeverría"
Facultad de Ingeniería Industrial
Centros de Estudios de Ingeniería de Sistemas

Propuesta
de
WEB DOCENTE

Trabajo de diploma para optar por el
Título de Ingeniería Informática

Autor: Yamilka Fernández Tur

Tutor: Lic. Edistio Verdecia Martínez

Ciudad de La Habana,
Junio del 2006

“Nunca consideres el estudio como una obligación sino como una oportunidad para penetrar en el bello y maravilloso mundo del saber”

Albert Einstein

DECLARACIÓN DE AUTORÍA

Declaro que yo, Yamilka Fernández Tur, soy el único autor de este trabajo y autorizo a la Universidad de las Ciencias Informáticas (UCI) y al Centro de Estudios de Ingeniería y Sistemas (CEIS) para que hagan el uso que estimen pertinente con este trabajo.

Para que así conste firmo la presente a los _____ días del mes de _____ del _____.

Firma del autor

Yamilka Fernández Tur

Firma del tutor

Edistio Yoel Verdecia Martinez

OPINIÓN DEL USUARIO DEL TRABAJO DE DIPLOMA

El Trabajo de Diploma titulado “WEB DOCENTE”, fue desarrollado en la Universidad de las Ciencias Informáticas. Esta entidad considera que, en correspondencia con los objetivos trazados, el trabajo realizado le satisface:

Totalmente _____

Parcialmente en un _____ %

Los resultados de este Trabajo de Diploma le reportan a esta entidad los beneficios siguientes:

Y para que así conste, se firma la presente a los ____ días del mes de _____ del año _____

Representante de la entidad

Cargo

Firma

Cuño

AGRADECIMIENTOS

La cualidad más hermosa que debe revestir un ser humano es reconocer a quienes lo han llevado a conseguir sus objetivos, por eso quiero expresar mis más profundos agradecimientos a todos los que han hecho posible de alguna forma que haya llegado hasta aquí:

- *A mis padres por su cariño, confianza y apoyo, por conducirme por el buen camino, gracias por todo.*
- *A Edístio, mi tutor, que tanto me ha ayudado, que aunque un poco lejos, siempre estuvo ahí todo el tiempo, supervisando mi trabajo.*
- *A mi hermana, por su cariño y preocupación, por estar siempre al tanto de todo, y de más...*
- *A Yosnel, a Mayli, a mis amigos que han estado siempre conmigo, que me han apoyado, y han confiado en mí, por poder contar con ellos.*
- *A mi familia por estar al tanto de todo, a mi tío en especial.*
- *A Alejandro por su apoyo, a Ailec por su buena ayuda, a Molina, y a los profesores que colaboraron con todas mis dudas.*
- *A mis compañeros de estudio que estuvieron presente siempre que los necesitaba, por su ayuda y colaboración durante estos 5 años, los de la UMCC y los de la CUJAE.*
- *A todos los profesores que he tenido durante mi vida de estudiante que han influido en que yo esté ahora aquí, sobre todo los de la Universidad.*

DEDICATORIA

At mis padres por ser mi máxima fuente de inspiración, por darme la vida y todo su amor, por formarme como la persona que hoy soy....

At ellos les regalo mi tesis.....su sueño...

At mi hermana...

At mis abuelos...a Mama, aunque no esté con nosotros en vida...

RESUMEN

Actualmente en todas las universidades del mundo, incluyendo las universidades cubanas, cobra una importancia vital el uso de las TIC's y dentro de este el que cada asignatura perteneciente al currículo de una determinada carrera posea un sitio Web en el cual se pueda colocar toda la información relacionada con ella. En la Universidad de las Ciencias Informáticas, donde el uso de las TIC es fundamental en el proceso de enseñanza – aprendizaje se exige esto con gran fuerza; está estipulado que cada asignatura posea un sitio Web donde debe estar toda la información referente a ella.

Estas Webs Docentes tienen como objetivo satisfacer al usuario en cuanto a obtención y manipulación de información que apoye el proceso de enseñanza-aprendizaje; los sitios existentes tienen como regularidades fundamentales no brindar el mismo tipo de información ni estar organizada de la misma manera, esto hace que en muchos casos los usuarios no puedan elaborar estrategias propias de manejo de información.

Esta situación nos permite afirmar que no existe en estos momentos una aplicación Web que logre tener toda la información referente a una asignatura concentrada y mostrada de forma coherente, cuyo uso pueda ser tan flexible como los procesos de la Universidad lo requieran, que permita a los usuarios manejar la información de manera rápida y eficiente, por lo que se propone como objetivo principal realizar el análisis de las necesidades vinculadas a la publicación de los materiales docentes, llegando al diseño de un Modelo de **"Web Docente"** que sirva para apoyar el proceso de enseñanza-aprendizaje en la universidad, y que se integre con el resto de los servicios de la Intranet Universitaria, dándole mayor protagonismo al uso de las TIC.

Para realizar el análisis y diseño de este sitio se siguió la metodología RUP (Proceso Unificado del Software), y para su implementación se propone utilizar la plataforma LAMP.

Índice

Índice

Capítulo 1 - Fundamentación Teórica.....	22
1.1 Introducción.....	22
1.2 Las TIC en la Educación	22
1.3 Descripción actual de la situación existente en la UCI con los sitios Web de las distintas asignaturas docentes.	23
1.3.1 Descripción de algunas de las Web Docentes existentes en la UCI.....	24
1.3.2 Semejanzas y diferencias de los sitios Web de las asignaturas en la UCI.....	25
1.4 Análisis de algunas Web Docentes existentes en otras universidades del mundo.	26
1.4.1 Criterio acerca de la Web de la Tecnología Educativa.....	27
1.5 Análisis de las soluciones existentes con respecto a la propuesta.....	28
1.6 Arquitectura Cliente-Servidor	30
1.7 Tecnologías propuestas para la solución del problema.....	31
1.7.1 .NET.....	31
1.7.3 LAMP	33
1.7.4 Comparación entre las tecnologías.....	35
1.8 Tecnología LAMP.....	39
1.8.1 Linux (El Sistema Operativo)	39
1.8.2 Apache (El Servidor Web).....	40
1.8.3 MySQL (Sistema Gestor de Base de Datos)	40
1.8.4 Personal Home Page (El lenguaje de programación) PHP.....	41
1.9 Metodología seleccionada en el proceso de desarrollo del software.	41
1.9.1 Proceso Unificado de Desarrollo (RUP)	42
1.9.2 Lenguaje Unificado de Modelado	43
1.9.3 Rational Rose.....	45
1.10 Conclusiones.....	46
Capítulo II Descripción de la solución propuesta	48
2.1 Introducción.....	48
2.2 Modelo del negocio actual	48
2.3 Modelo de Dominio	48
2.4 Introducción a la descripción del modelo de Dominio.....	49
2.4.1 Descripción del Modelo de Dominio	49
2.5 Requisitos funcionales	50
2.6 Requerimientos no Funcionales	52
2.7 Descripción del modelo propuesto	54
2.8 Definición de los casos de uso del sistema.....	55
2.8.1 Definición de los actores del sistema	56
2.8.2 Diagrama de casos de usos del sistema.....	58
2.8.3 Descripción de los casos de uso del sistema	59

2.9 Conclusiones.....	86
Capítulo III Construcción de la solución propuesta.....	88
3.1 Introducción.....	88
3.2 Modelo de Análisis.....	88
3.2.1 Clases del modelo de análisis.....	88
3.3 Diagramas de Interacción	89
3.3.1 Diagramas de Colaboración	89
3.3.2 Diagramas de Secuencia	89
3.3 Diagramas de clases de diseño.....	89
3.4 Principios de diseño	90
3.5 Tratamiento de errores.....	90
3.6 Diseño de la base de datos.....	92
3.6.1 Diagrama de clases persistentes	92
3.6.2 Modelo de Datos	93
3.7 Diagrama de despliegue	93
3.8 Diagrama de Componentes	94
3.9 Conclusiones.....	98
Capítulo IV Estudio de factibilidad.....	100
4.1 Introducción.....	100
4.2 Estimación de Costos	100
4.2.1 Planificación	101
4.2.1 Costos.....	105
4.3 Beneficios tangibles e intangibles.....	108
4.4 Análisis de Costo y Beneficios	108
4.5 Conclusiones.....	109
Conclusiones	111
Recomendaciones	113
Referencias Bibliográficas.....	115
Bibliografía.....	117
Glosario de Términos	119
Anexos.....	121

Índice de Figuras

Fig.2.1 Modelo de Dominio.....	49
Fig.2.2 Diagrama de casos de uso del sistema.....	58
Fig.3.1 Diagrama de clases persistentes	92
Fig.3.2 Diagrama de despliegue.....	94
Fig.3.3 Módulo General	95
Fig.3.4 Módulo Gestión-Profesores.....	95
Fig.3.5 Módulo Administración.....	96
Fig.3.6 Módulo acceso-Informaciones	97
Fig.3.7 Módulo Reportes	97

Índice de Tablas

Tabla 2.1 Caso de uso 1: Autenticar Usuario	59
Tabla 2.2 Caso de uso 2: Gestionar usuarios con roles.....	61
Tabla 2.3 Caso de uso 3: Gestionar noticias	64
Tabla 2.4 Caso de uso 4: Gestionar Documentos Docentes	67
Tabla 2.5 Caso de uso 5: Manipular Documentos Docentes	69
Tabla 2.6 Caso de Uso 6: Ver Noticias Generales.....	70
Tabla 2.7 Caso de uso 7: Acceder a informaciones específicas	71
Tabla 2.8 Caso de uso 8: Crear Encuesta.....	72
Tabla 2.9 Caso de uso 9: Solicitar Encuesta.....	73
Tabla 2.10 Caso de uso 10: Ver resultados de Encuesta.....	74
Tabla 2.11 Caso de uso 11: Crear auto-evaluación.....	75
Tabla 2.12 Caso de uso 12: Solicitar auto-evaluación	77
Tabla 2.13 Caso de uso 13: Ver resultados de auto-evaluaciones	78
Tabla 2.14 Caso de uso 14: Acceder a sitios de navegación básica	79
Tabla 2.15 Caso de uso 15: Sugerir sitios de navegación básica.....	80
Tabla 2.16 Caso de uso 16: Aprobar solicitudes de profesores privilegiados	82
Tabla 2.17 Caso de uso 17: Enviar sugerencias al administrador del sitio.....	83
Tabla 2.18 Caso de uso 18: Obtener Reportes.....	85
Tabla 4.1 Entradas Externas al sistema.....	101
Tabla 4.2 Salidas Externas	102
Tabla 4.3 Peticiones	102
Tabla 4.4 Ficheros Lógicos Internos.....	104
Tabla 4.5 Obtención de los puntos de función sin ajustar	104
Tabla 4.6 Instrucciones fuentes estimadas.....	105
Tabla 4.7 Factores de Escala.....	106
Tabla 4.8 Multiplicadores de Esfuerzo	106
Tabla 4.9 Resumen de los resultados del estudio de factibilidad del proyecto.....	107

INTRODUCCIÓN

Introducción

En la UCI, desde su creación cada asignatura posee un sitio Web donde se encuentra información relacionada con esta. En ellos encontramos de forma general los materiales de estudio (programas, secuencias de actividades, conferencias, clases prácticas, clases teórico prácticas y su sistema de evaluación), permitiendo la descarga de estos materiales.

Es común encontrar que los sitios incluyan noticias, información acerca del claustro de profesores de la asignatura, concursos en línea, convocatorias, que presenten vínculos con otras páginas importantes y enlaces a los servicios que ofrece la intranet universitaria como puede ser el correo, superación, consejos de especialistas y otros.

Aunque todavía no es frecuente en algunos sitios se incluyen temas relacionados con el trabajo científico-metodológico, estrategias curriculares, extensión universitaria, centros virtuales, búsquedas, en general temas vinculados al trabajo integral del profesor.

Esta cantidad de información a manejar por los usuarios y es de variada complejidad, lo que hace difícil que un usuario pueda elaborar sus propias estrategias de búsqueda de información en los diferentes sitios.

Los actuales sitios no permiten tampoco la manipulación de las estadísticas que son una forma de medir determinados aspectos de su utilización por parte de alumnos y profesores. Esto puede ser importante, y ayudar a resolver algunas interrogantes a las que hoy no se les puede dar respuesta. Otro problema es que no hay métricas de calidad para medir la accesibilidad de los sitios y su posicionamiento dentro del resto de los sitios de la universidad.

Ninguno de los sitios dedicados a mantener la información de las diferentes asignaturas logran tener toda la información concentrada y mostrada de forma coherente e intuitiva, que le permita a sus usuarios habituales encontrar la

información que necesita rápidamente y garantizar la retroalimentación necesaria para los dirigentes docentes, profesores y los propios estudiantes

Por tanto el **problema** a resolver queda formulado con la siguiente interrogante:

¿Cómo diseñar una Web Docente que logre concentrar y mostrar de forma coherente y flexible la información referente a una asignatura y que le permita a todos sus usuarios satisfacer sus necesidades?

El **objeto de estudio** se enmarca en la situación existente con los sitios Web referidos a las diferentes asignaturas en la UCI en lo que respecta obtención y utilización de forma flexible de la información necesaria a los usuarios, así como la necesidad de otras informaciones importantes para ayudar a los dirigentes docentes.

El **campo de acción** lo constituye el mejoramiento de los mecanismos para ofertar toda la información necesaria a los distintos usuarios, mediante el diseño de una Web Docente flexible.

El **objetivo general** de esta investigación es realizar el análisis de las necesidades vinculadas a la publicación de los materiales docentes, llegando al diseño de una Web Docente que pueda brinde la información de las asignaturas de una forma flexible y que se integre con el resto de los servicios de la intranet universitaria, brindando un nivel de satisfacción adecuado a todos sus usuarios.

Para darle cumplimiento al objetivo general nos trazamos los siguientes **objetivos específicos**:

- 1) Realizar el análisis de las necesidades vinculado a la publicación de los materiales docentes.
- 2) Realizar un diseño que permita obtener una aplicación flexible para que los usuarios queden satisfechos.

- 3) Brindar los mecanismos que permitan obtener estadísticas que sirvan para medir la utilización de la web docente por alumnos y profesores.

Con vistas al cumplimiento de los objetivos anteriores se propone la realización de las siguientes tareas:

- Caracterizar los sitios docentes existentes en la actualidad en la Universidad.
- Elaborar la fundamentación teórica del objeto de estudio.
- Caracterizar otras Web docentes a nivel internacional.
- Seleccionar la Metodología de Análisis y Diseño de Sistemas Informáticos, que garantice la calidad del sistema.
- Seleccionar la plataforma y herramientas en la que se desarrollará la aplicación, fundamentando dicha elección.
- Diseñar una Base de datos que soporte las funcionalidades necesarias al sistema.

Con el diseño de este sistema pretendemos lograr con su futura implementación un modelo de Web Docente que pueda ser utilizada por las distintas asignaturas de la UCI, logrando así uniformidad entre los sitios, por tanto el usuario teniendo un modelo ya sabrá a donde dirigirse para la búsqueda de información, cumpliendo así sus expectativas. Con la creación de esta herramienta, se logrará flexibilizar al máximo toda la información, para la satisfacción de todos los usuarios no solo en la UCI, sino en todos aquellos lugares que requieran de la utilización de una Web Docente.

El presente documento se estructura en resumen, introducción, cuatro capítulos, conclusiones, recomendaciones, referencias bibliográficas, bibliografía y anexos donde se incluye todo lo relacionado con el trabajo investigativo realizado.

En el **Capítulo I: Fundamentación Teórica**, se analizan las Tecnologías de la Información y la Educación (TIC), se describen las Web Docentes existentes en la Universidad de las Ciencias Informáticas, y se analizan Web Docentes de otras Universidades del mundo, así como el criterio de dos estudiosos del tema. Además se analizan las tendencias y tecnologías actuales a nivel mundial y que pueden ser de gran utilidad en el desarrollo de la propuesta de solución. Se tienen en cuenta las metodologías de desarrollo de software, así como la necesidad de la utilización del software libre.

En el **Capítulo II: Descripción de la solución propuesta**, se aborda lo referente al modelado del sistema actual a través del modelo de dominio. Se plantean los requisitos funcionales y no funcionales de la aplicación a desarrollar, y a partir de estos se hace la modelación del sistema, determinándose los casos de uso, y los actores que interactúan con este.

En el **Capítulo III: Construcción de la propuesta de solución**, se describe la forma en que se realizará la implementación a través de la modelación de los diagramas de diseño y despliegue, y el modelo de datos. Se especifican los principios a utilizar para el diseño gráfico y el tratamiento de errores.

En el **Capítulo IV: Estudio de la factibilidad**, se estudia la factibilidad del sistema, se enumeran los beneficios tangibles e intangibles y se analizan los costos que representan la elaboración del mismo.

Capítulo 1

Fundamentación teórica

Capítulo 1 Fundamentación Teórica.

1.1 Introducción

En este capítulo se analiza la relación de la Web docente con las Tecnologías de la Información (TIC) y la Educación; se describen las Webs Docentes existentes actualmente en la UCI, identificándose así los principales problemas que motivan esta investigación. Se caracterizan sitios Web que existen en otras Universidades del mundo buscando puntos de contacto y comparación, además de poder ofrecer novedad en el tema.

Se exponen las tendencias y tecnologías actuales a nivel mundial y que pueden ser de gran utilidad en el desarrollo de la propuesta de solución. Se tienen en cuenta las metodologías de desarrollo de software, así como la necesidad del software libre como cuestión estratégica de nuestro país. Al final se selecciona como metodología RUP, y para la implementación se selecciona la plataforma LAMP, siendo las tecnologías más apropiadas teniendo en cuenta que cumplen con las necesidades del usuario, de la universidad y del país en general.

1.2 Las TIC en la Educación

Una Web Docente es un sitio Web que cuenta con características que ayudan a los alumnos que lo visitan a alcanzar objetivos previamente definidos, para que al interactuar con este logren incorporar determinados conceptos, manejen con soltura ciertos procedimientos y/o hayan adquirido o afianzado ciertas actitudes.

Las TIC son incuestionables, forman parte de la cultura tecnológica que nos rodea y con la que debemos convivir, amplían nuestras capacidades físicas y mentales y las posibilidades de desarrollo social.

Entre los aportes mas significativos del uso de las TIC se encuentran:

- **Fácil acceso a una inmensa fuente de información.**
- Proceso rápido y fiable de todo tipo de datos.
- **Canales de comunicación inmediata.**
- Capacidad de almacenamiento.

- Automatización de trabajos.
- **Interactividad.**
- **Digitalización de toda la información.**

Entre las formas básicas del uso de las TIC en la educación definimos las siguientes que son de nuestro interés: [1]

- Las TIC para aprender SOBRE las TIC.
- Aprender DE las TIC en el aula de informática.
- Aprender DE y CON las TIC.
- **Aprender CON las TIC.**

En esta última forma es donde se encuentra ubicado nuestro objetivo, la creación de la Web Docente de una asignatura, ya que aquí las TIC se utilizan como complemento de las clases presenciales (o como espacio virtual para el aprendizaje), por lo que podemos considerar que entramos en el ámbito del aprendizaje distribuido, planteamiento de la educación centrado en el estudiante que, con la ayuda de las TIC posibilita el desarrollo de actividades e interacción tanto en tiempo real como asíncronas. Los estudiantes utilizan las TIC cuando quieren y donde quieren (máxima flexibilidad) para acceder a la información, para comunicarse, para debatir temas entre ellos o con el profesor, para preguntar, para compartir e intercambiar información.

1.3 Descripción actual de la situación existente en la UCI con los sitios Web de las distintas asignaturas docentes.

En una universidad como la UCI, donde se estudia Ingeniería Informática, carrera que por su constante desarrollo exige mantenerse actualizada, resulta indispensable la existencia de Web docentes para las distintas asignaturas que mantengan al usuario (estudiante y/o profesor) con la bibliografía al día. Resulta importante además que estos usuarios puedan desarrollar un estándar de trabajo derivado de la homogeneidad de todos los sitios para facilitarle el manejo y adquisición de información al usuario.

En la actualidad en la UCI la mayoría de los departamentos poseen uno o varios sitios webs desde donde se pueden acceder a las diferentes asignaturas que lo conforman, en estos sitios se puede encontrar la información docente referida a la los materiales de estudio (conferencias, clases prácticas, clases teóricos prácticas, programa de evaluación), y se permite la descarga de estos; también estos sitios incluyen noticias, concursos, información acerca de los profesores, vínculos con otras páginas importantes, ellos tienen como objetivo satisfacer al usuario en cuanto a obtención de información relacionada con una asignatura del currículo.

Encontramos que todos los departamentos no poseen una página principal, desde donde se pueda acceder con facilidad a las asignaturas que el departamento coordina, entre estos se pueden mencionar: Departamento de Ingeniería y Gestión de Software, y el Departamento de Técnicas de Programación que tienen la particularidad de tener un sitio directo para cada una de las asignaturas que están integradas a ellos.

1.3.1 Descripción de algunas de las Web Docentes existentes en la UCI

Idiomas Extranjeros:

Es uno de los sitios donde encontramos más variedad de información referida al trabajo científico-metodológico, estrategias de extensión, centros virtuales (en cada uno de estos parámetros particularidades como permitir realizar búsquedas, accesos a diferentes tipos de informaciones). Está confeccionado bajo el precepto de una enseñanza semi-presencial.

Matemática:

Es un sitio donde a partir de un menú se puede acceder a cualquier asignatura desde cualquier otra y en esta descargar las informaciones referentes a ella. Existen concursos, informaciones importantes relacionadas con la materia, noticias, se especifica el método de evaluación y se colocan los objetivos para las pruebas parciales a medida que corresponden. Además posee un área de descarga que tiene los softwares específicos de la asignatura.

Técnicas de Programación:

En este departamento no existe una página central que reúna a todas las asignaturas que pertenecen a él, sino que por cada una de ellas encontramos un sitio Web independiente del otro. En general todas presentan noticias y avisos, contenidos, cantidad de visitas realizadas al sitio, bibliografía y documentos para los cursos; nos encontramos que existen dos variantes del sitio con las mismas funcionalidades.

Sistema Digitales:

Es un sitio donde se selecciona una asignatura se pueden bajar los materiales docentes, se le ofrece al usuario un Foro para debatir temas acerca de las asignaturas; además de lo anterior permite la descarga de algunos programas que se utilizan en la asignatura, contiene tutoriales sobre diferentes temas, información sobre los profesores a través de una base de datos, leer noticias de las asignaturas, conocer las visitas recibidas y cantidad de usuarios que están conectados actualmente al sitio.

Ingeniería y Gestión de Software:

Es un departamento que cuenta con sitios para tres de sus asignaturas, los sitios son independientes para cada una de las asignaturas, todos los sitios tienen el mismo diseño, desde ellos se puede acceder a la intranet, a Inter-nos, a sistios de referencia sobre UML. Los sitios cuentan con un sistema para su administración, noticias, información acerca de los proyectos e información acerca del claustro de profesores.

Como se puede ver los sitios tienen la mas variada información, y están conformados de diferente forma.

1.3.2 Semejanzas y diferencias de los sitios Web de las asignaturas en la UCI

Entre los sitios Web existentes hay similitudes, podemos plantear que todos responden en sentido general a la presencia de los siguientes parámetros:

- Materiales de las asignaturas.

- Posibilidad de poder descargar los materiales.
- Calendario de pruebas.
- Sistema de evaluación de la asignatura.
- Bibliografía.
- Muchas poseen un sistema para su administración
- Se puede conocer en algunos de ellos la cantidad de visitas realizadas, no se detalla otro tipo de información.

También existen marcadas diferencias, fundamentalmente referidas a:

1. No todos poseen la misma información, ni esta organizada de la misma forma.
2. No son iguales de flexibles, unos brindan mayores posibilidades.
3. No existe un estándar de trabajo que se cumpla para su elaboración y que permita a los usuarios identificar las funcionalidades comunes y trabajar más eficientemente con ellos.

Por lo anteriormente expuesto y por entrevistas realizadas a algunos directivos docentes de la Universidad podemos decir que las Web docente existentes actualmente en la UCI no satisfacen totalmente a todos los tipos de usuarios.

1.4 Análisis de algunas Web Docentes existentes en otras universidades del mundo.

Después de realizar una búsqueda profunda en Internet, no pudimos tener acceso al sitio específico de una asignatura en particular de alguna universidad del mundo, pero si pudimos analizar el estado de varios de los departamentos, analizando las características de algunos sitios dedicados al tratamiento de información de asignaturas de un departamento en general, podemos comentar que:

- **Departamento de Arquitectura y Tecnología de Computadores de la Universidad de Granada**, un sitio bastante flexible que posee noticias, webmail, presentación, docencia, investigación, personal, premios, enlaces, administración y vínculos con páginas principales de la Universidad y otros departamentos y facultades.

- **Departamento de Lenguajes y Computación de la Universidad de Almería**, posee como información: miembros del departamento, grupo de investigación, reglamento, enlaces interesante, docencia, en este último encontramos los programas de las asignaturas y doctorados.
- **Departamento de Sistemas Informáticos-UCLM del Centro de Datos de Albacete**. Está separado por links (biblioteca, cosas generales), alumnos (prácticas, notas, correo, perfil, red campus, directorio), estudios (actividades, investigación, estadísticas, tutorías, planes de estudio), general (inicio, información administrativa). Ofrece un buscador interno, noticias y actividades a realizar.
- **Departamento de Lenguajes y Ciencias de la Computación de la Universidad de Málaga**, posee como información: Dirección, personal, docencia, publicaciones, investigación, asignación de docencia del curso actual, proyecto fin de carrera, tablón de anuncios.

1.4.1 Criterio acerca de la Web de la Tecnología Educativa.

Hay que tener en cuenta que aunque no tuvimos acceso a los sitios específicos de las asignaturas en toda su profundidad, sí contamos con el criterio del Dr. Pere Marquès Graells, autor de un estudio relacionado con la utilización de la Web como tecnología educativa. En este estudio el autor nos muestra que los principales contenidos de una página Web de una asignatura deben ser:

1. Presentación de la asignatura
2. Temario de contenidos y actividades de aprendizaje
3. Metodología docente y sistema de evaluación
4. Canales de comunicación interpersonal

En la presentación de la asignatura se especifica especialmente la contextualización (la importancia de la asignatura en los estudios que se están realizando, el perfil de los alumnos a los que está destinada.); los objetivos o finalidades que se pretenden conseguir, y los conocimientos previos necesarios que deben tener los estudiantes. En el temario de contenidos y actividades de

aprendizaje, se enumeran los diversos temas y se presenta un guión de sus apartados principales; se incluyen los principales apartados de cada tema a modo de guión sintético; los documentos; los materiales didácticos; la propuesta de ejercicios y actividades de aprendizaje que se pueden realizar; la bibliografía específica; enlaces a otras páginas Web de interés, y también se podrán incluir enlaces a páginas Web que muestren actividades de aprendizaje realizadas por los alumnos de cursos anteriores.

En cuanto a la metodología docente y sistema de evaluación, se indicarán los recursos y la metodología que se utilizarán; el sistema de evaluación: las actividades que se tendrán en cuenta para la evaluación de los aprendizajes (individuales y grupales) de los estudiantes, criterios de valoración, calendario, y también se puede elaborar una Web de exámenes ejemplos, con modelos de examen de cursos anteriores. Y por último en los canales de comunicación interpersonal, se pueden incluir las orientaciones para las tutorías presenciales y on-line; los enlaces a foros generales de estudiantes y profesores o para discutir temas concretos; el listado de alumnos con su e-mail y su Web personal, y un tablón de anuncios del profesor, donde el profesor informa a sus alumnos sobre cuestiones de interés relacionadas con la asignatura.

Hay que tener en cuenta que algunas Web de asignaturas solamente incluyen algunos de estos apartados. Las más sencillas presentan solamente un *plan docente* (objetivos, contenidos, metodología, sistema de evaluación) más o menos completo sin presentar actividades ni recursos de aprendizaje.

1.5 Análisis de las soluciones existentes con respecto a la propuesta.

Para llegar a realizar un análisis final tenemos que contar con los aspectos que deben estar adecuadamente ubicados, y actualizados en la Web de una asignatura. Para esto hemos analizado todos los sitios antes mencionados, el criterio del Dr. Pere Marquès Graells, autor de reconocido prestigio internacional, y el criterio del Doctor cubano Pedro Hurrutiner Silva, que plantea:

“Cuando una universidad dispone de una aceptable intranet, y logra generalizar el empleo de un recurso de este tipo, el proceso de formación dispone de un valioso instrumento para su perfeccionamiento”.

El autor además plantea que: “La presencia de una asignatura en la intranet de la universidad supone que en una determinada plataforma estén adecuadamente ubicados y actualizados aspectos tales como:

- Programa de la asignatura
- Calendario para el desarrollo de la asignatura en el semestre
- Notas del profesor.
- Principales textos y otros materiales de consulta disponibles en soporte magnético.
- Guías de estudio para clases prácticas, seminarios y otros tipos de actividades practicas, caso de utilizarse en la asignatura.
- Noticias de la asignatura.
- Debates.
- Evaluaciones para controlar el autoaprendizaje de los estudiantes.
- Direcciones y/o resúmenes de Internet con información adicional sobre temas de la asignatura.
- Registro de los resultados de las evaluaciones por la red.
- Control del acceso de los estudiantes de la asignatura.”

Luego de un exhaustivo análisis de los criterios anteriores y de los diferentes sitios Web docentes de la UCI, como de otras universidades del mundo; se llega a la conclusión que aunque se le brinda diversidad de información a los usuarios, son varios los aspectos que aún no se contemplan en estas Web, y que son necesarios para su mejor funcionamiento y para lograr el cumplimiento de los objetivos propuestos. Entre otros elementos se tienen como principales aspectos que no se permite medir la accesibilidad al sitio, actualmente no se controla el acceso de los estudiantes a la asignatura; no se lleva un registro de los resultados de las evaluaciones por la red; no se realizan a través de ninguno de los sitios evaluaciones para controlar el auto aprendizaje de los estudiantes;

aunque justo es señalar que este no es el objetivo fundamental de una web docente.

El desarrollo del presente trabajo va encaminado a realizar el análisis y diseño una Web Docente que cumpla con los aspectos fundamentales tratados por el Dr. Pedro Hurrutiner Silva y obtener una Web Modelo para todas las asignaturas docentes de la UCI y de otras universidades que así lo requieran.

1.6 Arquitectura Cliente-Servidor

La arquitectura cliente/servidor es un modelo para el desarrollo de sistemas de información en el que las transacciones se dividen en procesos independientes que cooperan entre sí para intercambiar información, servicios o recursos. Se denomina cliente al proceso que inicia el diálogo o solicita los recursos y servidor al proceso que responde a las solicitudes. [3]

En este modelo las aplicaciones se dividen de forma que el servidor contiene la parte que debe ser compartida por varios usuarios, y en el cliente permanece sólo lo particular de cada usuario.

Los clientes realizan generalmente funciones como:

- Manejo de la interfaz de usuario.
- Captura y validación de los datos de entrada.
- Generación de consultas e informes sobre las bases de datos.

Por su parte los servidores realizan, entre otras, las siguientes funciones:

- Gestión de periféricos compartidos.
- Control de accesos concurrentes a bases de datos compartidas.
- Enlaces de comunicaciones con otras redes de área local o extensa.

Siempre que un cliente requiere un servicio lo solicita al servidor correspondiente y éste le responde proporcionándolo. Normalmente, pero no necesariamente, el cliente y el servidor están ubicados en distintos procesadores. Los clientes se

suelen situar en ordenadores personales y/o estaciones de trabajo y los servidores en procesadores departamentales o de grupo. [3]

Entre las principales características de la arquitectura cliente/servidor se pueden destacar las siguientes: [3]

- El servidor presenta a todos sus clientes una interfaz única y bien definida.
- El cliente no necesita conocer la lógica del servidor, sólo su interfaz externa.
- El cliente no depende de la ubicación física del servidor, ni del tipo de equipo físico en el que se encuentra, ni de su sistema operativo.
- Los cambios en el servidor implican pocos o ningún cambio en el cliente.

1.7 Tecnologías propuestas para la solución del problema

Para darle solución al problema abordado en este trabajo, se ha realizado un estudio de tres de las tecnologías existentes para la realización de aplicaciones Web, y que son las más extendidas en la actualidad:

- .NET (Microsoft),
- Java 2 Platform, Enterprise Edition (Sun Microsystems),
- LAMP (Open Sorce, combina varias empresas).

1.7.1 .NET

.NET es un proyecto de Microsoft para crear una nueva plataforma de desarrollo de software con énfasis en transparencia de redes, con independencia de plataforma y que permita un rápido desarrollo de aplicaciones. Basado en esta plataforma, Microsoft intenta desarrollar una estrategia horizontal que integre todos sus productos, desde el Sistema Operativo hasta las herramientas de mercado.

Con esta Plataforma Microsoft incursiona de lleno en el campo de los Servicios Web y establece el XML como norma en el transporte de información en sus

productos y lo promociona como tal en los sistemas desarrollados utilizando sus herramientas.

.NET intenta ofrecer una manera rápida y económica pero a la vez segura y robusta de desarrollar aplicaciones - o como la misma plataforma las denomina, soluciones - permitiendo a su vez una integración más rápida y ágil entre empresas y un acceso más simple y universal a todo tipo de información desde cualquier tipo de dispositivo.

El "framework" o marco de trabajo, constituye la base de la plataforma. Net y denota la infraestructura sobre la cual se reúnen un conjunto de lenguajes, herramientas y servicios que simplifican el desarrollo de aplicaciones en entorno de ejecución distribuido.

Debido a la publicación de la norma para la infraestructura común de lenguajes .NET soporta ya más de 20 lenguajes de programación y es posible desarrollar cualquiera de los tipos de aplicaciones soportados en la plataforma con cualquiera de ellos, lo que elimina las diferencias que existían entre lo que era posible hacer con uno u otro lenguaje. Algunos de los lenguajes desarrollados para el marco de trabajo.NET son: C#, Visual Basic, C++ , J# , Perl, Python , Fortran y Cobol.NET.

1.7.2 Java 2 Platform Enterprise Edition

Java 2 Platform Enterprise Edition (J2EE) es un conjunto de especificaciones de APIs Java para la construcción de aplicaciones empresariales. La mayor parte de las abstracciones de las APIs corresponden a interfaces y clases abstractas. Existen múltiples implementaciones de distintos fabricantes, incluso algunas OpenSource. Una aplicación construida con J2EE no depende de una implementación particular.

Es un framework estándar ampliamente usado en el mundo de la industria, del que existen numerosas implementaciones comerciales y gratuitas, para la construcción de aplicaciones empresariales (escalables, transaccionales y

seguras) con Java. J2EE está basado en la arquitectura del lado del servidor (Served-based). Este tipo de arquitectura concentra la mayoría de los procesos de la aplicación en el servidor o en un pedazo de este. La plataforma completa puede ser implementada en un solo sistema, o la plataforma de servicios puede ser distribuida a través de varios sistemas, pero todas las APIs especificadas deben ser incluidas en alguna parte del sistema completo.

J2EE proporciona, entre otras, las siguientes APIs: JDBC (acceso a bases de datos relacionales), APIs para XML (representación de contenido de datos), servlets+JSP+JSTL (capa vista de una aplicación Web) y EJB (componentes distribuidos, transaccionales, seguros y escalables que encapsulan la lógica de negocio). Ofrece un conjunto de especificaciones y técnicas que proporcionan soluciones completas, seguras, estables y escalables para el desarrollo, despliegue y gestión de aplicaciones en múltiples niveles de funcionalidad basadas en servidores. J2EE reduce el coste y la complejidad de desarrollo, lo cual redundará en rapidez de desarrollo.]

1.7.3 LAMP

LAMP es el acrónimo de **L**inux, **A**pache, **M**ySQL y (**P**HP, **P**erl o **P**hyton), consideradas como unas de las mejores combinaciones que el software libre puede proporcionar y que permiten a cualquier organización o individuo tener un servidor Web versátil y poderoso, independientemente del hecho que no es necesario pagar licencias por su utilización, su mantenimiento se reduce a actualizar paquetes que se pueden descargar por Internet y su nivel de seguridad es muy bueno, al liberarse parches de seguridad al muy poco tiempo que se declara una alerta.

Una característica muy interesante es el hecho que estos cuatro productos pueden funcionar en una amplia gama de *hardware*, con requerimientos relativamente pequeños pero que no por eso dejan de ser menos estables que en equipos de grandes capacidades [11], además que los montos de inversión y el tiempo requerido con LAMP son bajos, al tiempo que los beneficios finales son

duraderos al no tener que actualizar todo el sistema cada cierto tiempo y tener alta disponibilidad y seguridad.

Todos los elementos que forman LAMP son software libre, de modo que disfrutan de las siguientes ventajas propias del mismo: Libertad de copia y distribución. Se puede conseguir gratuitamente en Internet. Hay variadas fuentes donde conseguir cualquiera de las distribuciones. Libertad de modificación. Junto a los programas ejecutables, se puede obtener su código fuente. Esto, si se tienen los conocimientos necesarios, permite verificar la seguridad y eficiencia de los mismos, además de modificar y/o añadir las características y comportamientos que deseemos.

1.7.4 Comparación entre las tecnologías

Comparación entre las Tecnologías.

Tecnologías	Ventajas	Desventajas
.NET	<ul style="list-style-type: none"> • Código administrado: El CLR controla los recursos del sistema para que la aplicación se ejecute correctamente. • Interoperabilidad multilenguaje: El código puede ser escrito en cualquier lenguaje de la plataforma • Compilación just-in-time: Con esto se aumenta el rendimiento de la aplicación al ser específico para cada plataforma. • Seguridad: Se puede especificar que una pieza de código tenga permisos de lectura de archivos pero no de escritura. 	<ul style="list-style-type: none"> • Dependencia de un solo proveedor. • Por ser un cambio muy fuerte en arquitectura, puede contener los problemas de primeras versiones. • Al poder combinar múltiples lenguajes, puede dar lugar a código mantenible sólo por ciertas personas. • Poco reaprovechamiento de la experiencia de recursos humanos especializados en Microsoft, ya que cambia drásticamente la plataforma • Monopoliza las herramientas de desarrollo.

Tecnologías	Ventajas	Desventajas
J2EE	<ul style="list-style-type: none">• Está soportado por más de 30 proveedores certificados en el mercado.• Tiene más antigüedad en el mercado que el .NET y el LAMP.• Mejor historia de integración con sistemas legados.• Es más adaptable a cambios en hardware• Modelo avanzado de programación• Buena portabilidad a otros servidores de aplicaciones.• Amplia variedad de herramientas de desarrollo.	<ul style="list-style-type: none">• En una solución por el uso de softwares de varios proveedores, puede haber problemas de integración. Se recomiendan soluciones de un solo proveedor.• Herramientas poco integradas entre proveedores.• Las aplicaciones de otros lenguajes se consideran fuera de la plataforma.• Configuración del entorno compleja• Ideas generales complejas.• Para aplicaciones de escritorio múltiples máquinas virtuales.

Tecnologías	Ventajas	Desventajas
LAMP	<ul style="list-style-type: none"> • Soporte a gran cantidad de arquitecturas, Intel y compatibles, SPARC, MIPS y PPC. • Código relativamente sencillo y con pocos cambios de una plataforma a otra, garantizando la portabilidad. • Parches generados en poco tiempo después de encontrarse un agujero de seguridad. • Actualizaciones del software vía Internet. • Posibilidad de incrementar los servicios y funciones desde el código fuente, lo que mejora el rendimiento de las aplicaciones. [11] 	<ul style="list-style-type: none"> • Es muy distinto de Windows, lo que dificulta el trabajo a quienes estén acostumbrados a él. • Las actualizaciones requieren de conocer ciertos aspectos generales de la arquitectura y en ocasiones tener conocimientos profundos del sistema. • Configurar algunos servicios de red requieren de más tiempo que en Windows e incluso se deben parametrizar los equipos. • Es necesario capacitar al personal o contratar personal calificado, lo que implica un incremento en los costos de la administración de personal.

1.7.5 Selección de la plataforma de desarrollo

En general la tecnología Microsoft es suficientemente fiable, la programación en ASP.NET tiene la gran ventaja de permitir la utilización de muchos lenguajes diferentes (C#, VisualBasic, Pascal, Cobol), y SQL Server es un gestor de bases de datos potente, flexible y fiable; en cambio, es una tecnología cara y de software propietario, lo que implica la monopolización de esta por parte de sus productores, aunque es justo destacar que se están realizando esfuerzos como el Proyecto Mono para incluirla dentro de la corriente del software libre.

La arquitectura Java (que incluye JSP, servlets, aplicaciones Java y J2EE) con la combinación de Linux, Apache, Tomcat o Jboss (para J2EE) y posgreSQL es potente y robusta, al mismo precio que la LAMP, y se utiliza mucho en desarrollos profesionales. Tal vez la dificultad en este caso sea que Java es un lenguaje menos conocido y algo más complejo que los usados por los otros dos grandes sistemas.

Por otro lado, la tecnología LAMP es gratuita (todo el software es gratuito en su uso para Internet), y Linux y Apache son posiblemente de lo mejor en sus correspondientes ámbitos. Sin embargo, MySQL tiene algunas limitaciones (no tiene algunas funcionalidades que sí tienen SQL Server o PostgreSQL), aunque resulta muy potente y útil para prácticamente cualquier tipo de aplicaciones. Y en cuanto a PHP, como lenguaje de programación es un lenguaje específico para Internet que cada vez está teniendo más aceptación, y del que podemos encontrar numerosas aplicaciones gratuitas fácilmente adaptables a nuestras necesidades.

Tendencia al Software Libre.

En Cuba el software libre se ha utilizado como una solución para vencer al criminal bloqueo impuesto a nuestro país que sufre por la prohibición del Gobierno de los Estados Unidos, que impide la compra de software a empresas productoras estadounidenses, que ocupan un lugar importante en el mercado internacional. El software libre permite el desarrollo de productos y aplicaciones;

además supone un ahorro de costos fundamentalmente para las escuelas, esta da la libertad de copiar y redistribuir el software, por lo que pueden hacer copias para todas las computadoras que tengan. Como institución de avanzada en el campo de la informática la UCI ha tenido una tendencia hacia la utilización en grado creciente de software libre por lo que partiendo de esta premisa y por la seguridad que implementa el Apache, más la implementada por MYSQL y PHP, unido a la de los servidores Linux que satisfacen completamente los requerimientos de seguridad para el sistema propuesto y por la calidad de los softwares, se ha decidido seleccionar la tecnología **LAMP** como plataforma a utilizar para la implementación de la aplicación.

1.8 Tecnología LAMP

1.8.1 Linux (El Sistema Operativo)

Linux es el núcleo del sistema operativo, que ha ido implementando soporte para una gran parte del hardware actual (USB, cámaras digitales, escáneres, impresoras, grabadoras, redes, etc.). Dicho núcleo viene arropado por librerías y utilidades distribuidas bajo la licencia libre GPL o similares (de aquí la denominación GNU/Linux).

- Su excelente relación calidad-precio le ha granjeado la admiración e incondicional apoyo de muchísimos usuarios alrededor del mundo. Su adopción en el ámbito de los servidores Web ha sido espectacular. Es multitarea, multiusuario, multiplataforma, multiprocesador, tiene protección de la memoria entre procesos, soporta muchísimos tipos de sistemas de archivos, dispone de una amplia variedad de protocolos de red soportados en el núcleo y, finalmente, permite compartir por red ficheros e impresoras, incluso con otros sistemas operativos. La potencia, estabilidad, gratuidad, modificabilidad y portabilidad de Linux lo hace el sistema operativo perfecto, y ya tiene una posición líder en el ámbito de Internet, siendo cuestión de tiempo que la tenga en el resto de ámbitos informáticos.

1.8.2 Apache (El Servidor Web)

Apache es el **servidor Web** por excelencia, con algo más de un 60% de los servidores de Internet confiando en él. Entre sus características más sobresalientes están:

- **Fiabilidad:** Alrededor del 90% de los servidores con más alta disponibilidad funcionan con Apache.
- **Gratuidad:** Apache es totalmente gratuito, y se distribuye bajo la licencia Apache Software License, que permite la modificación del código.
- **Extensibilidad:** se pueden añadir módulos para ampliar las ya de por sí amplias capacidades de Apache. Hay una amplia variedad de módulos, que permiten desde generar contenido dinámico (con PHP, Java, Perl, Python), monitorizar el rendimiento del servidor, atender peticiones encriptadas por SSL, hasta crear servidores virtuales por IP o por nombre (varias direcciones Web son manejadas en un mismo servidor) y limitar el ancho de banda para cada uno de ellos. Dichos módulos incluso pueden ser creados por cualquier persona con conocimientos de programación.

1.8.3 MySQL (Sistema Gestor de Base de Datos)

Existen diferentes arquitecturas para los sistemas de gestión de bases de datos, pero la más extendida, y la que más éxito ha tenido, es la arquitectura relacional. MySQL es un servidor de bases de datos relacionales muy rápido, multiusuario y robusto. Es software libre, publicado bajo la licencia GPL (GNU Public License) y mantenido por la compañía sueca MySQL AB. Este gestor no tiene muchas de las características de los gestores comerciales más importantes, como Oracle, SyBase o SQLServer. No obstante, eso no ha impedido que sea el más indicado para aplicaciones que requieren muchas lecturas y pocas escrituras y no necesiten de características muy avanzadas, como es el caso de las aplicaciones Web. MySQL está disponible para un enorme número de sistemas operativos.

Buena parte de su éxito se debe, sin duda, a formar parte de la tecnología LAMP. El 25 de marzo de 2003 se marcó la versión 4.0.12 como la primera

versión estable de MySQL 4. Este nuevo MySQL introduce grandes mejoras entre las que podemos destacar el soporte de transacciones, claves foráneas (con borrado y actualización en cascada), bloqueo a nivel de fila, caché de consultas, la instrucción UNION y el borrado y actualización multitabla.

1.8.4 Personal Home Page (El lenguaje de programación) PHP.

Es el acrónimo de Hypertext Preprocessor. Es software libre, lo que implica menores costes y servidores más baratos que otras alternativas, a la vez que el tiempo entre el hallazgo de un fallo y su resolución es más corto. Es muy rápido. Su integración con la base de datos MySQL, también veloz, le permite constituirse como una de las alternativas más atractivas para sitios de tamaño medio-bajo. Su sintaxis está inspirada en C. Su librería estándar es realmente amplia, lo que permite reducir los llamados 'costes ocultos', uno de los principales defectos de ASP.

PHP se puede considerar multiplataforma. Funciona en toda máquina que sea capaz de compilar su código, entre ellas diversos sistemas operativos para PC y diversos Unix. El código escrito en PHP en cualquier plataforma funciona exactamente igual en cualquier otra. El acceso a las bases de datos de PHP es muy heterogéneo, pues dispone de un juego de funciones distinto por cada gestor. PHP es suficientemente versátil y potente como para hacer tanto aplicaciones grandes que necesiten acceder a recursos a bajo nivel del sistema como pequeños scripts que envíen por correo electrónico un formulario rellenado por el usuario.

1.9 Metodología y herramientas seleccionadas para el proceso de desarrollo del software.

Antes de comenzar con el desarrollo de un producto de software es imprescindible seleccionar la metodología de desarrollo de software a seguir, con el fin de garantizar la calidad. Si se quiere que el sistema sea robusto y mantenible es necesario que el problema sea analizado y la solución sea cuidadosamente diseñada, se debe seguir un proceso robusto, que incluya las actividades principales. Si se sigue un proceso de desarrollo que se ocupa de

plantear cómo se realiza el análisis y el diseño, y cómo se relacionan los productos de ambos, la calidad aumenta considerablemente, por esta razón seleccionamos como proceso de desarrollo de software a RUP (Rational Unified Process, traducción), que constituye la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos, se utiliza además UML para la confección de los diagramas.

1.9.1 Proceso Unificado de Desarrollo (RUP)

El proceso Unificado de desarrollo, RUP, es un producto de Rational (IBM). Se caracteriza por ser iterativo e incremental, estar centrado en la arquitectura y guiado por los casos de uso. Incluye artefactos (que son los productos tangibles del proceso como por ejemplo, el modelo de casos de uso, el código fuente), y roles (papel que desempeña una persona en un determinado momento, una persona puede desempeñar distintos roles a lo largo del proceso). Esta metodología se basa en UML (Unified Modeling Language) como lenguaje de modelado y tiene la ventaja de venir acompañada de una potente herramienta que soporta todos los procesos básicos de RUP: Rational Rose Enterprise Edition 2003; además garantiza la elaboración de todas las fases de un producto de software orientado a objeto.

EI RUP divide el proceso de desarrollo en ciclos, teniendo un producto final al final de cada ciclo, cada ciclo se divide en fases que finalizan con un hito donde se debe tomar una decisión importante, en cada una de estas fases se lleva a cabo una o varias iteraciones, donde cada una resulta un incremento de la precedente, por lo que se dice que RUP es iterativo e incremental. Las fases son:

- Inicio: se hace un plan de fases, se identifican los principales casos de uso y se identifican los riesgos.
- Elaboración: se hace un plan de proyecto, se completan los casos de uso y se eliminan los riesgos.

- Construcción: se concentra en la elaboración de un producto totalmente operativo y eficiente y el manual de usuario.
- Transición: se implementa el producto en el cliente y se entrena a los usuarios. Como consecuencia de esto suelen surgir nuevos requerimientos a ser analizados.

Los flujos de trabajo del proceso unificado de desarrollo son:

- Modelación del negocio
- Requerimiento
- Análisis y diseño
- Implementación
- Prueba
- El resto de los flujos están asociado con la gestión y la administración del desarrollo de la aplicación.

Con esta tesis, se llegará hasta el flujo de trabajo de análisis y diseño.

1.9.2 Lenguaje Unificado de Modelado

El Lenguaje Unificado de Modelado (UML) es un lenguaje que permite modelar, construir y documentar los elementos que forman un sistema software orientado a objetos. Se ha convertido en el estándar de facto de la industria, debido a que ha sido impulsado por los autores de los tres métodos más usados de orientación a objetos: Grady Booch, Ivar Jacobson y Jim Rumbaugh. Estos autores fueron contratados por la empresa Rational Software Co. para crear una notación unificada en la que basar la construcción de sus herramientas CASE.

Este lenguaje preescribe un conjunto de notaciones y diagramas estándar para modelar sistemas orientados a objetos, y describe la semántica esencial de lo que estos diagramas y símbolos significan. Mientras que ha habido muchas notaciones y métodos usados para el diseño orientado a objetos, ahora los modeladores sólo tienen que aprender una única notación. Esta ha sido ampliamente aceptada debido al prestigio de sus creadores y debido a que incorpora las principales ventajas de cada uno de los métodos particulares en los

que se basa (principalmente Booch, OMT y OOSE). El uso de UML permite integrar con mayor facilidad en el equipo de desarrollo a nuevos miembros y compartir con otros equipos la documentación, pues es de esperar que cualquier desarrollador versado en orientación a objetos conozca y use UML (o se esté planteando su uso). El enfoque que toma es el de un ciclo de vida iterativo incremental, el cual permite una gran flexibilidad a la hora de adaptarlo a un proyecto y a un equipo de desarrollo específicos. El ciclo de vida está dirigido por casos de uso, es decir, por la funcionalidad que ofrece el sistema a los futuros usuarios del mismo. Así no se pierde de vista la motivación principal que debería estar en cualquier proceso de construcción de software: el resolver una necesidad del usuario/cliente.

UML ofrece nueve diagramas que se pueden utilizar para modelar sistemas:

- **Diagramas de Casos de Uso para modelar los procesos 'business'.**
- **Diagramas de Secuencia para modelar el paso de mensajes entre objetos.**
- **Diagramas de Colaboración para modelar interacciones entre objetos a través de mensajes.**
- Diagramas de Estado para modelar el comportamiento de los objetos en el sistema.
- Diagramas de Actividad para modelar el comportamiento de los Casos de Uso, objetos u operaciones.
- **Diagramas de Clases para modelar la estructura estática de las clases en el sistema.**
- Diagramas de Objetos para modelar la estructura estática de los objetos en el sistema.
- **Diagramas de Componentes para modelar componentes.**
- Diagramas de Implementación para modelar la distribución del sistema.

1.9.3 Rational Rose.

En la actualidad es la herramienta CASE más utilizada en el mercado mundial de y es la que se ha seleccionado para usar en este proyecto.

La Corporación Rational ofrece un Proceso Unificado (RUP) para el desarrollo de los proyectos de software, desde la etapa de Ingeniería de Requerimientos hasta la de Pruebas. Para cada una de estas etapas existe una herramienta de ayuda en la administración de los proyectos, Rose es la herramienta del Rational para la etapa de análisis y diseño de sistemas.

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

Esta herramienta propone la utilización de cuatro tipos de modelos para realizar un diseño del sistema, utilizando una vista estática y otra dinámica de los modelos del sistema, uno lógico y otro físico. Permite crear y refinar estas vistas creando de esta forma un modelo completo que representa el dominio del problema y el sistema de software

A continuación los cuatro tipos de modelos:

- **Desarrollo Iterativo:** Utiliza un proceso de desarrollo iterativo controlado, donde el desarrollo se lleva a cabo en una secuencia de iteraciones. Cuando la implementación pasa todas las pruebas que se determinan en el proceso, ésta se revisa y se añaden los elementos modificados al modelo de análisis y diseño. Una vez que la actualización del modelo se ha modificado, se realiza la siguiente iteración.
- **Generador de Código:** Se puede generar código en distintos lenguajes de programación a partir de un diseño en UML.

- **Ingeniería Inversa:** Proporciona mecanismos para realizar la denominada ingeniería inversa, a partir del código de un programa, se puede obtener su diseño.
- **Trabajo en Grupo:** Permite varias personas trabajando a la vez en el proceso iterativo controlado, para ello posibilita que cada desarrollador opere en un espacio de trabajo privado que contiene el modelo completo y tenga un control exclusivo sobre la propagación de los cambios en ese espacio de trabajo.

1.10 Conclusiones

En este capítulo se analizaron las TIC en la educación, se describieron las características de las Web docentes en la UCI determinándose los problemas que presentan, demostrando de esta forma la necesidad del cambio, además se expusieron Web docentes existentes en otras universidades del mundo, y se analizaron dos criterios fundamentales de autores relacionados con el tema para realizar la propuesta de solución.

Se realizó un estudio de las tecnologías para seleccionar la más adecuada a utilizar en el desarrollo de la propuesta, así como algunos conceptos y tendencias que se deben tener en cuenta. Se fundamentó la elección del lenguaje de programación, el sistema gestor de bases de datos, el servidor Web, y la metodología de desarrollo de software; así como el uso de otras tecnologías, determinándose la tecnología LAMP para la implementación, junto a una arquitectura cliente-servidor, y RUP como metodología de desarrollo del software.

Capítulo 2

Descripción de la solución propuesta

Capítulo II Descripción de la solución propuesta

2.1 Introducción

En este capítulo se describe la propuesta de solución ante el problema planteado, se presenta el modelo de dominio como alternativa al modelo de negocio debido a la poca estructuración de los procesos que se describen en el negocio y para mejor comprensión de los requisitos del sistema. Finalmente se plantean los requisitos funcionales y no funcionales de la aplicación a desarrollar y a partir de estos se hace la modelación del sistema, determinándose los casos de uso y los actores que interactúan con este.

2.2 Modelo del negocio actual

En los sitios Web pertenecientes a la disciplina de Técnicas de Programación, específicamente al de Introducción a la Programación acceden estudiantes y profesores (usuario en sentido general) con las mismas posibilidades de consultar los contenidos (cursos de programación, ejercicios extra-clases, otros materiales), como las noticias que se ponen para conocimientos de todos y de variados temas. Al acceder a los documentos de estudio que se encuentran en contenido, tienen la posibilidad de abrirlos (solo lectura) o descargarlos. Existe un administrador del sistema que es quien se encarga de insertar y eliminar la documentación del sitio, y actualizar las noticias.

2.3 Modelo de Dominio

Para la identificación de requisitos y la comprensión del contexto, la metodología RUP propone dos procesos fundamentales: el modelo del negocio y el modelo del dominio. En este sistema debido al bajo nivel de estructuración que presenta el negocio que se está estudiando, se propone un modelo del dominio que está constituido por los conceptos (objetos), y las relaciones entre ellos, esto nos permite de manera visual mostrar al usuario los principales conceptos que se manejan en el dominio del sistema en desarrollo. También ayuda a los usuarios, clientes y desarrolladores e interesados, a utilizar un vocabulario común para poder entender el contexto en que se emplaza el sistema, además para la

captura de requisitos y poder construir un sistema correcto, en el que se necesita tener un firme conocimiento del funcionamiento del objeto de estudio. Este modelo va a contribuir posteriormente a identificar algunas de las clases que se utilizarán en el sistema.

2.4 Introducción a la descripción del modelo de Dominio

El modelo conceptual muestra los conceptos básicos del sistema, sus partes y relaciones. Se realiza a través de un diagrama de clases de UML simplificado, en el cual se representan las clases preliminares y las asociaciones preliminares entre las clases, captura los tipos más importantes de objetos en el contexto del sistema. Los objetos del dominio representan las "cosas" que existen o los eventos que suceden en el entorno en el que trabaja el sistema.

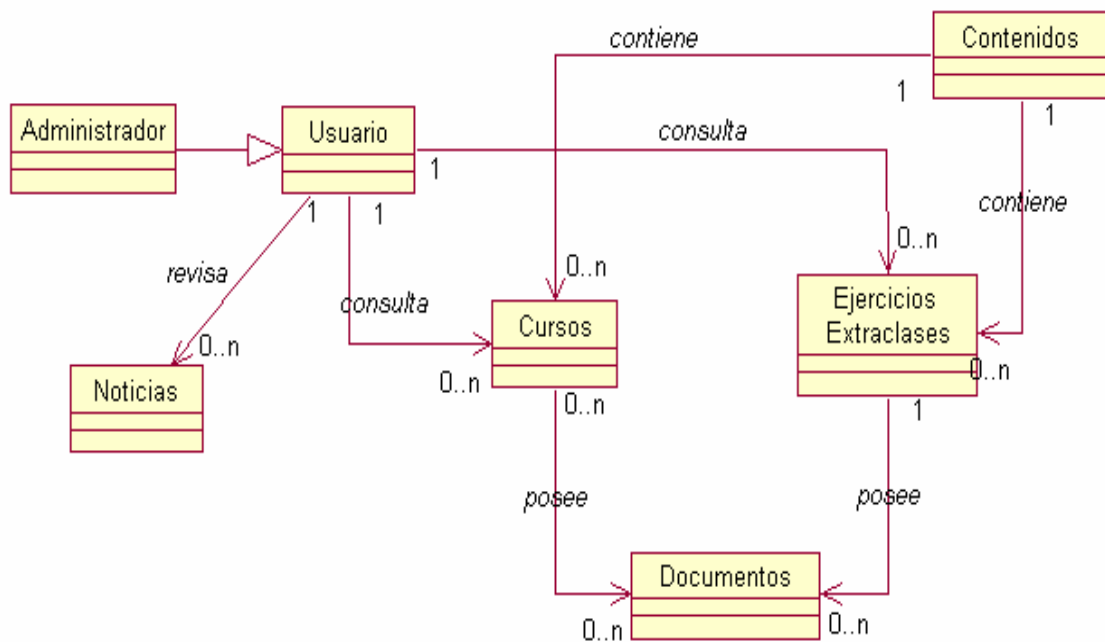


Fig.2.1 Modelo de Dominio

2.4.1 Descripción del Modelo de Dominio

A continuación se definen los conceptos asociados al dominio del problema, los mismos se considerarán niveles de información:

Usuario: La entidad usuario es una generalización de la entidad administrador a los estudiantes y profesores, son quienes interactúan con el sistema, consultando las diferentes secciones.

Administrador: La entidad administrador es una especificación de la entidad usuario. Es quien va a insertar, eliminar o modificar las noticias, cursos, ejercicios extraclases, documentos, teniendo total acceso a la base de datos del sistema.

Noticias: La entidad noticias contiene muchas noticias sobre diferentes temas relacionados con la docencia y la UCI en general, son consultadas por los usuarios.

Contenido: Es una categoría que agrupa a los diferentes cursos, y a los materiales extraclases.

Cursos: Es una entidad que está formada por libros, materiales, ejercicios (documentos) en sentido general, según su clasificación, estos responden a un tema de curso determinado. Son accedidos por los usuarios

Tareas extraclases: Es una entidad que está formada por varios documentos para el estudio de los usuarios

Documentos: Es una entidad que posee una clasificación, y que se puede encontrar según esta en un curso determinado, o en una tarea extraclase.

2.5 Requisitos funcionales

Los requisitos funcionales describen el qué debe hacer el sistema, son las capacidades o funciones que el sistema debe cumplir.

R1- Autenticar usuario y asignarle privilegios al usuario autenticado

1.1 Nombre de usuario

1.2 Contraseña

1.3 Asignar privilegios

R2- Gestionar usuarios con sus roles

R2.1 Insertar usuario y rol

2.1.1 Nombre de usuario

2.1.2 Contraseña

2.1.3 Rol

R2.2 Modificar usuario y rol

2.2.1 Nombre de usuario

R2.3 Eliminar usuario y rol

R3-Gestionar Noticias

R3.1 Insertar Noticias

R3.2 Modificar Noticias

R3.3 Eliminar Noticias

R4-Gestionar documentos docentes

R4.1 Insertar documentos docentes

R4.2 Modificar documentos docentes

R4.3 Eliminar documentos docentes

R5- Manipular documentos

R5.1 Acceder al P1

R5.2 Acceder a materiales extras

R5.3 Acceder al programa de la asignatura

R5.3.1 Acceder al sistema de evaluación de la asignatura

R5.3.2 Acceder al P1

R6- Ver noticias generales

R7- Acceder a informaciones específicas

R8- Crear encuestas

R9- Solicitar encuesta

R9.1 Listar encuesta

R9.2 Seleccionar tipo de encuesta

R10- Obtener resultado de la encuesta

R11- Crear auto-evaluación

R12- Solicitar auto-evaluación

R12.1 Listar auto-evaluaciones

R12.2 Seleccionar tipo de auto-evaluación

R12.3 Obtener evaluación

R13- Obtener resultados de la auto-evaluación

R14- Acceder a sitios de navegación básica

R15- Obtener reportes

R15.1 Obtener reportes de acceso a documentos docentes

R15.2 Obtener reportes de acceso a informaciones específicas

R16- Sugerir sitios de navegación básica

R17- Aprobar solicitudes de profesores privilegiados

R17.1 Aprobar publicar encuestas

R17.2 Aprobar publicar auto-evaluaciones

R17.3 Aprobar publicar sitios de navegación básica

R18- Enviar sugerencias al administrador del sitio

2.6 Requerimientos no Funcionales

Los requerimientos no funcionales son propiedades o cualidades que el producto debe tener, como restricciones del entorno o de implementación, rendimiento, entre otros aspectos; son características que debe cumplir la aplicación como complemento de los requisitos funcionales.

- Requerimiento de Software:

- ❖ Linux como sistema operativo
- ❖ Servidor Web Apache
- ❖ MySQL 3.27 o superior como servidor de Bases de Datos
- ❖ Lenguaje de programación PHP
- ❖ En el lado del cliente para la utilización del sistema solo se limitan a tener disponible un navegador Web.
- Requerimiento de Hardware:
 - ❖ Procesador Pentium II 450 MHz o superior
 - ❖ 128 Mb de memoria RAM (incluye la utilizada por el SO) y 2Gb de capacidad en disco duro
 - ❖ Para el cliente debe tener el display con resolución 800 x 600, procesador Pentium o superior con 32 Mb de RAM como mínimo y Linux como sistema operativo
- Apariencia o Interfaz externa: El software brinda una interfaz de fácil uso y amigable para que el usuario pueda navegar dentro del sitio con gran facilidad y no tenga dificultad al interactuar con el sistema, ajustándose a los estándares establecidos para el desarrollo de un buen diseño donde la aplicación sea lo más interactiva posible.
- Portabilidad: El sistema correrá no solo sobre Linux sino también sobre Window, permitiendo que el servidor pueda ser cambiado sin importar sistema operativo llevando a cabo esta acción sin necesidad de efectuar cambios significativos.
- Requerimiento de Soporte: Se requiere de cualquier servidor con Linux y Apache instalado, además de un Servidor de Base de Datos MySQL. Puede ser un mismo servidor o servidores independientes.
- 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 la rapidez de respuesta del sistema ante las solicitudes de los usuarios. Como la aplicación está concebida para un ambiente cliente/servidor debe garantizarse que los tiempos de respuestas sean generalmente rápidos al igual que la velocidad de procesamiento de la información. Se sigue una lógica de negocios en la comunicación entre el cliente y el servidor Web, que cada cual procesa lo que le corresponda.
- Seguridad: El sistema contendrá información específica para diferentes usuarios, por lo que es muy importante la integridad de la información que contenga el sistema, para ello se autenticará cada usuario que acceda al sitio, y a partir de ahí se otorgarán los privilegios.

2.7 Descripción del modelo propuesto

Para cumplimentar los objetivos propuestos al inicio de este trabajo, y teniendo en cuenta todos los requerimientos planteados, el sistema que se propone debe autenticar a los usuarios, y a partir de su identificación (rol que posea) es la información que se le muestra en la página de inicio, donde puede interactuar con ella en correspondencia de los privilegios otorgados.

- Para utilizar todas las funcionalidades del sistema, se considera la existencia de 6 roles.
- El administrador del sistema, es quien gestiona toda la información, y obtiene algunas estadísticas específicas, gestiona los profesores que tendrán el rol de administrador o profesor privilegiado, y aprobará las solicitudes de este profesor de publicar encuestas, auto-evaluaciones para los estudiantes o sitios de navegación básica.
- El profesor privilegiado que podrá interactuar creando y sugiriendo lo que le administrador debe aprobar, obteniendo resultados e informaciones específicas.
- El estudiante que es quien puede auto-evaluarse en un contenido determinado.

- El profesor que es quien junto al estudiante conforman otro rol: usuario, que puede solicitar encuestas, acceder a sitios de navegación básica y enviar sugerencias al administrador.
- El usuario general que está formado por todos los roles antes mencionados, y es quien se logea para autenticarse y así obtener sus privilegios en dependencia del rol que posee, también manipula todos los documentos docentes (acceso al P1, donde van estar todas las clases de la asignatura distribuidas por semanas y temas: conferencias, clases prácticas, laboratorios; acceso a los contenidos extras, donde hay documentación sobre diferentes cursos, ejercicios de preparación; acceso al programa de la asignatura y al sistema de evaluación); además tiene acceso a las noticias generales.

En resumen, con este sistema se va lograr una Web Docente que cumpla con las expectativas de usuarios, profesores y administradores, y a partir de ella se podrá tener un mejor funcionamiento de la actividad docente en la Universidad de las Ciencias Informáticas.

2.8 Definición de los casos de uso del sistema

Los casos de uso de un sistema son procesos que responden a las funcionalidades definidas en los requerimientos funcionales. Estos especifican una secuencia de acciones que el sistema puede llevar a cabo interactuando con sus actores.

1. Autenticar usuario
2. Gestionar usuarios con roles
3. Gestionar Noticias
4. Gestionar Documentos docentes
5. Manipular documentos docentes
6. Ver noticias generales
7. Acceder a informaciones específicas
8. Crear encuesta
9. Solicitar encuesta
10. Ver resultados de encuesta

11. Crear auto-evaluación
12. Solicitar auto-evaluación
13. Ver resultados de auto-evaluación
14. Acceder a sitios de navegación básica
15. Sugerir sitios de navegación básica
16. Aprobar solicitudes de profesores privilegiados
17. Enviar sugerencias al administrador del sitio
18. Obtener reportes

2.8.1 Definición de los actores del sistema

Los actores de un sistema son agentes externos: aquellas personas o sistemas que interactúan con él. Representan los usuarios del sistema y otras aplicaciones que interactúan con él. Estos suelen corresponderse con trabajadores o actores del negocio. Los actores definidos en este sistema son los siguientes:

Actor	Descripción
Administrador	Tiene todos los privilegios para operar con la base de datos del sistema. Puede insertar, modificar y eliminar noticias, informaciones y otros documentos, además de ser quien gestiona a determinados usuarios con sus roles. También obtiene determinados reportes y es quien aprueba la publicación o no de las encuestas, auto-evaluaciones y solicitudes realizadas por el profesor privilegiado.
Profesor Privilegiado	Es el profesor que fue insertado en la base de datos por el administrador con este rol, es quien puede acceder a las informaciones específicas, crear encuestas, auto-evaluaciones, ver los resultados de estas; además de sugerirle al administrador sitios Web a incluir en la navegación básica.
Estudiante	Es un estudiante común que puede solicitar realizar auto-

Capítulo 2 – Descripción de la solución propuesta

	evaluaciones.
Profesor	Es el profesor de la UCI que no tiene privilegios. Forma parte.

Aunque no son actores específicamente, sino generalizaciones de ellos, tenemos al usuario general, que es quien agrupa a todos los actores para realizar determinadas acciones comunes para ellos, y al usuario que agrupa al estudiante y al profesor con el mismo fin. Debido a la gran importancia de estas funcionalidades de ambos usuarios por sus agrupaciones, se les hace referencia en toda la investigación como si cumplieran este rol.

2.8.2 Diagrama de casos de usos del sistema

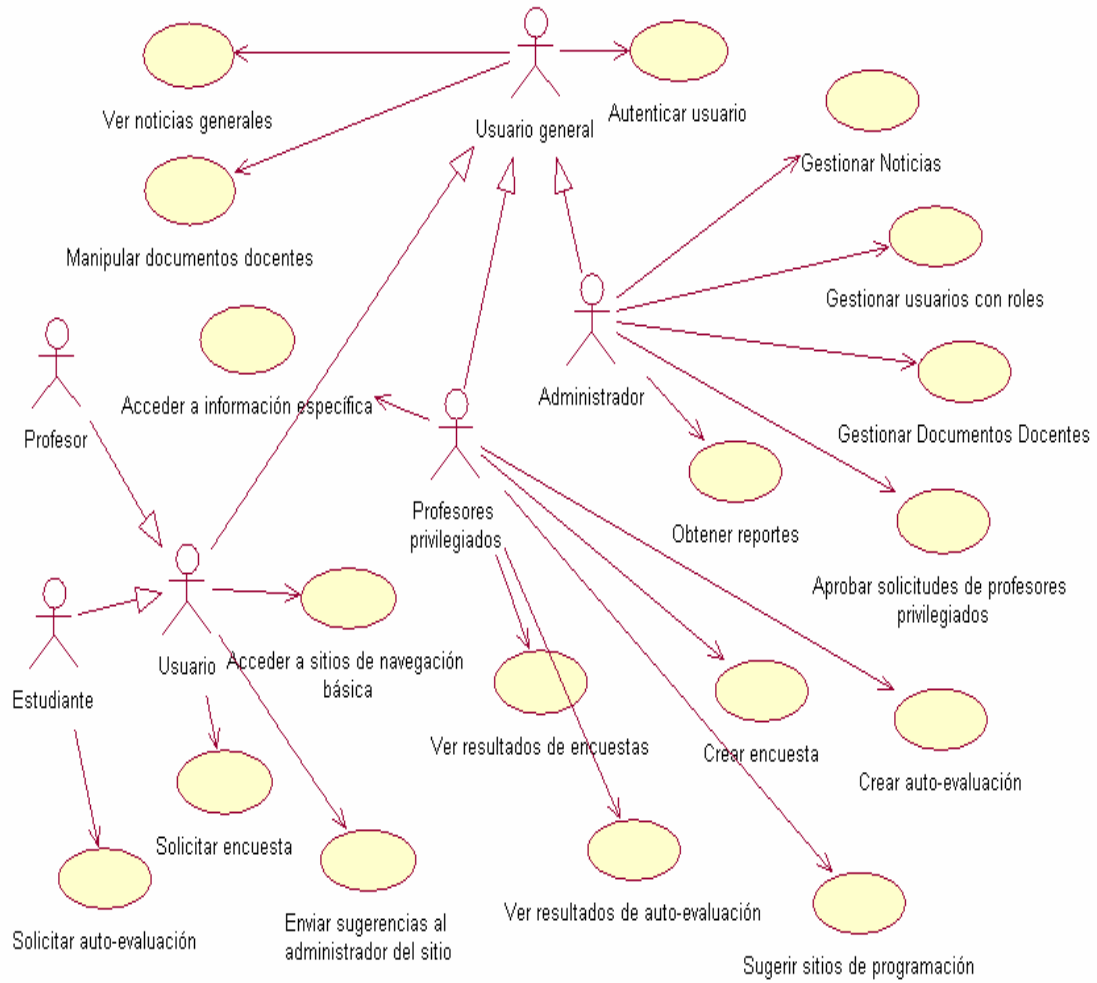


Fig.2.2 Diagrama de casos de uso del sistema

2.8.3 Descripción de los casos de uso del sistema

Caso de Uso 1:	Autenticar usuarios	
Actores:	Usuario General (inicia)	
Resumen:	El caso de uso comienza cuando un usuario decide acceder a la Web Docente de la asignatura, para esto el sistema verifica que tipo de usuario es tomando el nombre de usuario de entrada a la sección de la PC, y comparándolo con los datos existentes y en dependencia del usuario se le otorgarán los privilegios.	
Precondiciones:	El usuario introdujo sus datos al iniciar la sección	
Referencias	R1	
Prioridad	Alta	
Flujo Normal de Eventos		
	Acción del Actor	Respuesta del Sistema
	1) El usuario desea acceder a la WEB Docente de la asignatura introduciendo su nombre de usuario y contraseña	2) El sistema verifica el nombre de usuario con el usuario autenticado en la PC para conocer su rol y en dependencia de este mostrar las opciones que le corresponden. Si no coincide entonces 3) 3) El sistema verifica el usuario con la red de la UCI y accede al usuario como visitante del sistema.
Poscondiciones	Se autenticó el usuario que va a acceder al sistema con los privilegios otorgados según su rol.	

Tabla2.1 Caso de uso1: Autenticar Usuario

Caso de Uso 2:	Gestionar usuarios con roles	
Actores:	Administrador	
Resumen:	El administrador va a insertar los nombres de usuario con un rol determinado, pueden ser administradores o profesores privilegiados. También podrá modificar esta información o eliminarla.	
Precondiciones:	El usuario tiene que ser administrador.	
Referencias	R2	
Prioridad	Alto	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	
<p>1) El administrador selecciona gestionar usuarios con roles</p> <p>3) El administrador selecciona la acción</p>	<p>2) El sistema le muestra la página seleccionada con el formulario para inserte, elimine o modifique información referente al pedido.</p> <p>4) El sistema le muestra en relación con la selección una interfaz para que el seleccione la acción que va a realizar: insertar usuario (ver sección1), modificar usuario (ver sección 2), eliminar usuario (ver sección 3).</p> <p>5) Al terminar la acción el sistema muestra un mensaje de confirmación.</p>	
Sección1 “Insertar usuario”		
Acción del Actor	Respuesta del Sistema	
1) El administrador introduce el nombre	2) El sistema almacena la	

del usuario y la contraseña.	información
Sección 2 “Modificar Usuario”	
Acción del actor	Respuesta del Sistema
1) El administrador selecciona el usuario que desea modificar 3) El administrador modifica los datos que pueden ser el nombre de usuario, la contraseña y el rol	2) El sistema muestra una interfaz con la información existente sobre el usuario seleccionado 4) El sistema almacena la información.
Sección3 “Eliminar Usuario”	
Acción del actor	Respuesta del Sistema
1) El administrador selecciona el usuario que va a eliminar 3) El administrador elimina la información	2) El sistema muestra una interfaz con la información existente sobre el usuario. 5) El sistema elimina y se actualiza
Poscondiciones	

Tabla2.2 Caso de uso 2: Gestionar usuarios con roles

Caso de Uso 3:	Gestionar Noticias	
Actores:	Administrador	
Resumen:	El administrador del sistema podrá gestionar las noticias, seleccionará este vínculo, escogerá el tipo de noticia que va a gestionar y entonces ahí podrá insertar, eliminar o modificar, seleccionando la acción correspondiente.	
Precondiciones:	Si la acción es de eliminar o modificar tiene que existir la noticia	
Referencias	R3	
Prioridad	Alta	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	
1) El administrador selecciona el vínculo gestionar noticias. 3) El administrador selecciona el tipo de noticias que va a gestionar	2) Le muestra una interfaz para que el administrador seleccione que tipo de noticia va a gestionar (noticias generales o informaciones específicas). 4) El sistema le muestra en relación con la selección una interfaz para que el seleccione la acción que va a realizar: insertar noticias (ver sección1), modificar noticias (ver sección 2), eliminar noticias (ver sección 3).	
Sección1 “Insertar noticias”		
Acción del Actor	Respuesta del Sistema	
1) El administrador seleccionar el vínculo insertar noticias.	2) El sistema le brinda una interfaz con un formulario para que el administrador introduzca la	

	<p>información necesaria para que quede bien insertada la noticia.</p> <p>3) El sistema almacena la información.</p>
Sección2 “Modificar Noticias”	
Acción del actor	Respuesta del Sistema
<p>1) El administrador decide seleccionar el vínculo modificar noticias</p> <p>3) El administrador selecciona la noticia que desea modificar</p>	<p>2) El sistema le brinda una interfaz con las noticias existentes para que seleccione la que va a modificar</p> <p>4) Una vez seleccionada, el sistema le muestra una interfaz con un formulario con la información existente sobre ella, para verificar que el administrador desea modificar, si es así, mediante el formulario realizará los cambios.</p> <p>5) Esta información será guardada en la base de datos para su actualización.</p>
Sección3 “Eliminar Noticias”	
Acción del actor	Respuesta del Sistema
<p>1) El administrador decide seleccionar el vínculo eliminar noticias.</p> <p>3) El administrador selecciona la noticia que desea modificareliminar.</p>	<p>2) El sistema le brinda una interfaz con las noticias existentes para que seleccione la que va a eliminar</p> <p>4) Una vez seleccionada, el sistema le muestra una interfaz con un formulario con la información existente sobre ella, para verificar que</p>

Capítulo 2 – Descripción de la solución propuesta

	<p>el administrador desea eliminarla, si es así, lo hará mediante el formulario.</p> <p>5) Esta información será guardada en la base de datos para su actualización.</p>
Poscondiciones	<p>Si la acción fue insertar, se insertará una nueva noticia, si la acción fue eliminar o modificar se actualiza la base de datos, y muestra las noticias según los cambios efectuados.</p>

Tabla 2.3 Caso de uso 3: Gestionar noticias

Caso de Uso 4:	Gestionar Documentos Docentes	
Actores:	Administrador	
Resumen:	El administrador del sistema podrá gestionar los documentos docentes (insertar, eliminar, modificar).	
Precondiciones:	Si la acción es de eliminar o modificar tiene que existir el documento.	
Referencias	R4	
Prioridad	Alta	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	
1)El administrador selecciona el vínculo gestionar documentos docentes. 3) El administador selecciona el tipo de documento que va a gestionar	2) Le muestra una interfaz para que el administrador seleccione: gestionar PI u otros materiales docentes. 3) Una vez seleccionado, el sistema le brinda una interfaz para que seleccione la acción que va a realizar: insertar (ver sección 1), modificar(ver sección 2), eliminar (ver sección 3).	
Sección1 “Insertar Documento Docente”		
Acción del Actor	Respuesta del Sistema	
1) El administrador selecciona el vínculo insertar documento docente	2) El sistema le brinda una interfaz con un formulario para que el administrador introduzca la información necesaria para que quede bien insertada el documento. 3) La información quedará guardada en la base de datos, para su	

Capítulo 2 – Descripción de la solución propuesta

	publicación.
Sección2 “Modificar Documentos Docentes”	
Acción del actor	Respuesta del Sistema
<p>1) El administrador decide acceder a modificar documentos docentes.</p> <p>3) El administrador realiza una selección.</p>	<p>2) El sistema le brinda una interfaz con toda la información para que decida cual es la que va a modificar.</p> <p>4) Una vez seleccionado, el sistema le brinda una interfaz con un formulario, con toda la información que él desea modificar para que confirme que desea realizar la acción, modificará mediante el formulario.</p> <p>5) El sistema guardará los cambios en la base de datos del sistema.</p>
Sección3 “Eliminar Documentos Docentes”	
Acción del actor	Respuesta del Sistema
<p>1) El administrador decide acceder a eliminar documentos docentes.</p> <p>3) El administrador realiza una selección.</p>	<p>2) El sistema le brinda una interfaz con toda la información para que decida cual es la que va a eliminar.</p> <p>4) Una vez seleccionado, el sistema le brinda una interfaz con un formulario, con toda la información que él desea eliminar para que confirme que desea realizar la acción, eliminará mediante el formulario.</p> <p>5) El sistema guardará los cambios en la base de datos del sistema.</p>
Poscondiciones	Si la acción fue insertar, se insertará un nuevo documento

	docente, si la acción fue eliminar o modificar se actualiza la base de datos, y muestra los documentos docentes según los cambios efectuados.
--	---

Tabla 2.4 Caso de uso 4: Gestionar Documentos Docentes

Capítulo 2 – Descripción de la solución propuesta

Caso de Uso 5:	Manipular documentos docentes	
Actores:	Usuario General	
Resumen:	El usuario general accede al sitio para buscar información docente, ya sea el P1 con todos los materiales docentes, los materiales que se encuentran en los contenidos extras, el programa de la asignatura o el sistema de evaluación de la misma. Nuestro sistema le brinda esta información y lleva una estadística de todos los usuarios que acceden a cada uno de estos materiales.	
Precondiciones:	Tienen que existir los materiales docentes que el usuario va a utilizar.	
Referencias	R4, R5	
Prioridad	Alta	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	
1) El usuario decide utilizar un material docente: P1 (conferencias, clases prácticas, laboratorios); contenidos extras (cursos de programación y materiales de estudio); programa de la asignatura o sistema de evaluación de la asignatura.	<p>2) El sistema le muestra al usuario la sección que él solicitó, y en esta el contenido específico al que él quiere acceder, en caso que el material no se encuentre, ver CA1</p> <p>3) El sistema guarda el nombre de usuario de cada usuario general y al documento que este accedió.</p>	
Flujos Alternos		

Acción del Actor	Respuesta del Sistema
CA1: El usuario desea ver un material de estudio.	El sistema no encuentra el material y le muestra un mensaje al usuario.

Tabla 2.5 Caso de uso 5: Manipular Documentos Docentes

Caso de Uso 6:	Ver Noticias Generales	
Actores:	Usuario General	
Resumen:	El usuario decide ver las noticias y accede a ellas. El sistema le muestra las noticias.	
Referencias	R3,R5	
Prioridad	Normal	
	Acción del Actor	Respuesta del Sistema
	1)El usuario selecciona el vínculo de noticias.	2) El sistema le muestra la interfaz de noticias al usuario general para que navegue.

Tabla 2.6 Caso de Uso 6: Ver Noticias Generales

Caso de Uso 7:	Acceder a informaciones específicas	
Actores:	Profesor privilegiado	
Resumen:	El profesor privilegiado desea ver las informaciones específicas, el sistema se las muestra y guarda su nombre de usuario.	
Referencias	R3, R6	
Prioridad	Normal	
	Acción del Actor	Respuesta del Sistema
	1) El profesor privilegiado decide ver las informaciones específicas, y selecciona su vínculo.	2) El sistema le muestra la interfaz de estas informaciones 3) El sistema guarda el nombre de usuario y la fecha de cada profesor que acceda a esta sección.

Tabla 2.7 Caso de uso 7: Acceder a informaciones específicas

Caso de Uso 8:	Crear Encuesta	
Actores:	Profesor privilegiado	
Resumen:	El profesor privilegiado crea una encuesta sobre un tema determinado, tiene una interfaz donde sube la encuesta (localización) actualizándose así la base de datos del sistema. De cada profesor que cree una encuesta se guarda su nombre de usuario y nombre de la encuesta.	
Referencias	R7	
Prioridad	Normal	
	Acción del Actor	Respuesta del Sistema
	<p>1) El profesor privilegiado crea una encuesta y decide enviarla al administrador para que sea publicada.</p> <p>3) El profesor introduce la información solicitada.</p>	<p>2) El sistema le muestra la interfaz con un formulario para que introduzca la información necesaria</p> <p>4) La base de datos del sistema se actualizará, y enviará un mensaje de confirmación.</p> <p>3) El sistema guarda el nombre de usuario del profesor y la fecha del envío.</p>
Poscondiciones	Quedará agregada una encuesta en la base de datos.	

Tabla 2.8 Caso de uso 8: Crear Encuesta

Caso de Uso 9:	Solicitar encuesta	
Actores:	Usuario	
Resumen:	El usuario solicita realizar una encuesta, las encuestas están divididas por temas, el sistema le muestra una serie de estas, él escoge una y la realiza.	
Precondiciones:	Tiene que estar creada y aprobada por el administrador la encuesta para que sea publicada.	
Referencias	R8, R9	
Prioridad	Normal	
	Acción del Actor	Respuesta del Sistema
	<p>1)El usuario selecciona el vínculo para realizar una encuesta.</p> <p>3)El usuario selecciona el tema de la encuesta que desea hacer</p> <p>5)El usuario responde las preguntas de la encuesta</p>	<p>2) El sistema le muestra una interfaz con varios temas de encuesta para que el usuario seleccione una.</p> <p>4) El sistema le muestra las preguntas que conforman la encuesta.</p> <p>6) El sistema le informa al usuario cuando ha finalizado la encuesta con un mensaje, sino ver CA1.</p>
	Flujos Alternos	
	Acción del Actor	Respuesta del Sistema
	CA1: El usuario no responde más a las preguntas, y deja la encuesta sin terminar.	El sistema no guarda las respuestas contestada de la encuesta.
Poscondiciones	Se resolvió una encuesta.	

Tabla 2.9 Caso de uso 9: Solicitar Encuesta

Caso de Uso 10:	Ver resultados de encuestas	
Actores:	Profesor Privilegiado	
Resumen:	Un profesor privilegiado decide ver los resultados de una encuesta determinada y el sistema se los muestra.	
Precondiciones:	Tienen que estar resueltas al menos 2 encuestas del mismo tipo.	
Referencias	R10, R9	
Prioridad	Normal	
Acción del Actor		
Respuesta del Sistema		
1) Un profesor privilegiado desea ver los resultados de un determinado tipo de encuesta y selecciona este vínculo	2) El sistema le muestra una sección para que seleccione el resultado de un tipo de encuesta determinado.	
3) El profesor selecciona el tipo de encuesta del cual quiere ver los resultados.	3) El sistema le muestra los resultados al profesor de la encuesta seleccionada en dependencia de los parámetros seleccionados para su búsqueda, sino o hay ninguna encuesta resuelta: CA1	
Flujos Alternos		
Acción del Actor		
Respuesta del Sistema		
CA1: El profesor quiere saber los resultados de una encuesta determinada.	1) E sistema no encuentra resultado y le muestra un mensaje a profesor.	
Poscondiciones	El profesor privilegiado se lleva algún criterio acerca del tema de la encuesta.	

Tabla 2.10 Caso de uso 10: Ver resultados de Encuesta

Caso de Uso 11:	Crear Auto-evaluación	
Actores:	Profesor privilegiado	
Resumen:	El profesor privilegiado crea una auto-evaluación sobre un tema determinado, tiene una interfaz donde sube la auto-evaluación(localización) actualizándose así la base de datos del sistema. De cada profesor que cree una auto-evaluación se guarda su nombre de usuario y nombre de la misma. Solamente podrán auto-evaluarse estudiantes.	
Precondiciones:	Tiene que ser un estudiante para que se le muestre esta sección.	
Referencias	R11	
Prioridad	Normal	
	Acción del Actor	Respuesta del Sistema
	<p>1) El profesor privilegiado crea una auto-evaluación y selecciona enviarla para que sea publicada.</p> <p>3) El profesor introduce toda la información solicitada.</p>	<p>2) El sistema le muestra la interfaz con un formulario que él tiene que completar.</p> <p>4) La base de datos del sistema se actualizará, y enviará un mensaje de confirmación.</p> <p>3) El sistema guarda el nombre de usuario del profesor y la fecha del envío.</p>
Poscondiciones	Quedará agregada una auto-evaluación en la base de datos.	

Tabla 2.11 Caso de uso 11: Crear auto-evaluación

Caso de Uso 12:	Solicitar auto-evaluación	
Actores:	Estudiante	
Resumen:	Un usuario decide solicitar realizar una auto-evaluación, la selecciona, la realiza y luego obtiene un resultado.	
Precondiciones:	Tiene que estar creada y aprobada una auto-evaluación	
Referencias	R12, R11	
Prioridad	Normal	
	Acción del Actor	Respuesta del Sistema
	<p>1) El estudiante decide realizar una auto-evaluación y selecciona este vínculo</p> <p>3) El usuario selecciona una auto-evaluación</p> <p>5) El usuario responde todas las preguntas</p>	<p>2) El sistema le muestra las diferentes auto-evaluaciones que hay por temas para que seleccione una.</p> <p>4) El sistema le muestra las preguntas a responder.</p> <p>6) El sistema después de respondidas todas las preguntas, le muestra el resultado al usuario, en caso que quede alguna pregunta sin responder, ver CA1 o el usuario quiera salir sin terminar, ver CA2.</p> <p>4) Si el usuario obtuvo calificación, el sistema se conecta con la base de datos general de la UCI, y guarda los datos personales del usuario que entró a esta sección; toma el resultado de la auto-evaluación y los guarda en la base de datos, así como el nombre de usuario del estudiante y fecha de realización.</p>

Flujos Alternos	
Acción del Actor	Respuesta del Sistema
CA1: El estudiante no responde una pregunta e intenta pasar a la siguiente CA2: El estudiante deja la auto-evaluación sin terminar.	CA1: El sistema le muestra un mensaje: Tiene que responder la pregunta para poder continuar con la auto-evaluación. CA2: El sistema no guarda nada en la Base de Datos.
Poscondiciones	Un usuario ha realizado una auto-evaluación, se guarda su nombre de usuario, fecha y calificación obtenida.

Tabla 2.12 Caso de uso 12: Solicitar auto-evaluación

Caso de Uso 13:	Ver resultados de auto-evaluación	
Actores:	Profesor Privilegiado	
Resumen:	El profesor privilegiado desea ver los resultados de las auto-evaluaciones realizadas por un criterio determinado y el sistema se las muestra.	
Precondiciones:	Tiene que estar realizada al menos una auto-evaluación para que se pueda mostrar algún resultado.	
Referencias	R13, R12	
Prioridad	Normal	
	Acción del Actor	Respuesta del Sistema
	<p>1) El profesor desea ver resultados de las auto-evaluaciones.</p> <p>3) El profesor selecciona la facultad y el grupo deseado, para ver si hay resultados del tipo de auto-evaluación determinada.</p>	<p>2) El sistema le muestra una sección para seleccionar un criterio de búsqueda por facultad, grupo y tipo de evaluación para las auto-evaluaciones de estudiantes.</p> <p>4) El sistema consulta, y le muestra al profesor: en caso de ningún estudiante del grupo seleccionado tener realizado alguna auto-evaluación, un mensaje; sino le muestra el nombre y el resultado obtenido.</p>
Poscondiciones	El profesor posee información acerca del estado de sus estudiantes y de su seguimiento docente.	

Tabla 2.13 Caso de uso 13: Ver resultados de auto-evaluaciones

Caso de Uso 14:	Acceder a sitios de navegación básica	
Actores:	Usuario	
Resumen:	El usuario desea informarse acerca de sitios de programación internacionales y accede a este enlace; el sistema por temas, le muestra una serie de sitios con una descripción de ellos, quien los sugirió, y la dirección para acceder.	
Precondiciones:	Deben estar al menos un sitio colocado en la Web Docente.	
Referencias	R17(17.3), R14	
Prioridad	Normal	
	Acción del Actor	Respuesta del Sistema
	1) El usuario desea conocer de otros sitios de programación internacionales, y accede a este enlace.	2) El sistema le brinda una sección con varios sitios clasificados por su contenido en distintos temas.
	3) El usuario selecciona el sitio que le interese conocer.	4) El sistema le muestra esta sección específica, donde se encuentra la descripción, URL y profesor que recomienda el sitio.

Tabla 2.14 Caso de uso 14: Acceder a sitios de navegación básica

Capítulo 2 – Descripción de la solución propuesta

Caso de Uso 15:	Sugerir sitios de navegación básica	
Actores:	Profesor privilegiado	
Resumen:	El profesor privilegiado eleva una solicitud al administrador de colocar en la Web Docente la dirección de un sitio internacional de programación que él cree de utilidad para los usuarios por su temática en general.	
Referencias	R15	
Prioridad	Normal	
Acción del Actor		
Respuesta del Sistema		
1) El profesor privilegiado decide enviarle la dirección de un sitio al administrador para que lo coloque en la Web Docente para el estudio de los usuarios, y selecciona este vínculo.	<p>1) El sistema le brinda una interfaz donde el le envía la dirección del sitio, y también coloca una breve descripción de él.</p> <p>2) El sistema actualiza su base de datos, y guarda el nombre de usuario del profesor que ha sugerido el sitio. Si el sitio ya se encuentra, ver CA1</p>	
Flujos Alternos		
Acción del Actor		
Respuesta del Sistema		
CA1: El profesor va a proponer un sitio de navegación básica	1) El sistema se percata que el sitio ya está en la base de datos, y le muestra un mensaje al usuario.	

Tabla 2.15 Caso de uso 15: Sugerir sitios de navegación básica

Capítulo 2 – Descripción de la solución propuesta

Caso de Uso 16:	Aprobar solicitudes de profesores privilegiados	
Actores:	Administrador	
Resumen:	El administrador del sistema es quien aprueba si los materiales de encuestas, auto-evaluaciones, y sitios internacionales que proponen los profesores privilegiados, deben o no ser publicados en la Web Docente, él analiza, y luego realiza el proceso de aprobación.	
Precondiciones:	Tienen que estar al menos una solicitud para publicación de alguno de los materiales.	
Referencias	R16	
Prioridad	Alta	
Acción del Actor		
Respuesta del Sistema		
1) El administrador accede a la interfaz de sugerencias de profesores privilegiados.	2) El sistema le muestra una interfaz para que seleccione entre encuestas (sección1), auto-evaluaciones (sección2) y sitios de navegación básica (sección3) para mostrarles la información.	
Flujo Normal de Eventos		
Sección1 “Encuestas”		
Acción del Actor		Respuesta del Sistema
1) El administrador decide acceder a encuestas 3) El administrador analiza toda la información, y si aprueba la encuesta la publica en la Web Docente, sino la elimina.	2) El sistema le muestra una interfaz donde le muestra el nombre de la encuesta, documento y los datos del profesor que la diseñó.	
Flujo Normal de Eventos		
Sección2 “Auto-evaluaciones”		

Acción del actor	Respuesta del Sistema
<p>1) El administrador decide acceder a auto-evaluaciones</p> <p>3) El administrador analiza toda la información, y si aprueba la auto-evaluación la publica en la Web Docente, sino la elimina.</p>	<p>2) El sistema le muestra una interfaz donde le muestra el tema al que pertenece la auto-evaluación, el documento de contenido, y los datos del profesor que la realizó.</p>
Flujo Normal de Eventos	
Sección3 “Sitios de navegación básica”	
Acción del actor	Respuesta del Sistema
<p>1) El administrador decide acceder a sitios de navegación básica.</p> <p>3) El administrador analiza toda la información, y si aprueba el sitio lo coloca en la Web Docente, sino lo elimina.</p>	<p>2) El sistema le muestra una interfaz donde le muestra el tema al que pertenece el sitio, una descripción de él, su dirección de acceso, y los datos del profesor que hace la sugerencia.</p>

Tabla 2.16 Caso de uso 16: Aprobar solicitudes de profesores privilegiados

Caso de Uso 17:	Enviar sugerencias al administrador del sitio	
Actores:	Usuario	
Resumen:	El usuario una vez interactuando con el sitio, puede enviarle al administrador sugerencias a través de la misma interfaz en una sección determinada.	
Referencias	R17	
Prioridad	Normal	
	Acción del Actor	Respuesta del Sistema
	<p>1) Selecciona la opción de sugerencias que se encuentra en la interfaz inicial.</p> <p>3) El Usuario escribe la sugerencia y la envía</p>	<p>2) El sistema le brinda la interfaz donde el usuario puede escribirle su sugerencia.</p> <p>3) El sistema almacena la sugerencia para después ser analizada por el administrador.</p>

Tabla 2.17 Caso de uso 17: Enviar sugerencias al administrador del sitio

Capítulo 2 – Descripción de la solución propuesta

Caso de Uso 18:	Obtener reportes	
Actores:	Administrador	
Resumen:	El administrador quiere obtener las estadísticas de los usuarios que han accedido a un determinado documento docente, y también los profesores privilegiados que han accedido a las informaciones específicas. El sistema le brinda una sección con estas informaciones.	
Precondiciones:	Al menos un usuario tiene que haber accedido a algún documento para que se lleve estadística.	
Referencias	R5, R18	
Prioridad	Normal	
	Acción del Actor	Respuesta del Sistema
	<p>1) El administrador desea conocer determinadas estadísticas relacionadas con la navegación del sitio, y selecciona este vínculo.</p> <p>3) El administrador realiza la selección.</p>	<p>2) El sistema le muestra una sección para que seleccione la estadística que desea solicitar: a documentos docentes, o a informaciones específicas.</p> <p>4) El sistema le muestra la interfaz de la sección seleccionada, si es la de información específica le mostrará los datos de los profesores que han accedido a esta sección. Si selecciona la otra opción el administrador el sistema le mostrará un formulario para que seleccione el documento del que quiere conocer los datos, la facultad, y el tipo de usuario.</p> <p>5) El sistema le muestra todos los datos, pero si no ha sido visitado por</p>

Capítulo 2 – Descripción de la solución propuesta

	ningún usuario que cumpla con la selección, ver CA1.
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
CA1: El administrador realiza la selección.	El sistema le muestra un mensaje de no se ha obtenido información por el término de selección buscado.
Poscondiciones	El administrador tiene conocimiento de cómo se está cumpliendo con la utilización de materiales docentes, y visitas a informaciones específicas para determinados profesores.

Tabla 2.18 Caso de uso 18: Obtener Reportes

2.9 Conclusiones

En este capítulo se analizó la propuesta de solución al obtenerse una serie de requerimientos que el sistema debe cumplir, los cuales se modelaron a través del diagrama de casos de uso; también se identificaron los actores y se hizo una descripción ampliada de los casos de uso donde se describen todas las acciones que realizan los actores en general. A partir de aquí se puede comenzar a construir el sistema que constituye la propuesta de solución, cumpliendo con todos los requerimientos funcionales y no funcionales que se plantean en este capítulo para que la aplicación funcione adecuadamente.

Capítulo 3

Construcción de la solución propuesta

Capítulo III Construcción de la solución propuesta

3.1 Introducción

En este capítulo se describen los elementos más importantes correspondientes a la etapa de análisis y diseño del sistema, modelándose los artefactos que contribuyen al manejo de las complicaciones que pueden implicar la construcción de aplicaciones Web, se describe la puesta en práctica de la construcción de la solución propuesta ante su descripción en el capítulo anterior. Se diseña la propuesta de solución mediante el modelado de los diagramas de clases de diseño utilizando UML. Se trata la concepción general del tratamiento de errores, se muestra el diagrama de despliegue y el de componentes. Además se plantea el diseño lógico y físico de la Base de Datos.

3.2 Modelo de Análisis

Se utiliza para obtener una visión general del sistema sobre los requisitos funcionales, expresados en un lenguaje técnico. Su objetivo es analizar los requisitos, refinarlos y estructurarlos en un modelo de objetos que sirva como primera impresión del modelo de diseño, siendo mas abstracto y simple que el que este ultimo.

3.2.1 Clases del modelo de análisis

Las clases de análisis se centran en los requisitos funcionales y son evidentes en el dominio del problema porque representan conceptos y relaciones del dominio. Tienen atributos y entre ellas se establecen relaciones. RUP clasifica a las clases en:

- Entidad: Modelan información que posee larga vida y que es a menudo persistente.
- Interfaz: Modelan la interacción entre el sistema y sus actores.
- Control: Coordinan la realización de uno o unos pocos casos de uso coordinando las actividades de los objetos que implementan la funcionalidad del caso de uso. **Ver Anexo1**

3.3 Diagramas de Interacción

Los diagramas de Interacción muestran las interacciones entre objetos ocurridas en un escenario (parte) del sistema. Se dividen en dos tipos de diagramas de UML: de colaboración y de secuencia, los cuales expresan información similar pero de forma diferente.

3.3.1 Diagramas de Colaboración

Muestran la interacción entre varios objetos y los enlaces que existen entre ellos. Formando parte de estos diagramas nos encontramos con objetos, enlaces y mensajes y durante la ejecución de un diagrama de colaboración se crean y destruyen objetos y enlaces. A diferencia de un diagrama de secuencia muestran las relaciones entre los objetos, no su secuencia en el tiempo. [12] **Ver Anexo 2**

3.3.2 Diagramas de Secuencia

Muestran las interacciones entre un conjunto de objetos, ordenadas según el tiempo en que tienen lugar, es decir muestra la secuencia en el tiempo en que se producen los mensajes. Representa una forma de indicar el período durante el que un objeto está desarrollando una acción directamente o a través de un procedimiento y permiten indicar cuál es el momento en el que se envía o se completa un mensaje mediante el tiempo de transición, que se especifica en el diagrama. [12] **Ver Anexo 2**

3.3 Diagramas de clases de diseño

Para comprender como se pueden modelar las aplicaciones Web hay que saber que su arquitectura básica incluye los navegadores, la red y los servidores. La forma tradicional de modelar clases no es aplicable a la hora de diseñar una aplicación Web, por lo que para resolver este problema los especialistas del Rational decidieron modelar los aspectos del lado del servidor como una clase y los aspectos del lado del cliente como otra clase. En este caso se elaboró un

diagrama de clases Web para cada caso de uso del sistema, y para algunas secciones de estos específicas, de forma tal que se facilite la comprensión de cómo se relacionan los distintos componentes en la realización de cada uno de ellos. **Ver Anexo 3**

3.4 Principios de diseño

La facilidad de uso es un aspecto clave al momento de evaluar el diseño de un sitio Web. La creatividad, simplicidad y elegancia del diseño y arquitectura informacional de una web contribuyen decisivamente a satisfacer las expectativas de sus visitantes [¡Error! No se encuentra el origen de la referencia.]. La interfaz es el medio por el cual el usuario interactúa con el sistema, por lo que debe ser lo más amigable y flexible posible para lograr que este se sienta identificado con la misma. Una aplicación con una interfaz bien diseñada debe tener, además de un buen diseño gráfico, una buena navegabilidad, usabilidad y distribución de los contenidos. Persiguiendo estos objetivos, para el diseño de la interfaz de usuario de este sistema se proponen para su implementación los siguientes principios:

- Mostrar al usuario solamente aquellas opciones a las que, dado su rol en el sistema, tenga derecho a acceder.
- Garantizar la legibilidad, que el color de los textos contraste con el del fondo, así como un tamaño de letra adecuada.
- Brindar al usuario la información bien distribuida, y accesible fácilmente para su manipulación.
- Rapidez: las páginas deben descargarse rápidamente.
- Eficiencia: satisfacer las expectativas de los navegantes de modo que permanezcan en el sitio y repitan sus visitas.

3.5 Tratamiento de errores

El tratamiento de errores posibilita el buen funcionamiento de una aplicación dándole una mejor apariencia ante los clientes. Según el diseño de esta aplicación el sistema le mostrará de una forma clara y precisa al usuario los

mensajes de error, los cuales serán tratados de dos formas, los del lado del cliente y los del lado del servidor. Los primeros consistirán en chequear situaciones excepcionales utilizando el lenguaje JavaScript y sus mensajes de alerta, se utilizarán sobre todo en la validación de entrada de información en la realización de encuestas y auto-evaluaciones. Las excepciones que ocurrirán internamente en la aplicación serán capturadas por el sistema y mostradas como mensajes de error al usuario (errores del lado servidor); estos mensajes aparecerán en otra ventana, serán específicos, concisos y entendibles para el usuario.

3.6 Diseño de la base de datos

3.6.1 Diagrama de clases persistentes

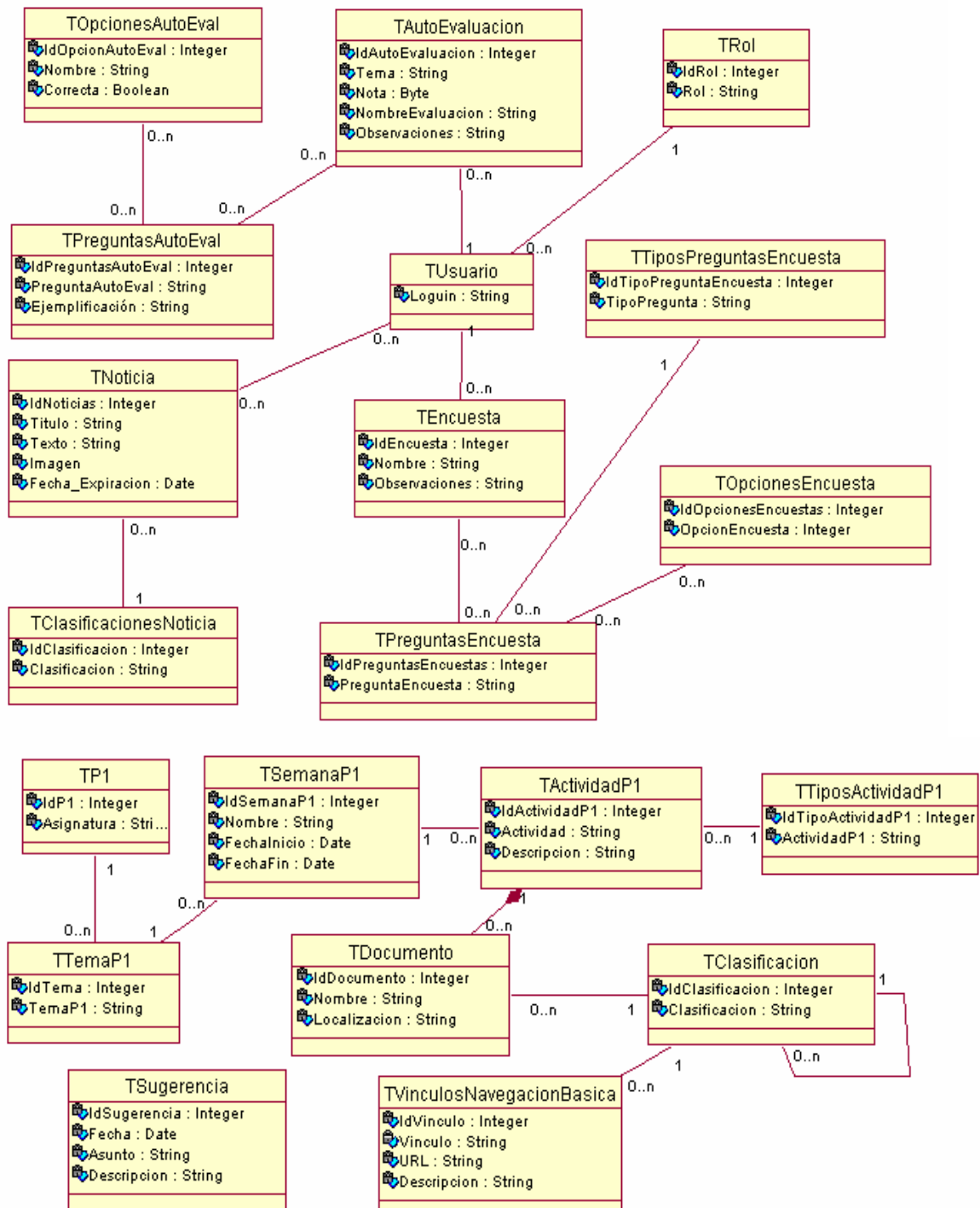


Fig.3.1 Diagrama de clases persistentes

3.6.2 Modelo de Datos. [Ver anexo 4](#)

3.7 Diagrama 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.

A partir del diseño de la aplicación, basado en una arquitectura Cliente-Servidor, donde un modelo cliente-servidor simple define dos capas: capa cliente (ambiente de trabajo del usuario por lo que tiene la interfaz de la aplicación) y capa servidora (contiene la base de datos) [**¡Error! No se encuentra el origen de la referencia.**], en este diagrama se representan cuatro nodos; uno de ellos es el PC Cliente, que representan las PC desde las cuales se podrán acceder utilizando el protocolo HTTP a la aplicación que se encuentra en el Servidor Web, donde están contenidas las capas de la arquitectura antes mencionada. Este Servidor Web es otro nodo que se comunica mediante el protocolo TCP/IP con el nodo Servidor de Base de Datos del Sistema, (que utiliza MYSQL como sistema gestor de Base de Datos) para realizar todas las actualizaciones y consultas al mismo. El Servidor Web donde se encuentra la aplicación accede además a otro nodo Servidor de Base de Datos, y es el de la UCI, donde se

accede para verificar a los usuarios, y obtener la información referente a ellos.

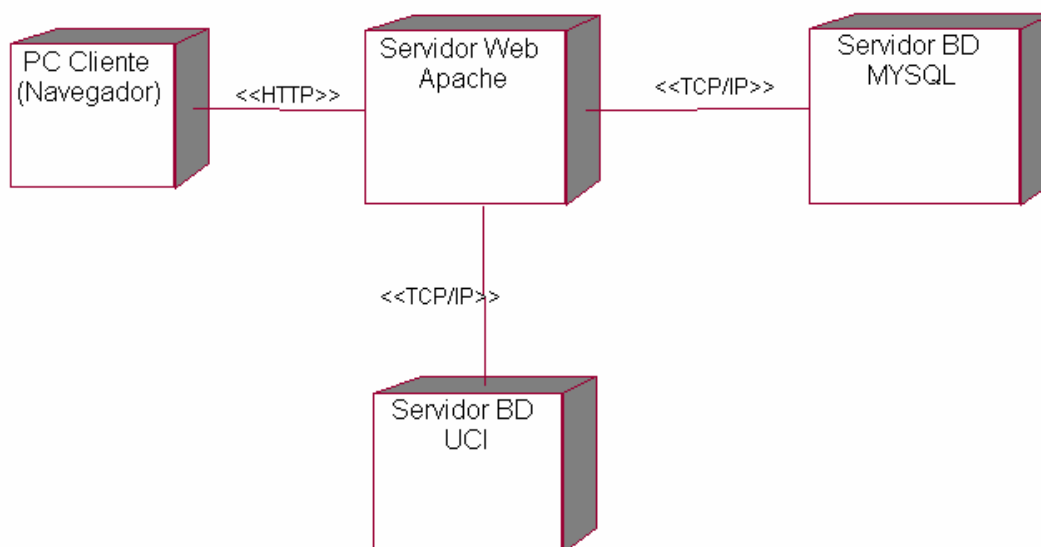


Fig.3.2 Diagrama de despliegue

3.8 Diagrama de Componentes

Los diagramas de componentes muestran la dependencia entre los distintos componentes de software, incluyendo componentes de código fuente, binario y ejecutable. Un componente es un fragmento de código software (un fuente, binario o ejecutable) que se utiliza para mostrar dependencias en tiempo de compilación. Para la realización de estos diagramas se dividieron por módulos las acciones:

Módulo General: Se muestran todos los módulos en los que hemos dividido los casos de uso.

Módulo Gestión-Profesores: Se reflejan todas las entradas del profesor privilegiado. (Todo lo que él puede crear)

Módulo Administración: Se muestra todo lo que el administrador del sistema tiene acceso para realizar.

Módulo Acceso-Informaciones: Es todo con lo que el usuario interactúa, cualquier usuario (de forma general).

Módulo Reportes: Se muestran los reportes que se obtienen.

Módulo General

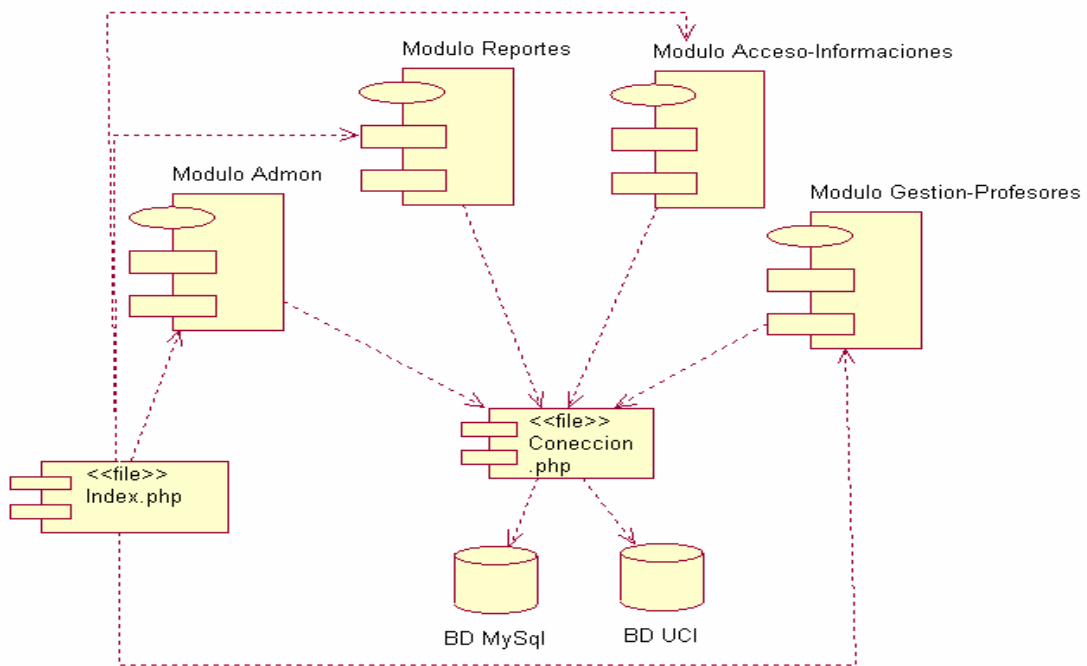


Fig.3.3 Módulo General

Módulo Gestión-Profesores

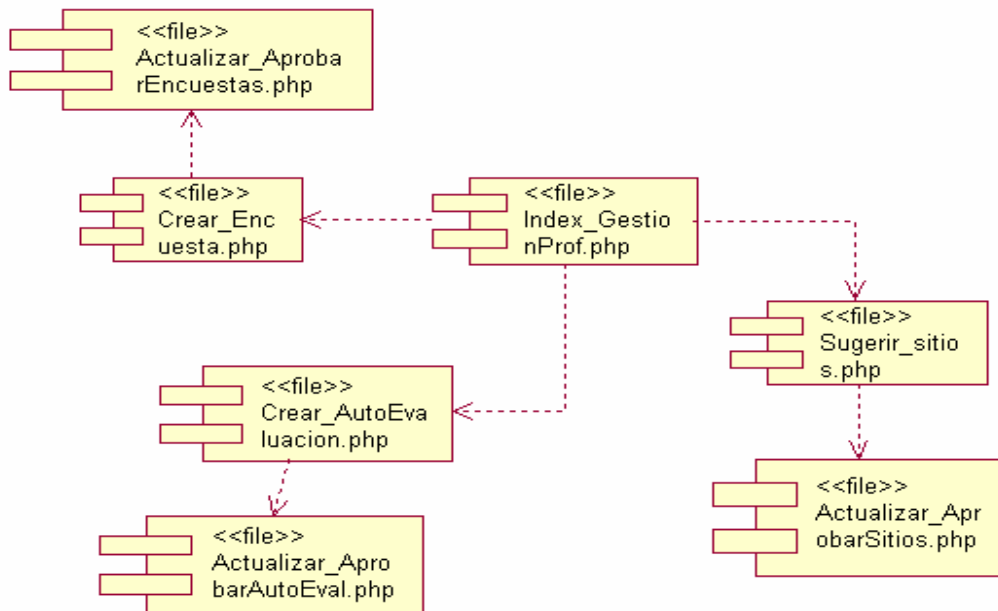


Fig.3.4 Módulo Gestión-Profesores

Módulo Administración

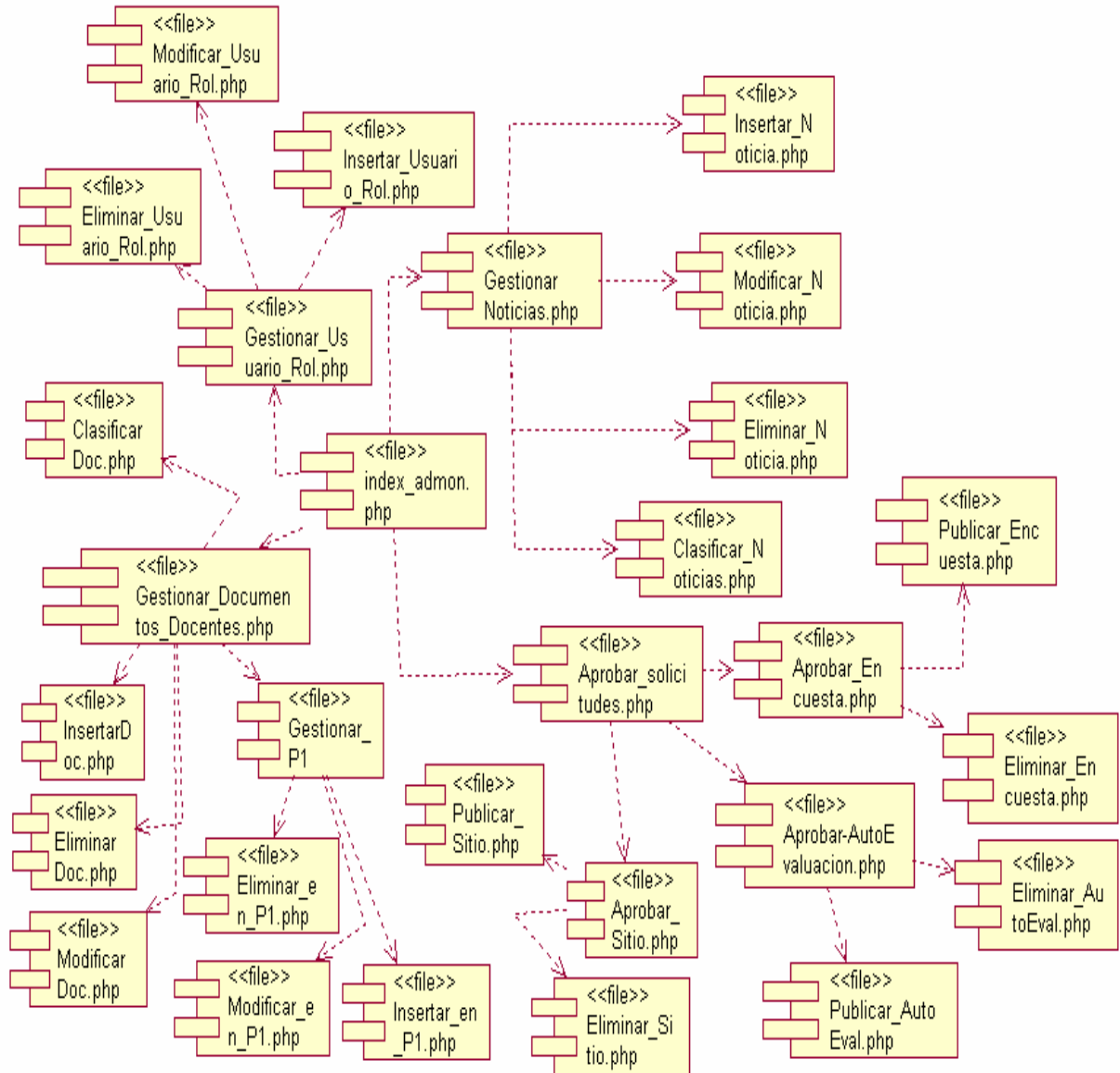


Fig.3.5 Módulo Administración

Módulo Acceso-Informaciones

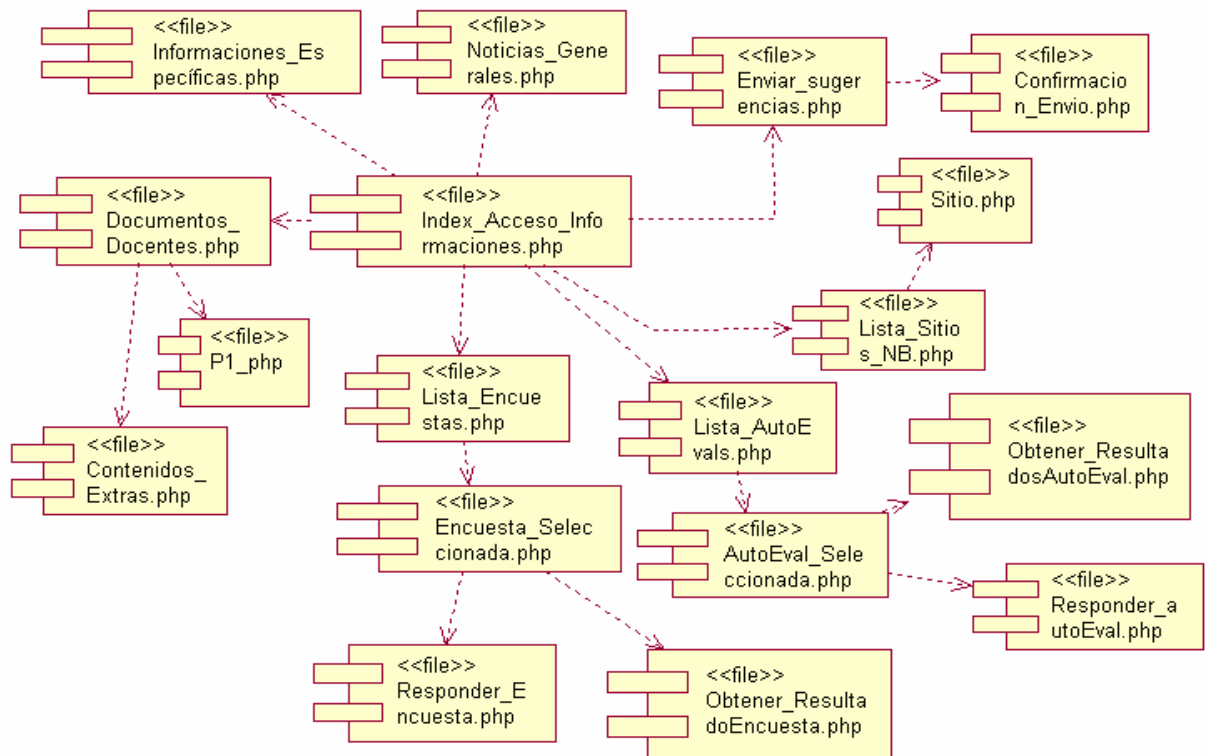


Fig.3.6 Módulo acceso-Informaciones

Módulo Reportes

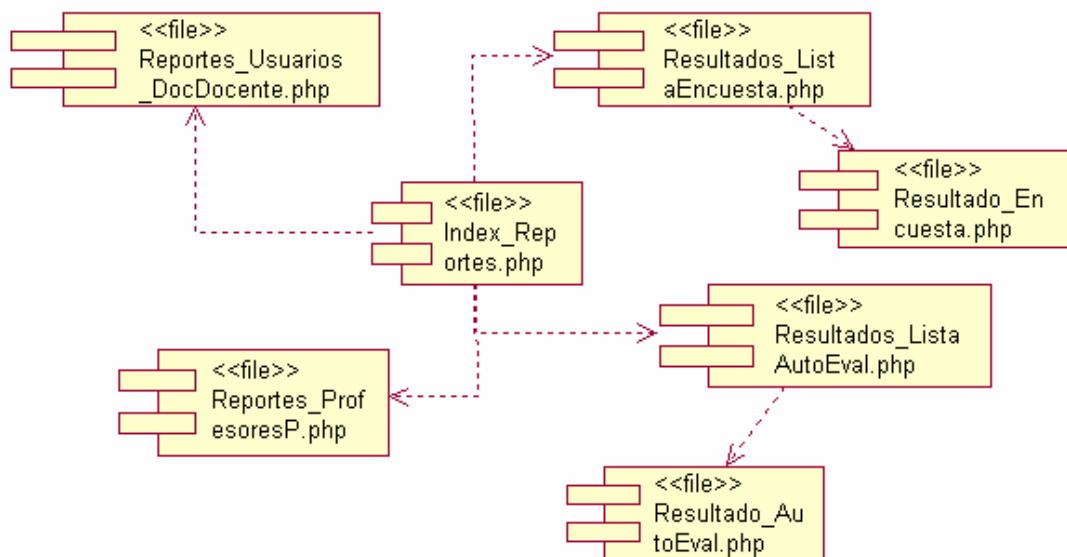


Fig.3.7 Módulo Reportes

3.9 Conclusiones

En este capítulo se modeló detalladamente la solución propuesta mediante el análisis y diseño de los requisitos del sistema. Se modelaron los diagramas de clase de análisis, de interacción, diagrama de las clases del diseño, el diagrama de despliegue y el de componentes. Se definieron los principios del diseño y se analizó el proceso de tratamiento de excepciones. Se diseñó la Base de Datos mediante la definición y modelación de las clases persistentes, a partir de las cuales se construyó el modelo de datos, lo que tiene gran importancia para el análisis de los objetos necesarios para el almacenamiento de información.

Capítulo 4

Estudio de factibilidad

Capítulo IV Estudio de factibilidad

4.1 Introducción

Para la realización de un proyecto es necesario estimar el esfuerzo humano, el tiempo de desarrollo que se requiere para la ejecución del mismo y también su costo. En este capítulo mediante el método de puntos de función del modelo de COCOMO se evalúa la factibilidad, beneficios y costo del sistema propuesto. Además se calcula el tiempo de desarrollo y el esfuerzo a partir de las entradas y salidas externas, los ficheros internos, peticiones, puntos de función desajustados, el lenguaje utilizado, los factores de escala y los multiplicadores de esfuerzo.

4.2 Estimación de Costos

Existen varios modelos para estimar los costos de los proyectos de software. Entre ellos uno de los más aceptados internacionalmente en la actualidad es el modelo COCOMO v2 (del inglés Constructive Cost Model). Este modelo expresa el esfuerzo de desarrollo en términos de Personas Mes, al final se realiza una estimación del costo monetario del proyecto basado en el tiempo estimado de desarrollo, la cantidad de personas involucradas y el salario de estas personas. En el caso de la aplicación Web Docente, al ser un sistema a desarrollar en la Universidad de Ciencias Informáticas, no se hace necesario calcular el costo monetario, pues se cuenta con el capital humano necesario (estudiantes y profesores), sin necesidad de contratar personal ajeno a la institución. Se realizó un estudio del esfuerzo necesario para el desarrollo del sistema, arrojando un resultado de 13 Personas Mes

4.2.1 PlanificaciónEntradas Externas

Nombre de la entrada externa	Cantidad de Ficheros	Cantidad de Elementos de datos	Clasificación (Bajo, Medio y Alto)
Introducir usuarios con roles	2	2	Bajo
Insertar noticias	2	6	Medio
Insertar documentos docentes	3	7	Medio
Eliminar noticias	2	6	Medio
Eliminar documentos docentes	3	7	Medio
Crear encuesta	4	5	Alto
Crear auto-evaluación	3	8	Medio
Sugerir sitios de navegación básica	2	4	Bajo
Enviar sugerencias al administrador del sitio	1	4	Bajo

Tabla 4.1 Entradas Externas al sistema

Salidas Externas

Nombre de la salida externa	Cantidad de Ficheros	Cantidad de Elementos de datos	Clasificación (Bajo, Media y Alto)
Listado de estudiantes por facultad que acceden a un documento	2	2	Bajo
Listado de estudiantes por grupo que acceden a un documento	2	2	Bajo
Listado de profesores privilegiados que acceden a informaciones específicas	3	3	Bajo
Listado de resultado de las encuestas	1	1	Bajo
Listado de resultados de las auto-evaluaciones	2	3	Bajo

Tabla 4.2 Salidas Externas

Peticiones

Nombre de la petición	Cantidad de Ficheros	Cantidad de Elementos de datos	Clasificación (Bajo, Media y Alto)
Buscar nombre de usuario de administrador y profesor privilegiado	1	2	Bajo
Buscar documentos docentes	2	3	Bajo
Buscar noticias	1	2	Bajo
Buscar Encuesta	1	1	Bajo
Buscar Auto-Evaluación	1	2	Bajo
Buscar sitios de navegación básica	1	2	Bajo

Tabla 4.3 Peticiones

Ficheros Internos

Nombre del fichero interno	Cantidad de records	Cantidad de Elementos de datos	Clasificación (Bajo, Media y Alto)
Opciones auto-evaluación	1	3	Bajo
Preguntas auto-evaluación	1	3	Bajo
Auto-evaluaciones	1	5	Bajo
Encuesta	1	3	Bajo
Pregunta encuesta	1	2	Bajo
Tipos preguntas encuestas	1	2	Bajo
Opciones encuestas	1	2	Bajo
Usuarios	1	2	Bajo
Roles	1	2	Bajo
Noticias	1	5	Bajo
Clasificaciones noticias	1	2	Bajo
Semanas P1_1	1	4	Bajo
Actividades P1_1	1	3	Bajo
Tipo actividades P1_1	1	2	Bajo
TemasP1_1	1	2	Bajo
Documentos_1	1	3	Bajo
P1_1	1	2	Bajo
Clasificaciones_1	1	2	Bajo
Vínculo navegación básica	1	4	Bajo
PreguntasAutoevaluación _Opciones	1	2	Bajo
Autoevaluaciones_Preguntas	1	2	Bajo

Encuestas_PreguntasEncuestas	1	2	Bajo
PreguntasEncuestas_OpcionesEncuestas	1	2	Bajo
Sugerencias	1	4	Bajo

Tabla 4.4 Ficheros Lógicos Internos

Elementos	Bajos		Medios		Altos		Subtotal de puntos de función
	No	X Peso	No	X Peso	No	X Peso	
Ficheros lógicos internos	24	7	0	10	0	15	168
Entradas externas	3	3	5	4	1	6	35
Salidas externas	5	4	0	5	0	7	20
Peticiones	6	3	0	4	0	6	18
Total							241

Tabla 4.5 Obtención de los puntos de función sin ajustar

4.2.1 Costos

Cantidad de Instrucciones Fuentes Estimadas (SLOC)

Características	Valor
Puntos de función desajustados	241
Lenguaje	PHP (80 %) MYSQL(20%)
Instrucciones fuentes por puntos de función	15 39
Instrucciones fuentes por lenguaje (miles de instrucciones)	(2,892) (1,879)
Instrucciones fuentes (miles de instrucciones)	4,771 (Representa las miles de instrucciones fuentes estimadas a realizar)

Tabla 4.6 Instrucciones fuentes estimadas

Factores de Escala

Factores	Valor	Justificación
PREC	3.72	Se han desarrollado proyectos parecidos en el pasado, pero el equipo no tiene mucha experiencia en este tipo de software.
FLEX	2.03	Se realizó un diseño para un buen cumplimiento de los requerimientos del sistema y para flexibilizarlo al máximo.
TEAM	2,19	El equipo que va desarrollar el software es cooperativo.
RESL	2.83	La herramienta de desarrollo seleccionada posee gran tolerancia ante fallos y capacidad de resolución de riesgos que pudieran ocurrir. Se estima que la reducción del riesgo es entorno a un 75%.
PMAT	6.24	El proceso tiene un bajo nivel de madurez.

Tabla 4.7 Factores de Escala

Multiplicadores de Esfuerzo

Nombre	Valor	Justificación
RUSE	1.00	Se implementa código reusable para el aprovechamiento de este en toda la aplicación
PDIF	1.00	La plataforma es estable
PREX	1.33	El equipo de trabajo no tiene experiencia previa con la plataforma y las herramientas de desarrollo.
FCIL	0.87	Se hace un uso notable de herramientas CASE para documentar el sistema.
SCED	1.00	Planificación normal
RCPX	1.00	Alta confianza requerida, base de datos mediana, documentación básica y complejidad normal
PERS	1.00	Nivel medio del personal, nominal

Tabla 4.8 Multiplicadores de Esfuerzo

La ecuación que plantea COCOMO para calcular el esfuerzo de desarrollo es la siguiente:

$$PM = A \times Size^E \times \prod_{i=1}^n EM_i$$

donde $E = B + 0.01 \times \sum_{j=1}^5 SF_j$

$$\prod FE_i = 17.01 \quad \sum ME_i = 1.157$$

$$E = 0,91 + 0,01 * 17,01 = 1,0801$$

$$PM = 2,94 * 4,771^{1,0801} * 1,157 = 18,39 \approx 18 \text{ hombres-mes.}$$

La ecuación que plantea COCOMO para calcular el tiempo de desarrollo es la siguiente:

$$TDEV = C \times PM^F$$

donde $F = D + 0.2 \times (E - B)$

$$F = 0,28 + 0,2 \times (1,0801 - 0,91) = 0,31$$

$$TDEV = 3,67 \times 18^{0,31} = 8,99 \approx 9 \text{ meses}$$

La ecuación que plantea COCOMO para calcular la cantidad de hombres es la siguiente:

$$CH = PM / TDEV$$

$$CH = 18/9 = 2 \text{ personas}$$

Para el desarrollo del software se utilizarán 4 roles desarrolladores: Un analista de sistema, dos programadores, y un ingeniero de prueba. Entonces como el sistema será desarrollado en su totalidad por 4 personas, indica un tiempo de desarrollo de 4 meses y medio, lo que lleva a:

$$TDEV = 5 \text{ meses (aproximadamente)}$$

$$CH = 4$$

Cálculo de:	Valor
Esfuerzo	18 hombres-mes
Tiempo de desarrollo	5 meses
Cantidad de hombres	4 hombres

Tabla 4.9 Resumen de los resultados del estudio de factibilidad del proyecto

4.3 Beneficios tangibles e intangibles

La Web Docente es un software de apoyo a la docencia que va a permitir a través de la web a todos los usuarios de la UCI encontrar y manejar eficazmente y de forma flexible variada información docente relacionada con una asignatura determinada, además de otras posibilidades. Este sistema pudiera ser utilizado en otros centros educacionales, ya que sería adaptable a nuevas características, aunque su fin es responder a necesidades actuales de la UCI, por tal motivo los beneficios obtenidos son totalmente intangibles y van dirigidos a:

- Lograr una mayor flexibilidad en la manipulación de gran variedad de información docente acerca de distintos temas de una asignatura determinada.
- Lograr un estándar de trabajo para todos los usuarios al ser tomada la WEB Docente como plantilla para todas las asignaturas.
- Lograr a profesores y administradores obtener estadísticas específicas y necesarias, para saber el nivel de funcionamiento, de utilización y efectividad del sitio.
- Lograr que los estudiantes puedan autoevaluarse y enviar sugerencias al administrador del sitio a través de la WEB.

4.4 Análisis de Costo y Beneficios

El desarrollo de un producto informático tiene asociado un costo y el llevarlo a cabo o no, está en dependencia de los beneficios que él mismo pudiese reportar. Los beneficios pueden ser económicos y de orden social, estos últimos son de tanta importancia como los primeros. La herramienta de software que se propone está dirigida a la educación, además su utilización propiciará un mayor protagonismo y flexibilidad de estudio para profesores y estudiantes en el proceso de Enseñanza-Aprendizaje, por tanto su mayor beneficio es de orden social.

Además de esto el sistema de Web Docente no requiere inversión alguna en software y herramientas de desarrollo, pues todas las herramientas que se han

empleado, y se utilizarán en su desarrollo son libres, además de estar totalmente orientado al usuario, con un fácil y sencillo manejo del sistema.

Por todo lo planteado se considera que es factible el desarrollo de la aplicación y que el esfuerzo de desarrollo (18 Personas Mes) como una estimación y que la utilización de cuatro roles para el proceso de desarrollo del software (real): dos programadores, un analista del sistema y un ingeniero de prueba durante cinco meses aproximadamente está plenamente justificado.

4.5 Conclusiones

En este capítulo se realizó el estudio de factibilidad correspondiente al sistema, realizándose el análisis entre los costos y los beneficios que reporta la aplicación, lo que logró determinar que es factible el desarrollo del software propuesto. Además se determinó que en un tiempo de desarrollo aproximado de cinco meses se debe desarrollar utilizando a cuatro desarrolladores.

Conclusiones

Conclusiones

Con el estudio realizado y la propuesta de diseño del sistema, se obtuvo la WEB Docente, sitio modelo para las distintas asignaturas en la Universidad de las Ciencias Informáticas, teniendo como punto de partida el sitio Web para la asignatura de IP. Esta WEB, una vez implementada, brindará la información de las asignaturas de una forma flexible, y de fácil manipulación, logrando un nivel de satisfacción adecuado a todos sus usuarios.

Se realizó un análisis de los sitios Web actuales en la UCI, y en otras universidades del mundo, llegando a las necesidades y problemáticas que fueron punto de partida de esta investigación.

Se realizó el análisis y diseño del sistema mediante diagramas de interacción, diagramas de clases de diseño Web, diagrama de despliegue, de componentes y de clases persistentes; se representó la base de datos física mediante el modelo de datos, todo en correspondencia con los requisitos que el sistema debe cumplir.

Con el diseño también se les permite obtener a profesores y administradores estadísticas específicas y necesarias para saber el nivel de funcionamiento, de utilización y efectividad del sitio.

Por todo lo planteado se llega a la conclusión que han sido cumplidos todos los objetivos específicos propuestos para darle cumplimiento al objetivo general de la investigación, y por tanto resolver el problema que constituyó el punto de partida de la misma.

Recomendaciones

Recomendaciones

Se recomienda:

- Implementar el sistema a partir del diseño propuesto en la investigación y utilizando la tecnología seleccionada.
- Extender este sistema a todas las asignaturas de la UCI, para que el usuario cree un estándar de trabajo y de esta forma facilitarle la interacción con el sitio.
- Continuar con el estudio del sistema con el objetivo de añadir nuevas funcionalidades.
- Llevar a cabo un riguroso control del estudio de factibilidad, en aras de contar con una base de análisis que permita predecir el costo de futuros proyectos.
- Enlazar el sistema propuesto una vez realizado su desarrollo con el resto de los sistemas académicos.

Referencias bibliográficas

Referencias Bibliográficas

1. Dr. Pére Marques Graells, Impacto de las Tic en la Educación
<http://dewey.uab.es/pmarques/siyedu.htm> 6/2/06
2. Horruitiner Silva, Pedro. "La Universidad cubana: El modelo de formación". Editorial Félix Varela, 2006. 21/3/06
3. Arquitectura cliente-servidor <http://www.csi.map.es/csi/silice/Global71.html>
17/4/06
4. .NET <http://es.wikipedia.org/wiki/.NET> 23/2/06
5. J2EE: Aplicaciones Avanzadas de Java para Entornos Profesionales
<http://sensei.lsi.uned.es:8000/cea-j2ee/guia.html> 23/2/06
6. Análisis corporativo de LAMP
<http://www.mexicoextremo.com.mx/content/view/187/2/> 23/2/06
7. LAMP http://www.ciberaula.com/curso/lamp/que_es/ 27/2/06
8. Ventajas de .NET
<http://www.desarrolloweb.com/articulos/1329.php?manual=48> 28/2/06
9. Servidores de aplicaciones J2EE y .NET
<http://www.qualitrain.com.mx/objecapsulas/obje15.htm> 27/2/06
10. Desarrollo Orientado a Objetos con UML
<http://www.clikear.com/manuales/uml/> 7/3/06
11. Lic. Jorge Luis Pérez Subirat, Diseño Informacional de los Sitios Web
http://www.bvs.sld.cu/revistas/aci/vol11_6_03/aci09603.htm 27/5/06
12. Ingeniería del software UML
<http://www.monografias.com/trabajos5/insof/insof.shtml> 5/6/06
13. Conferencia de modelo de implementación, Ingeniería de Software 3,
curso 2003-2004. 5/6/2006

Bibliografía

Bibliografía

- Jacobson, Ivar; Booch, Grady; Rumbaugh, James. "El Proceso Unificado de Desarrollo de Software". Addison Wesley, 2000 23/3/06
- Departamento de Arquitectura y tecnología de computadores, Universidad de granada <http://atc.ugr.es/> 15/3/06
- Dpto de sistemas informáticos uclm <http://www.info-ab.uclm.es/> 15/3/06
- Lenguajes y ciencias de la computación, Universidad de Málaga <http://www.lcc.uma.es/> 15/3/06
- Departamento de lenguajes y computación, Universidad de Almería <http://www.ual.es/Universidad/Depar/LengComp/leng-comp.html> 15/3/06
- Qué es UML? <http://es.tldp.org/Tutoriales/doc-modelado-sistemas-UML/multiple-html/c12.html> 4/4/06
- Programa de la asignatura: tecnología educativa <http://dewey.uab.es/pmarques/uabte999.htm> 17/4/06
- [Rational] "Lo nuevo de Rational Rose 2000". Rational Corporation, 2000. www.abists.com.mf/Fabs/Rational/notasTK 7/3/06
- Seminario del Rational Rose <http://kybele.escet.urjc.es/Documentos/ISI/Seminario%20Rose.pdf> 25/5/06

Glosario de términos

Glosario de Términos

COMUNICACIÓN = Transmisión de mensajes entre personas. Como seres sociales las personas, además de recibir información de los demás, necesitamos **comunicarnos** para saber más de ellos, expresar nuestros pensamientos

INFORMACIÓN = Datos que tienen significado para determinados colectivos. La **información** resulta fundamental para las personas, ya que a partir del **proceso** cognitivo de la información que obtenemos continuamente con nuestros sentidos vamos tomando las decisiones que dan lugar a todas nuestras acciones

TECNOLOGÍA = Aplicación de los conocimientos científicos para facilitar la realización de las actividades humanas. Supone la creación de productos, instrumentos, lenguajes y métodos al servicio de las personas.

TIC = Tecnología + Información + Comunicación

API = (del inglés **Application Programming Interface - Interfaz de Programación de Aplicaciones**) es un conjunto de especificaciones de comunicación entre componentes software.

SSL = Siglas de Secure Socket Layer. Es un protocolo desarrollado por Netscape Communications Corporation para dar seguridad a la transmisión de datos en transacciones comerciales en Internet.

Anexos

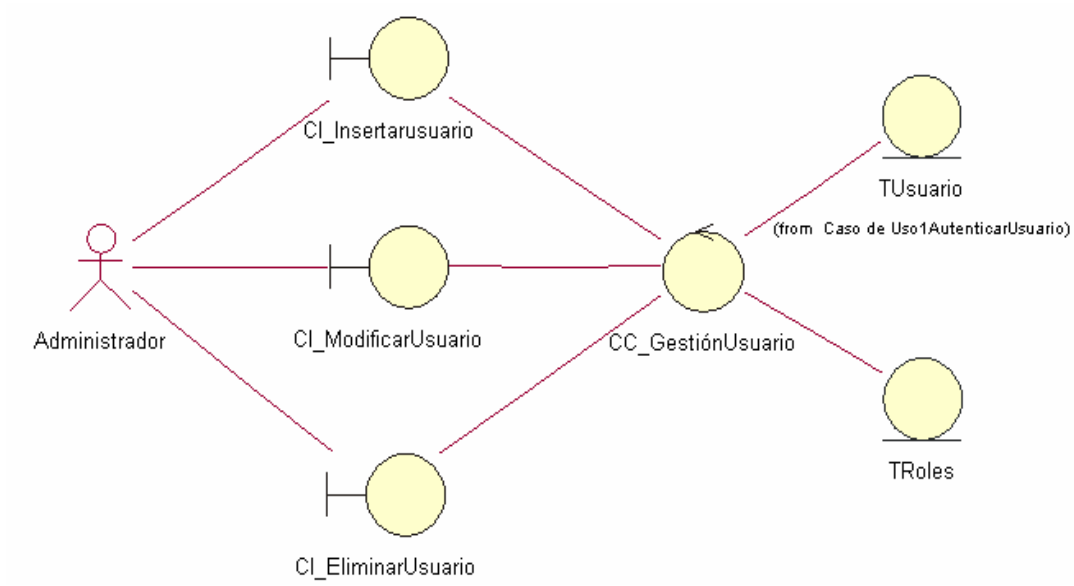
Anexos

Anexo 1. Clases del modelo de análisis

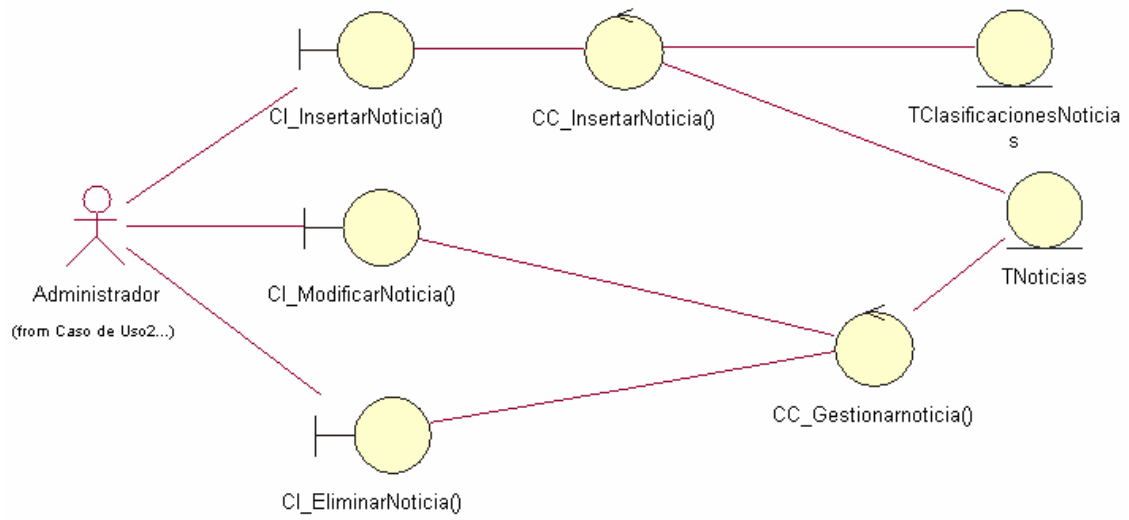
Anexo Caso de Uso 1. Autenticar Usuario



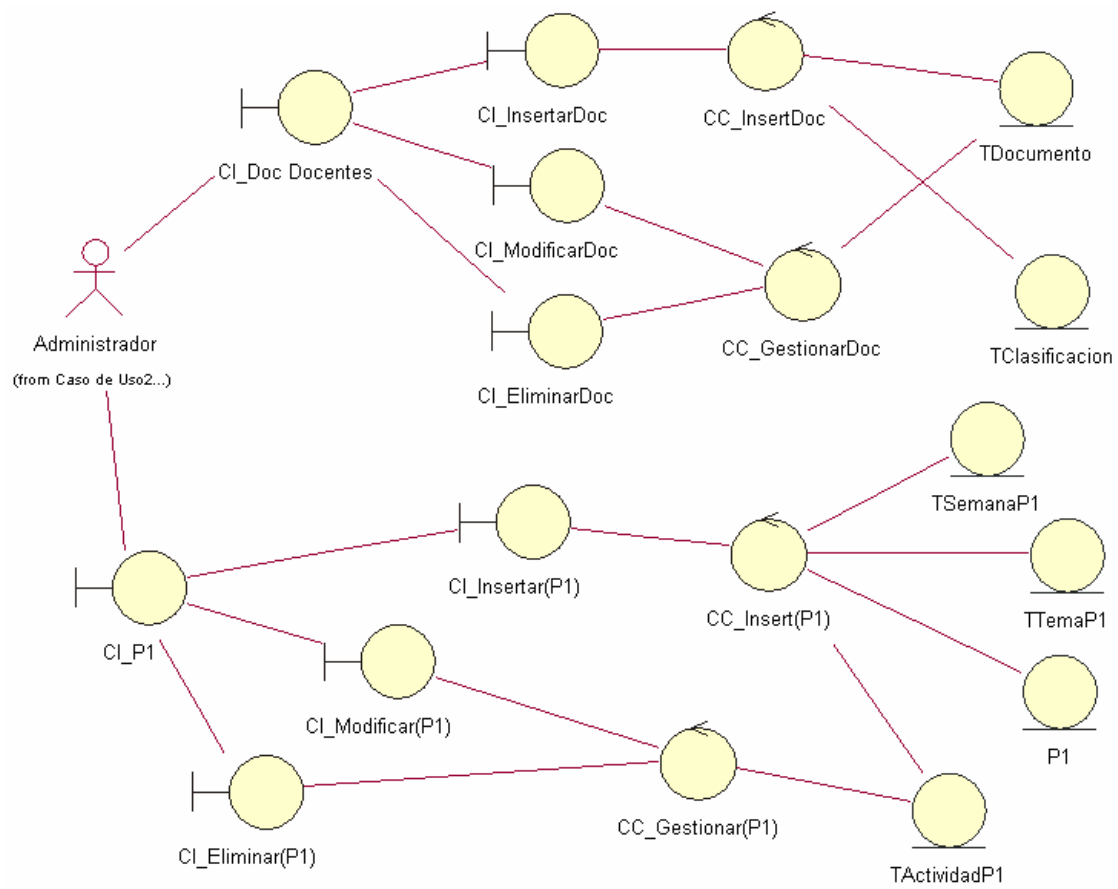
Anexo Caso de uso 2. Gestionar Usuario



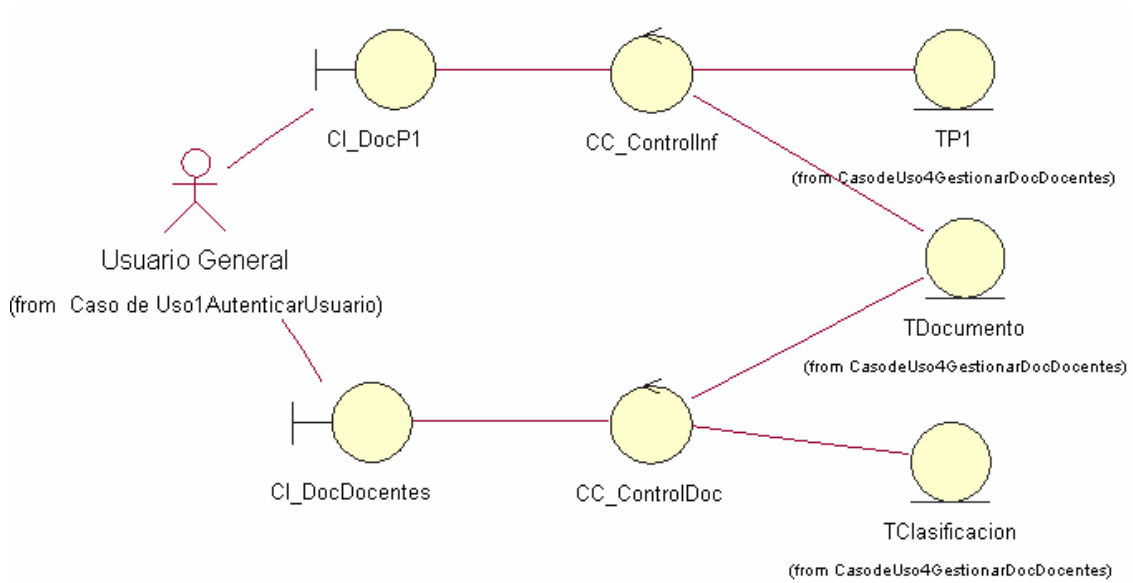
Anexo Caso de Uso 3. Gestionar Noticias



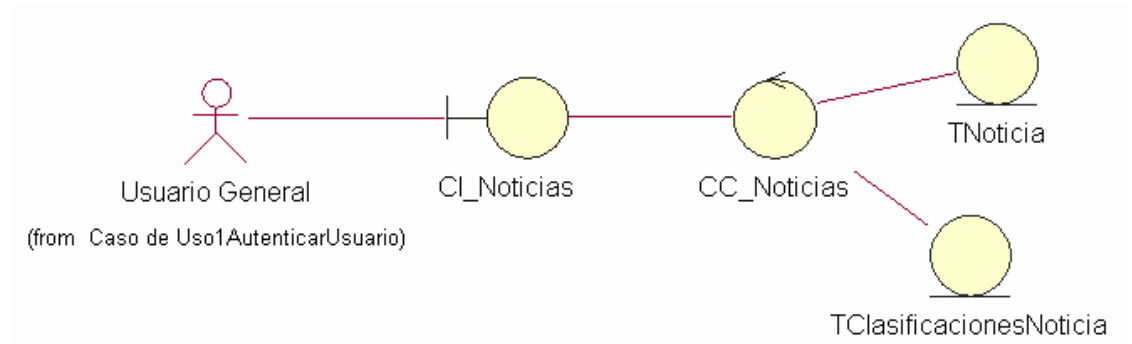
Anexo Caso de Uso 4. Gestionar Documentos docentes



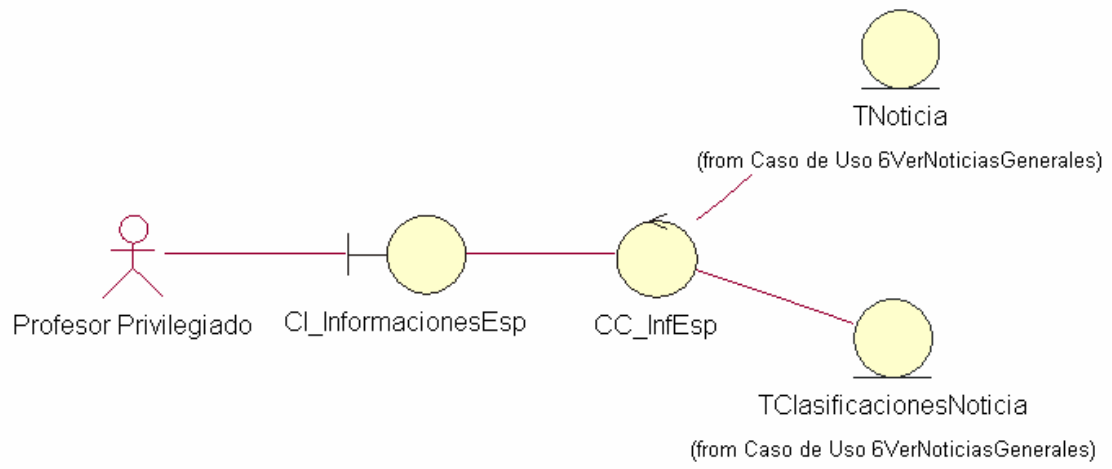
Anexo Caso de Uso 5. Manipular Documentos Docentes



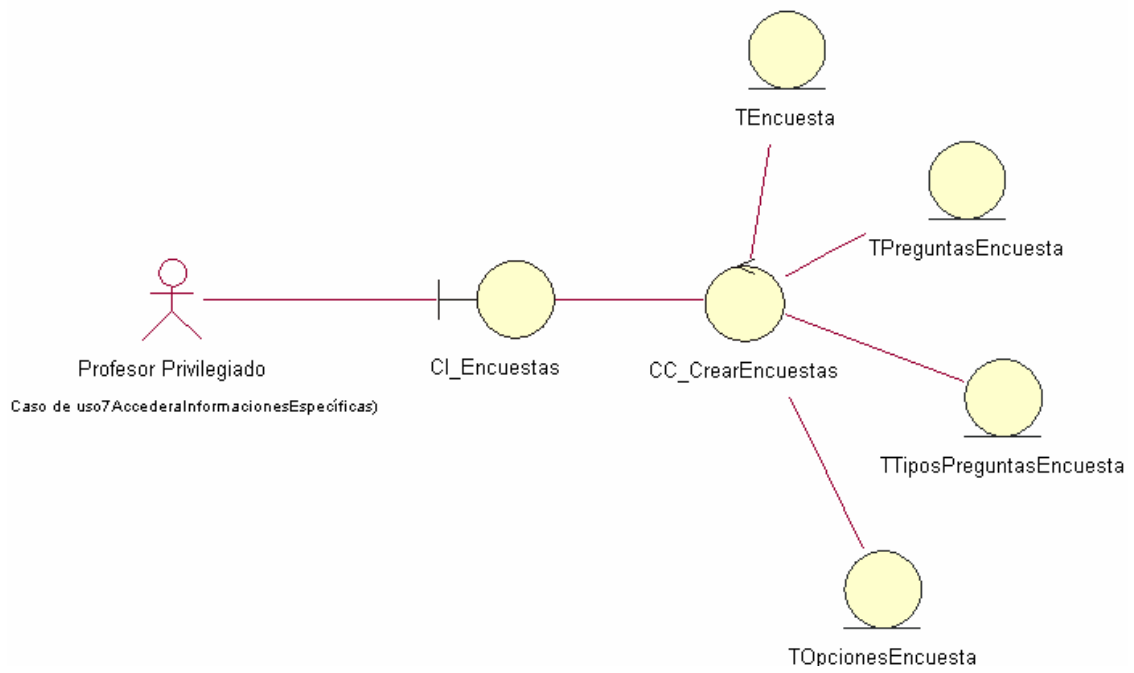
Anexo Caso de Uso 6.Ver Noticias Generales



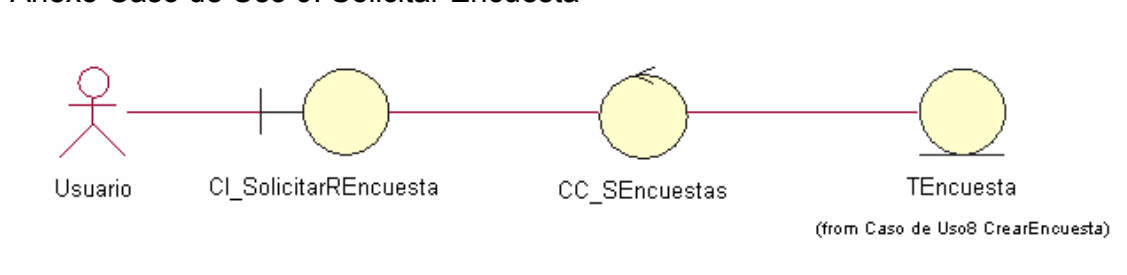
Anexo Caso de uso 7. Acceder a Informaciones Específicas



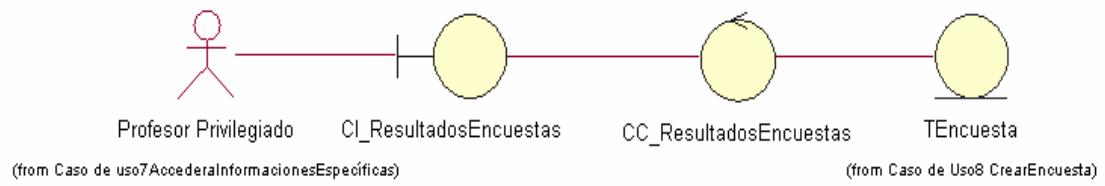
Anexo Caso de Uso 8. Crear Encuesta



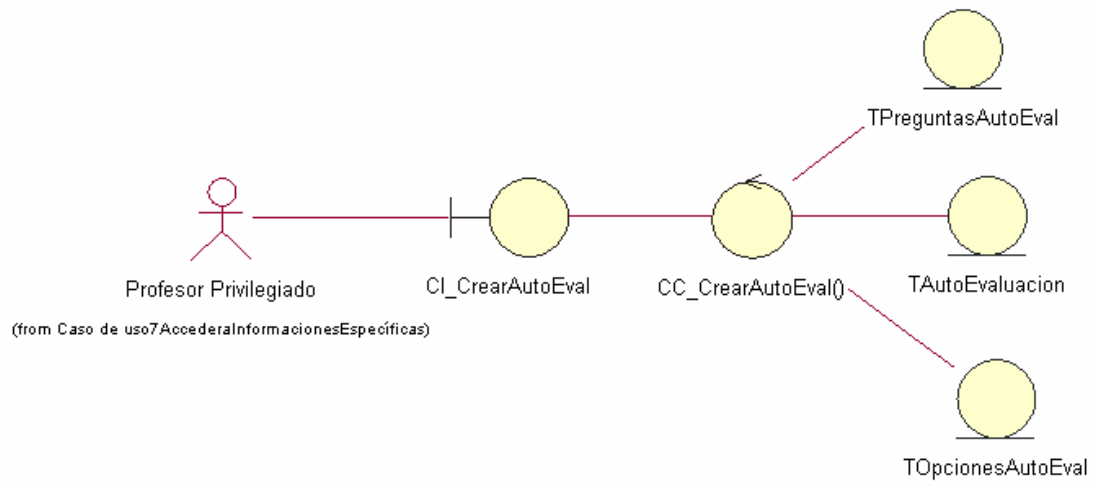
Anexo Caso de Uso 9. Solicitar Encuesta



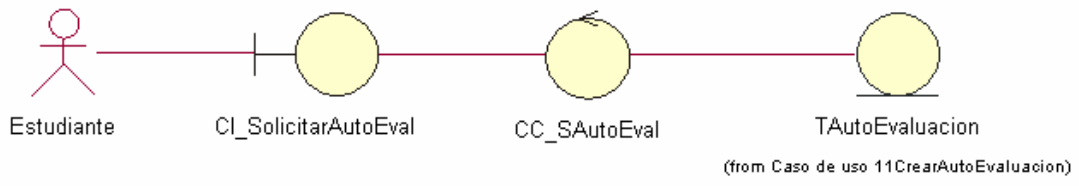
Anexo Caso de Uso 10. Ver Resultado de Encuesta



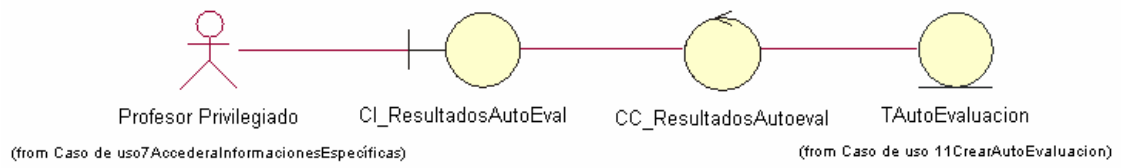
Anexo Caso de Uso 11. Crear Auto evaluación



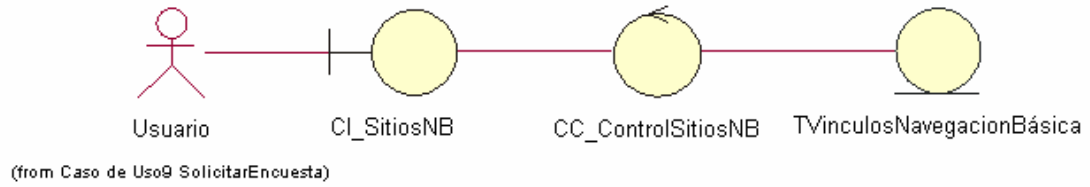
Anexo Caso de Uso 12. Solicitar Auto evaluación



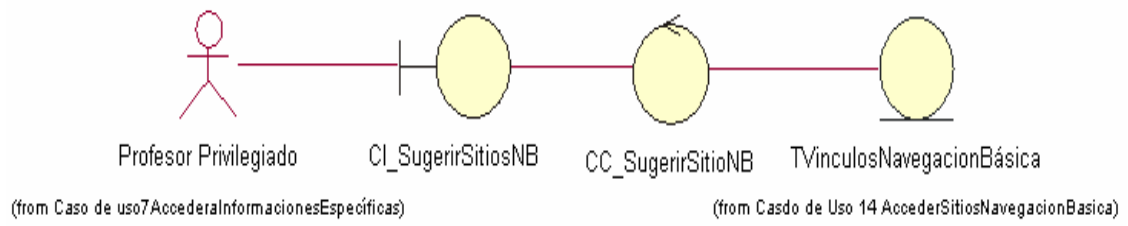
Anexo Caso de Uso 13. Ver Resultado de Auto evaluación



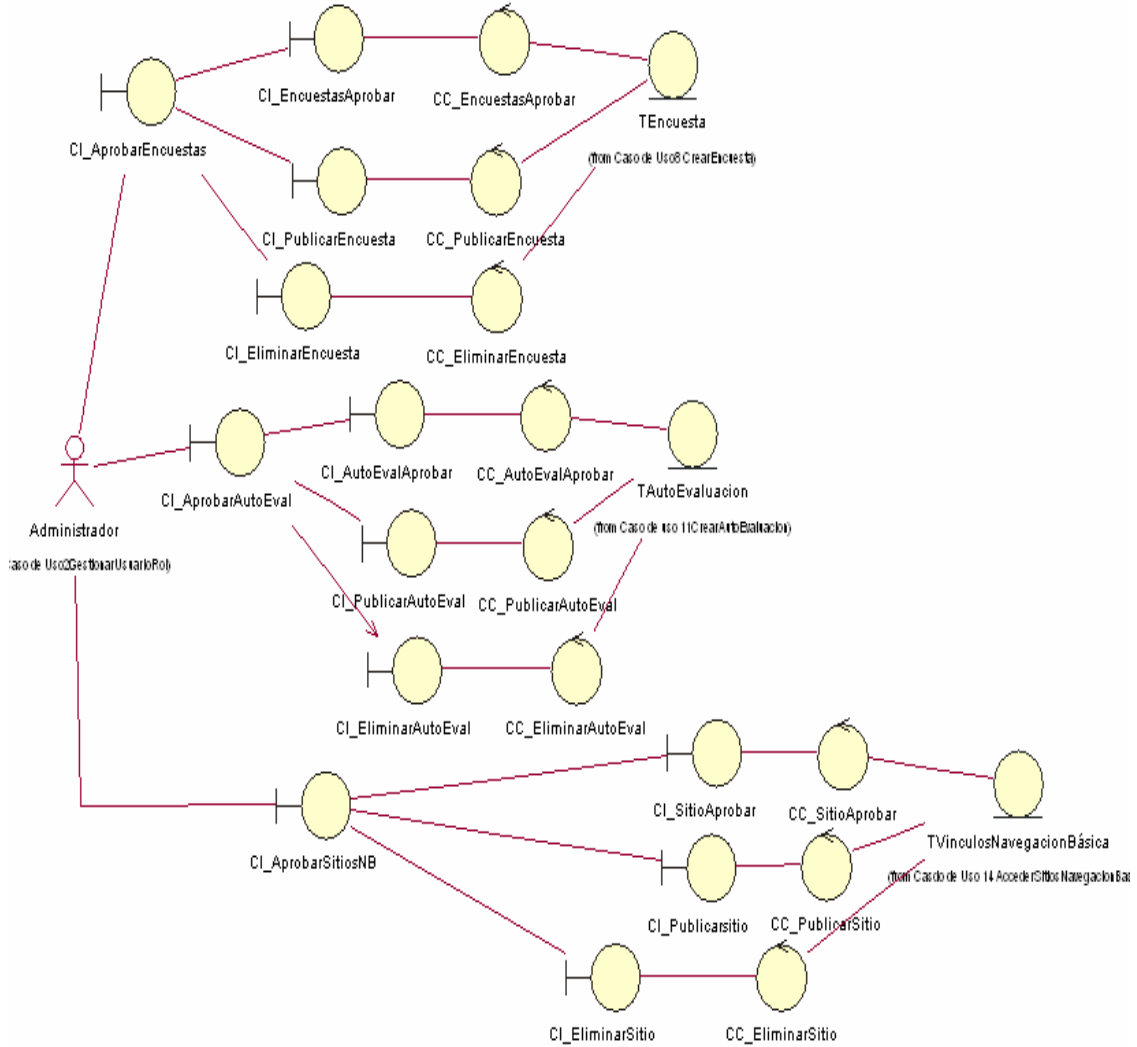
Anexo Caso de Uso 14. Acceder al Sitio de Navegación Básica



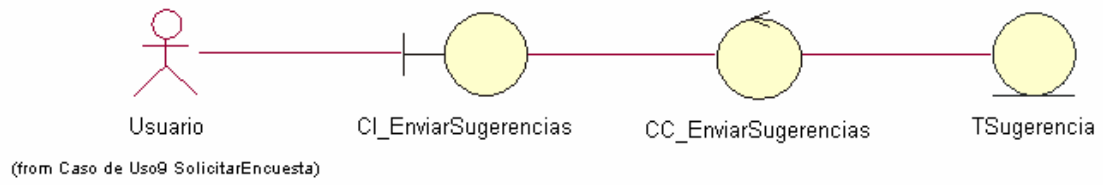
Anexo Caso de Uso 15. Sugerir Sitio de Navegación Básica



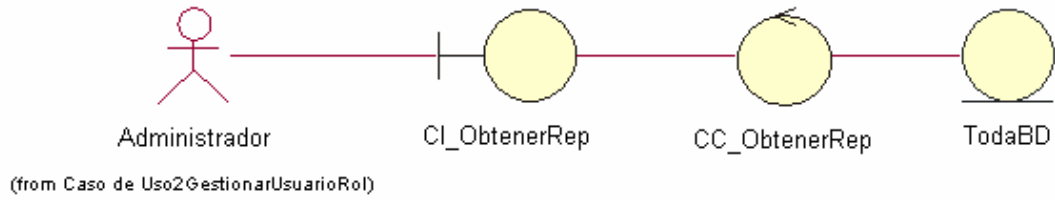
Anexo Caso de Uso 16. Aprobar solicitudes de Profesores privilegiados



Anexo Caso de Uso 17. Enviar sugerencias al Administrador del Sitio

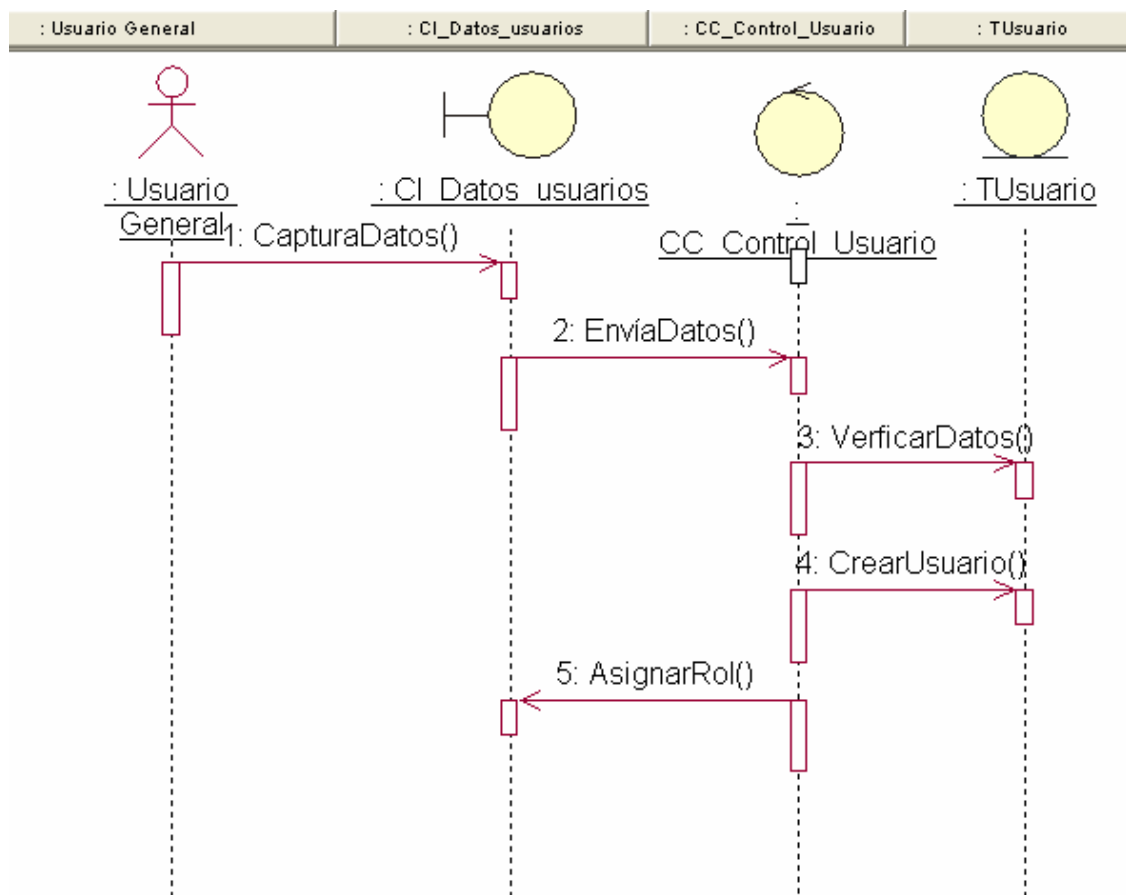
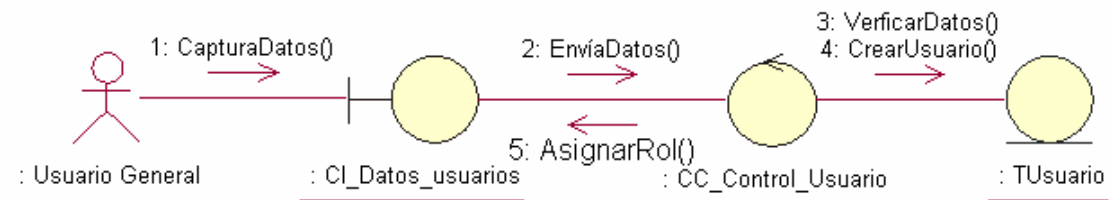


Anexo Caso de Uso 18. Obtener Reportes

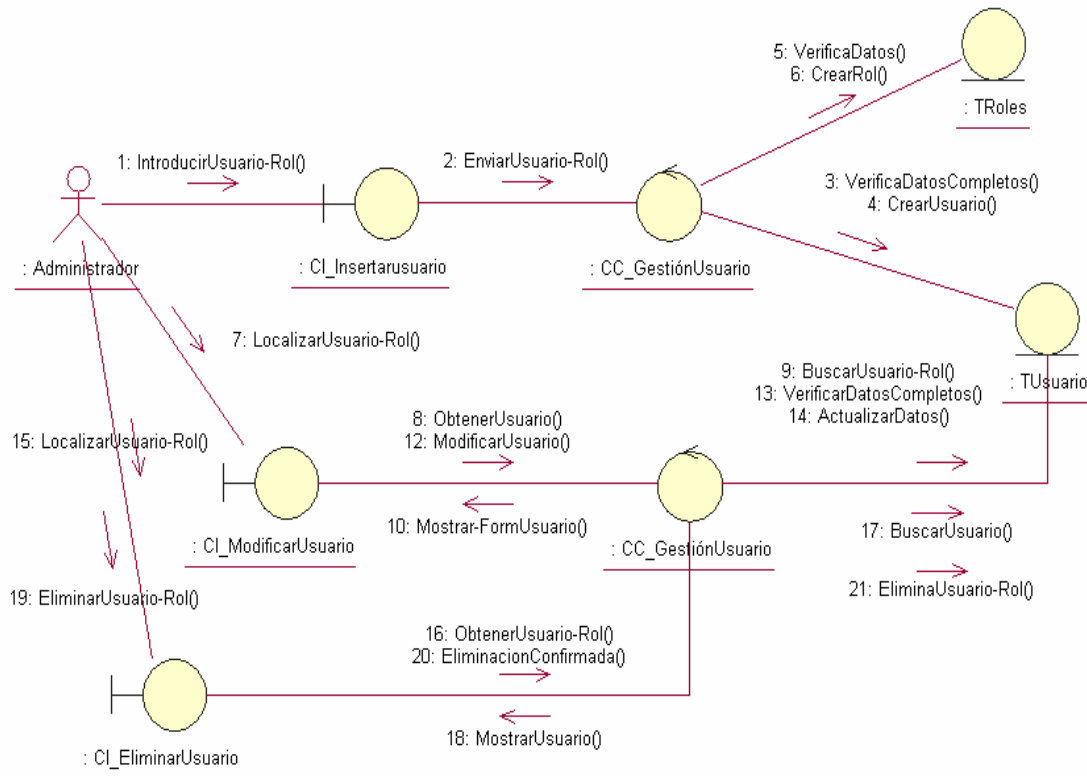


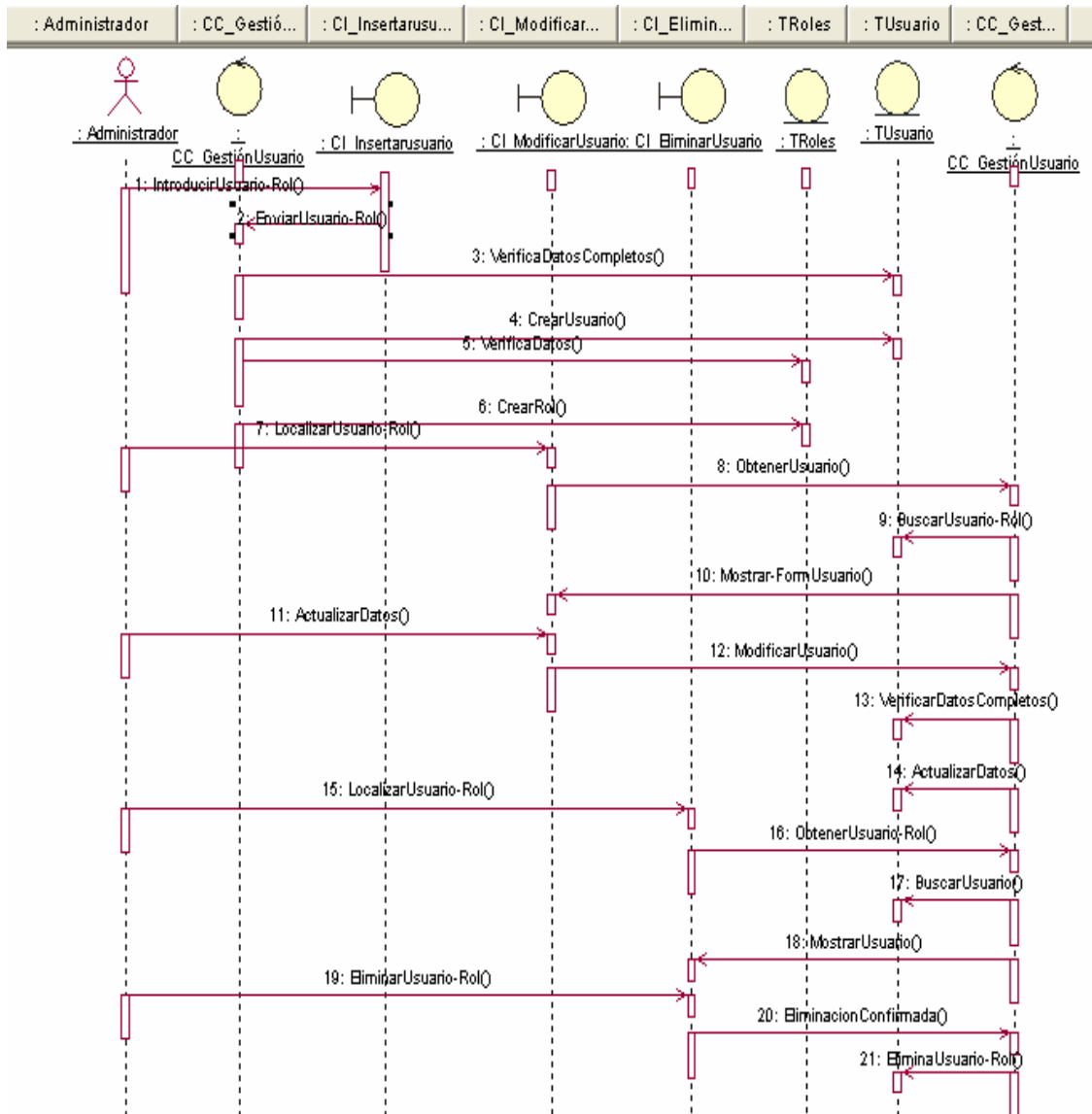
Anexo 2 Diagramas de Interacción

Anexo Caso de Uso 1. Autenticar Usuario

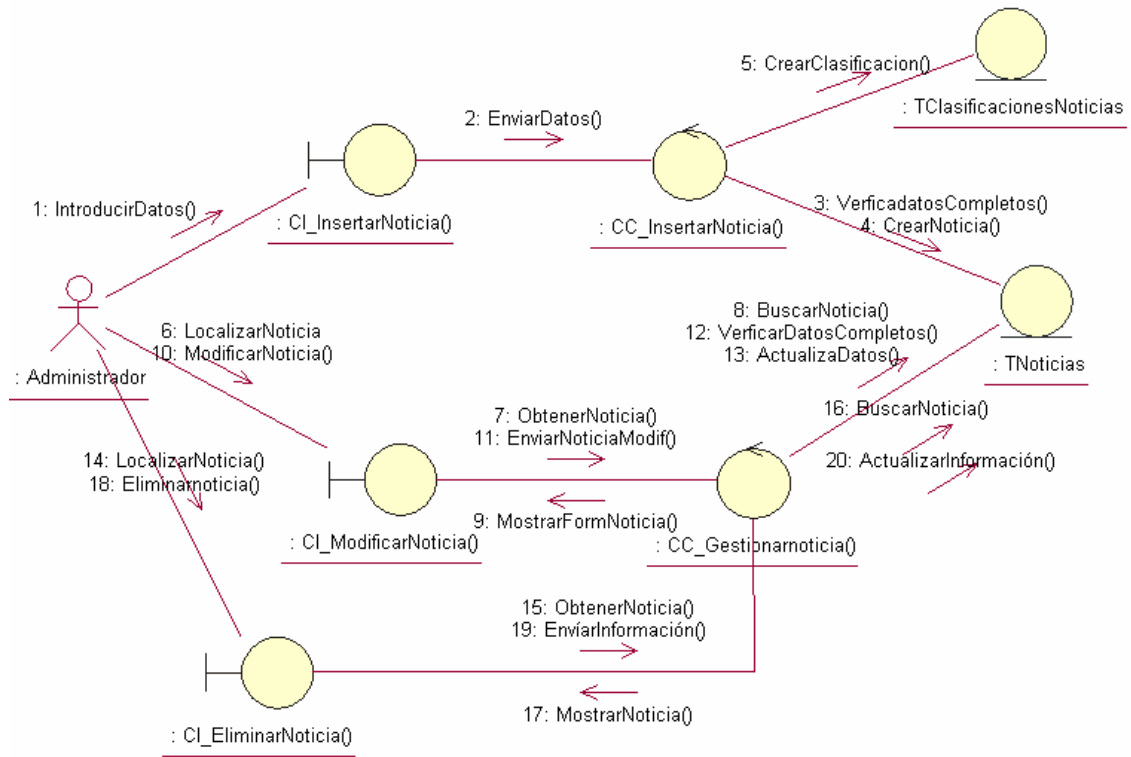


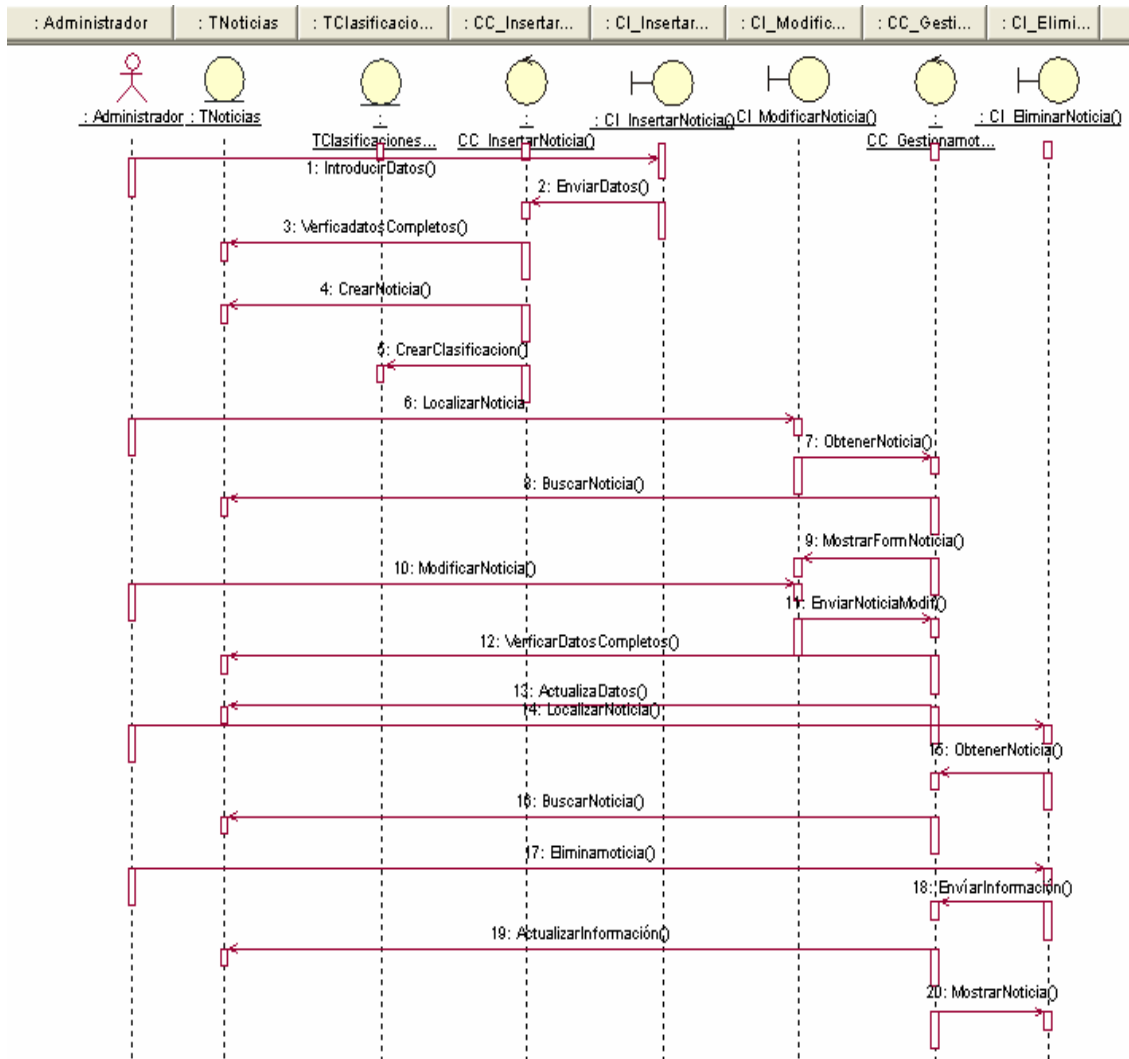
Anexo Caso de Uso 2. Gestionar Usuario



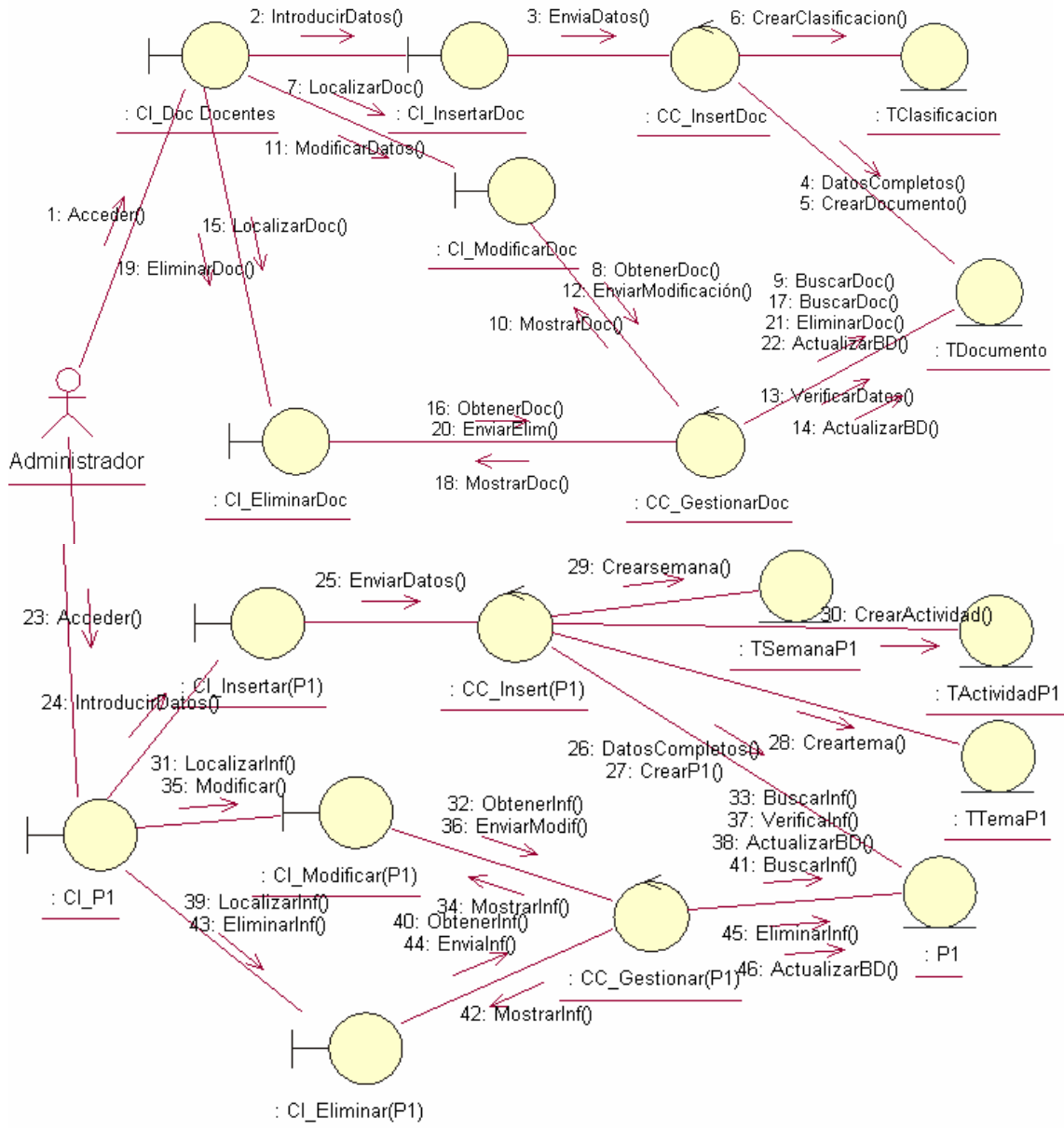


Anexo Caso de Uso 3. Gestionar Noticia

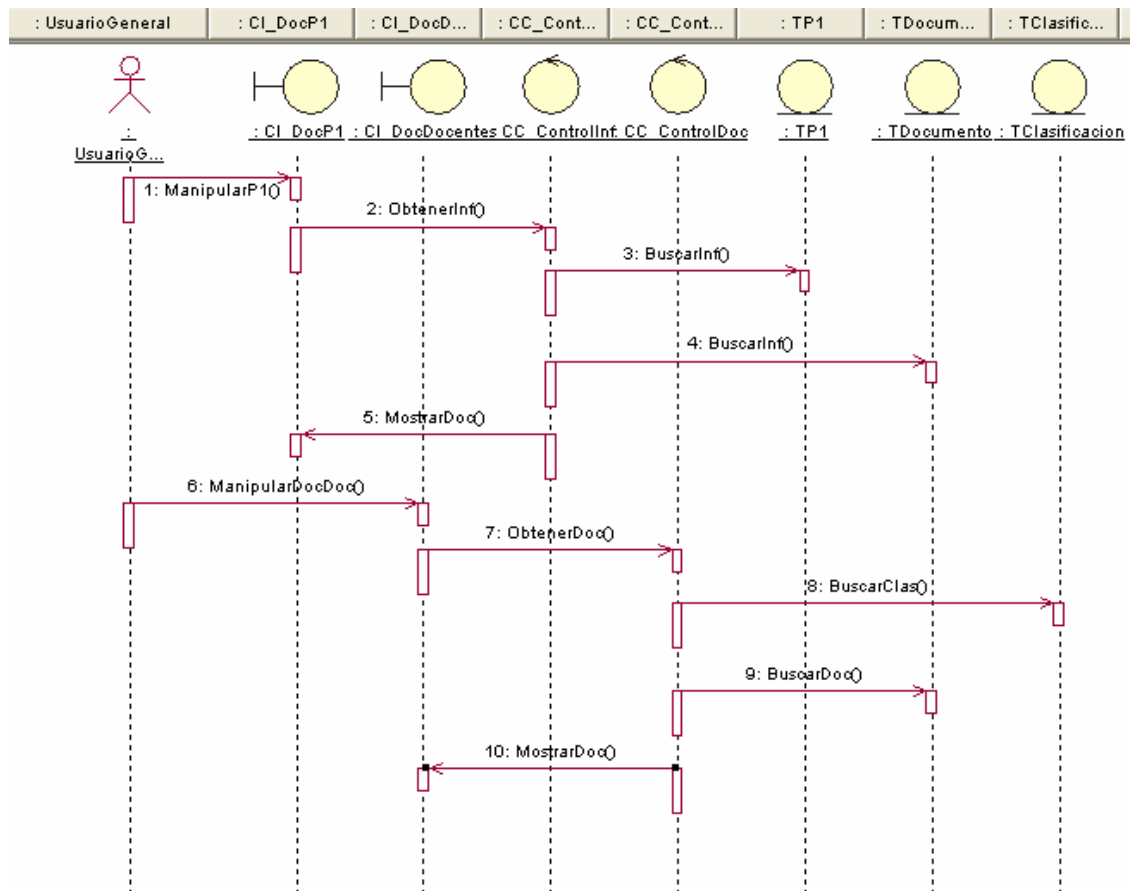
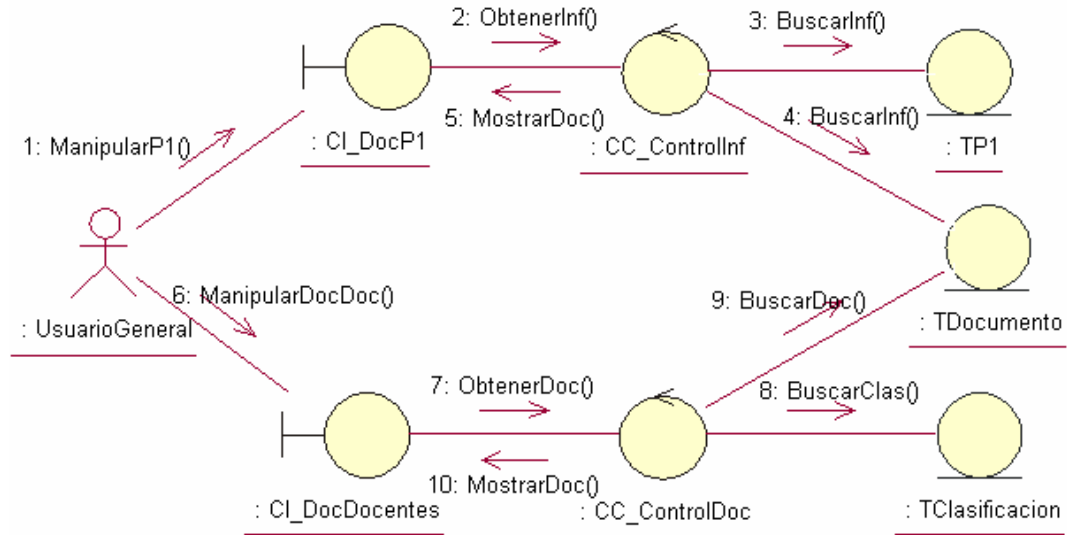




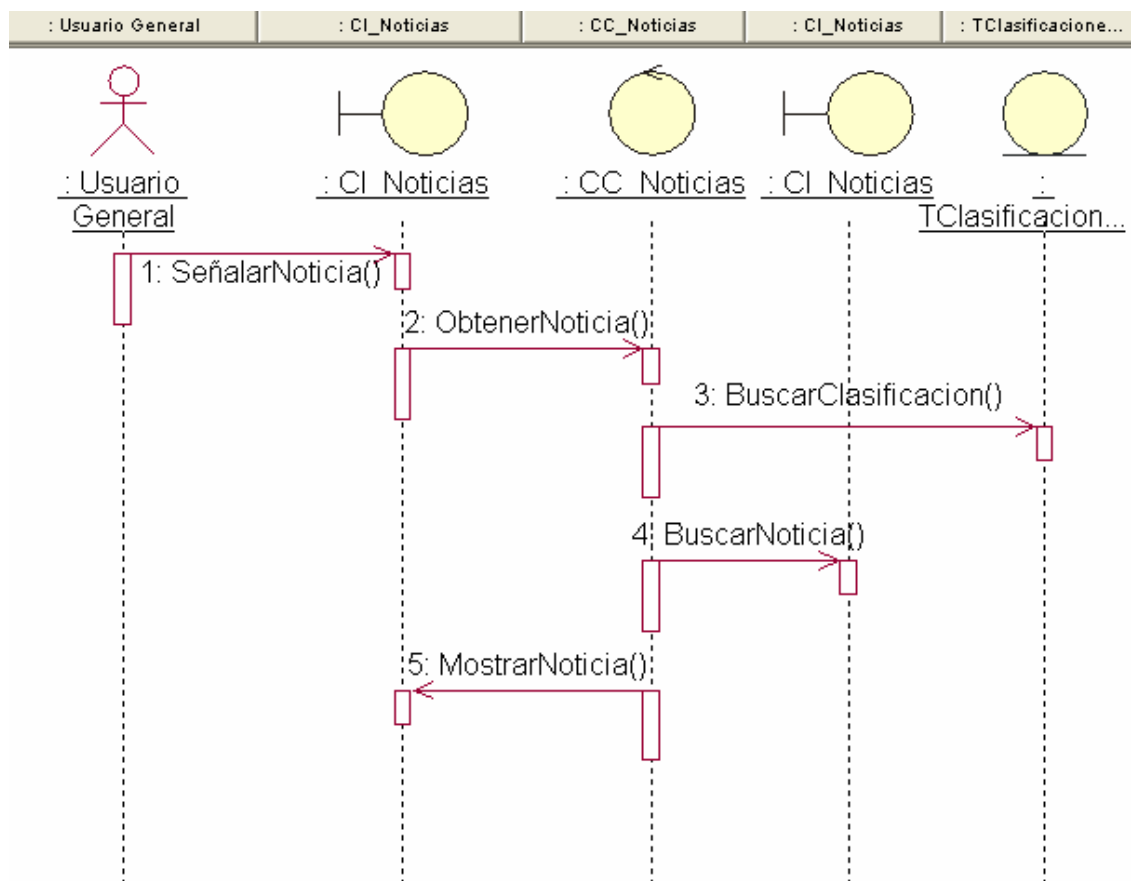
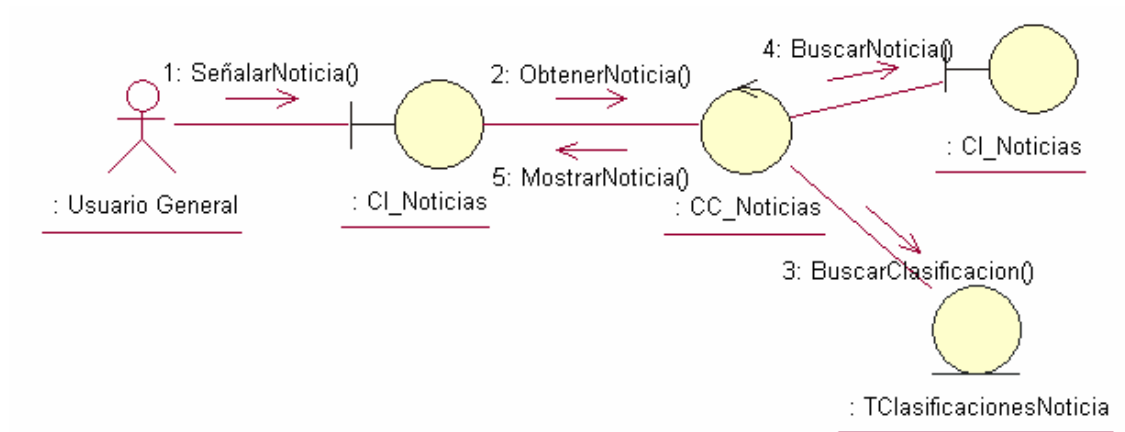
Anexo Caso de Uso 4. Gestionar Documentos Docentes



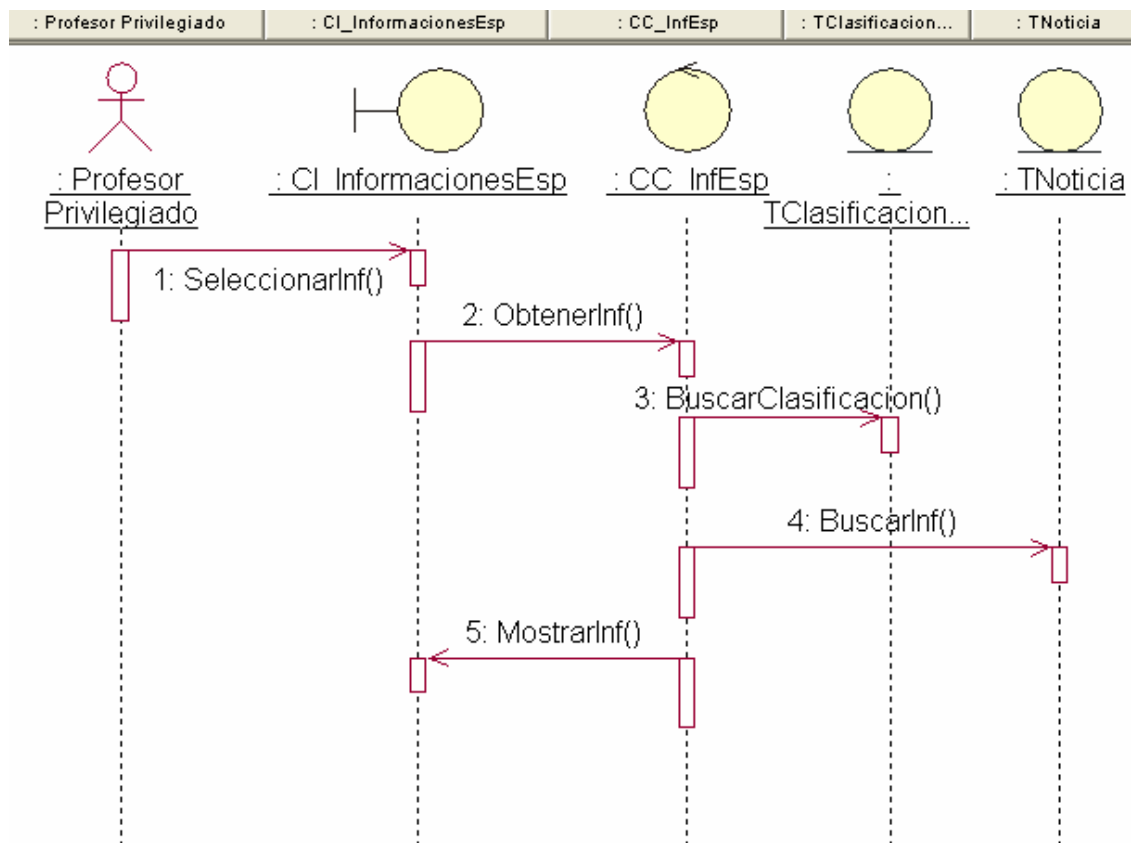
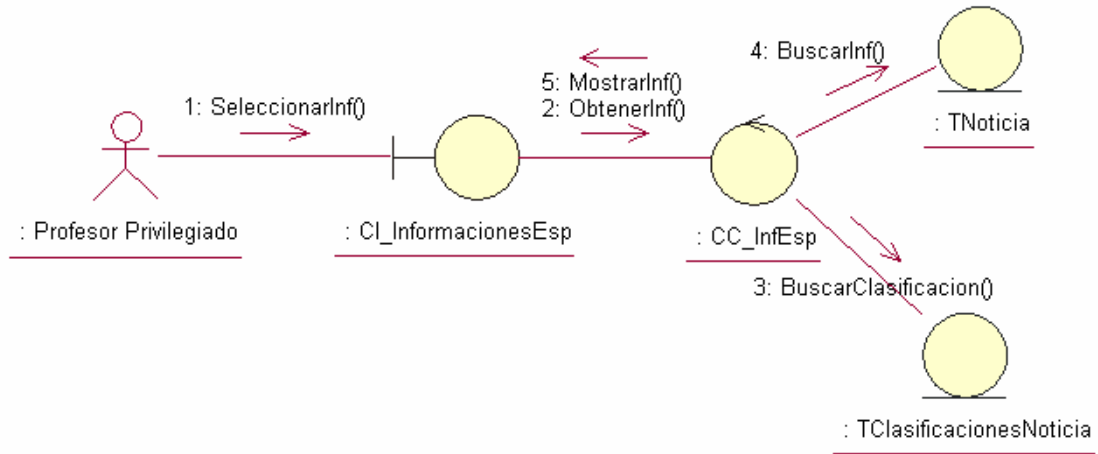
Anexo Caso de Uso 5. Manipular documentos Docentes



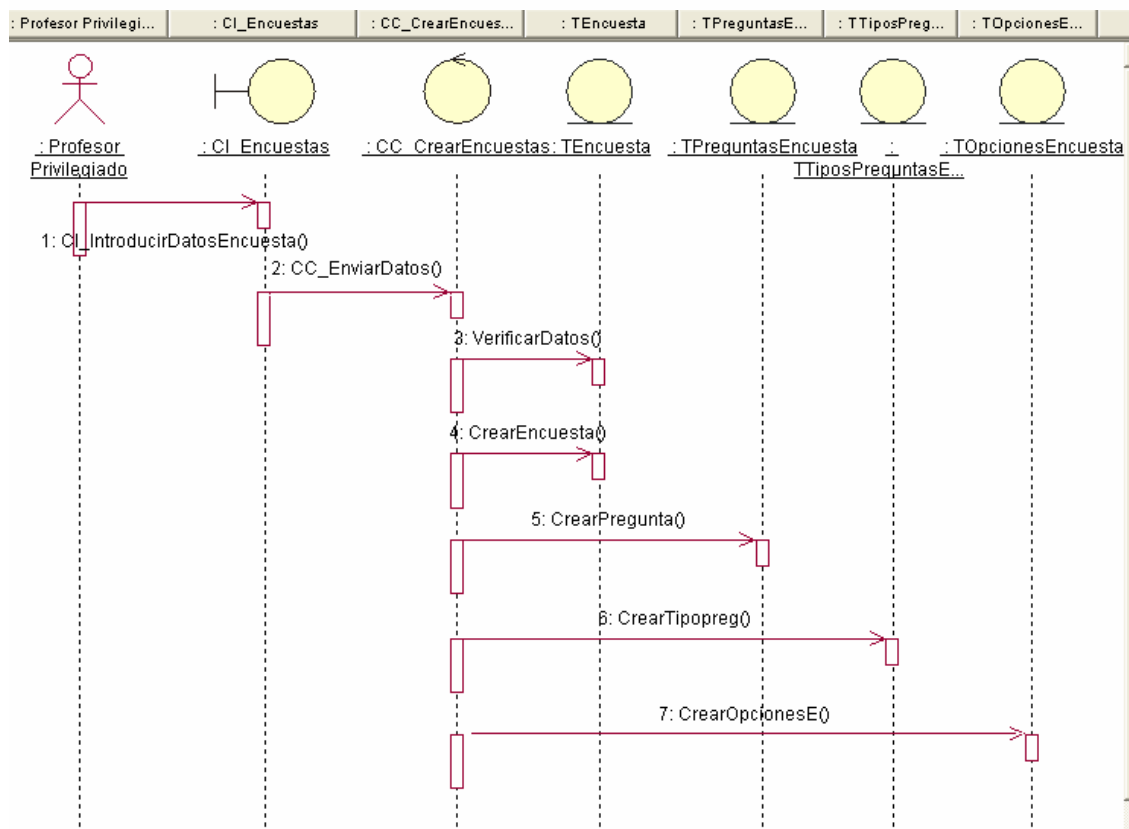
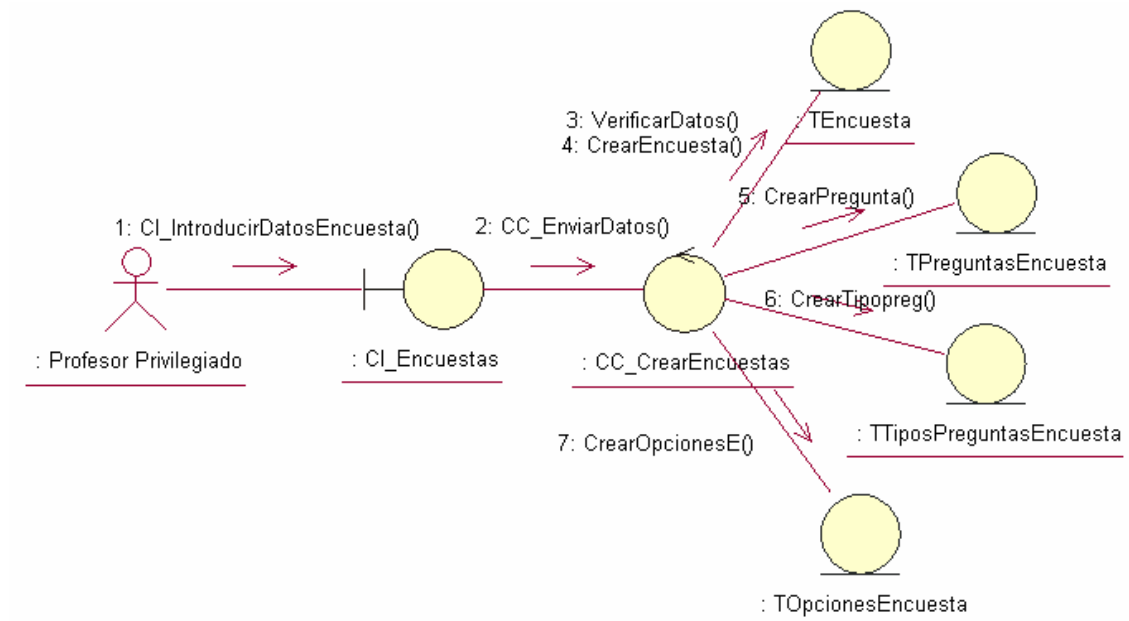
Anexo Caso de Uso 6. Ver Noticias Generales



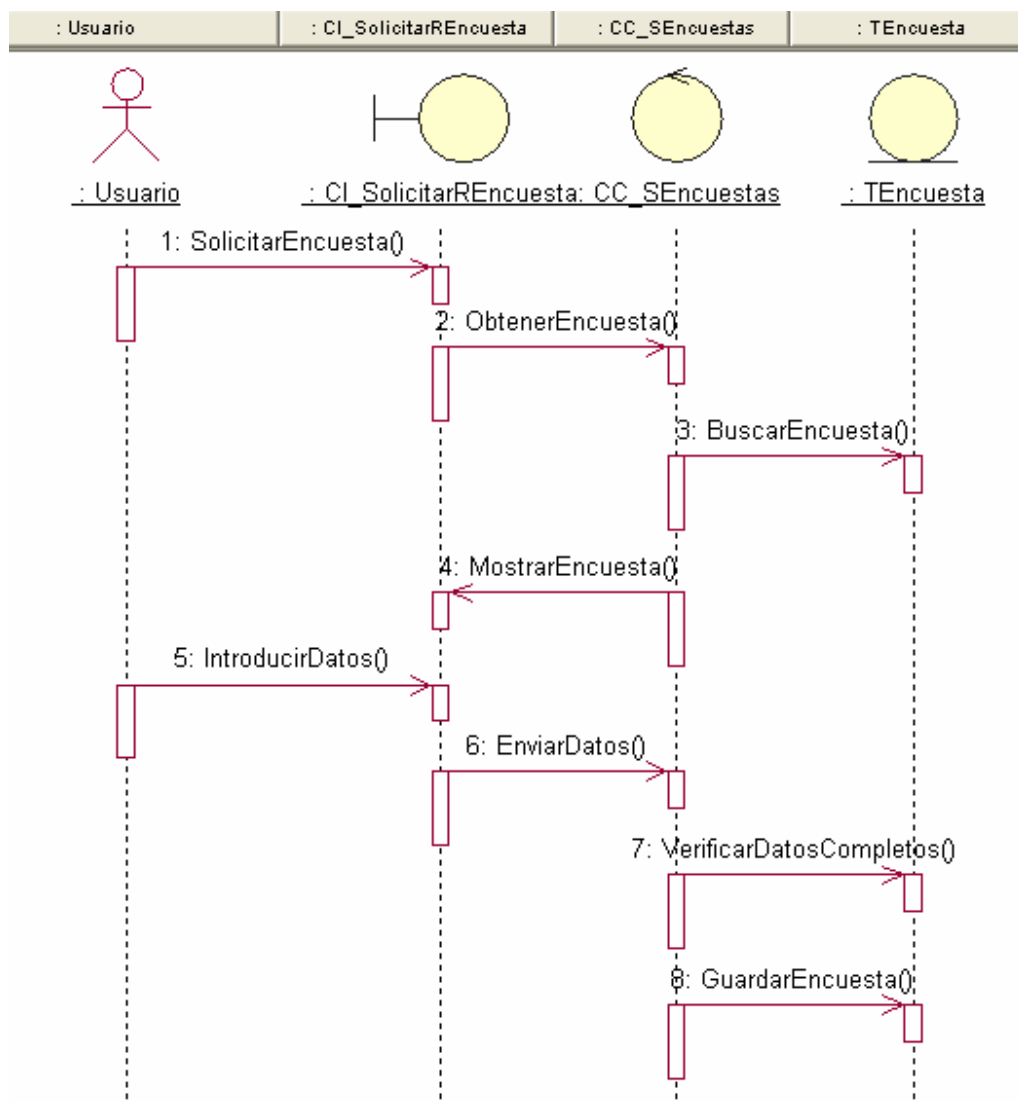
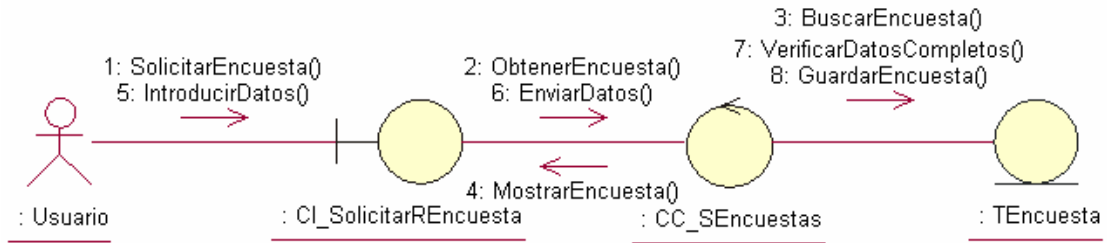
Anexo Caso de Uso 7. Acceder a Informaciones Específicas



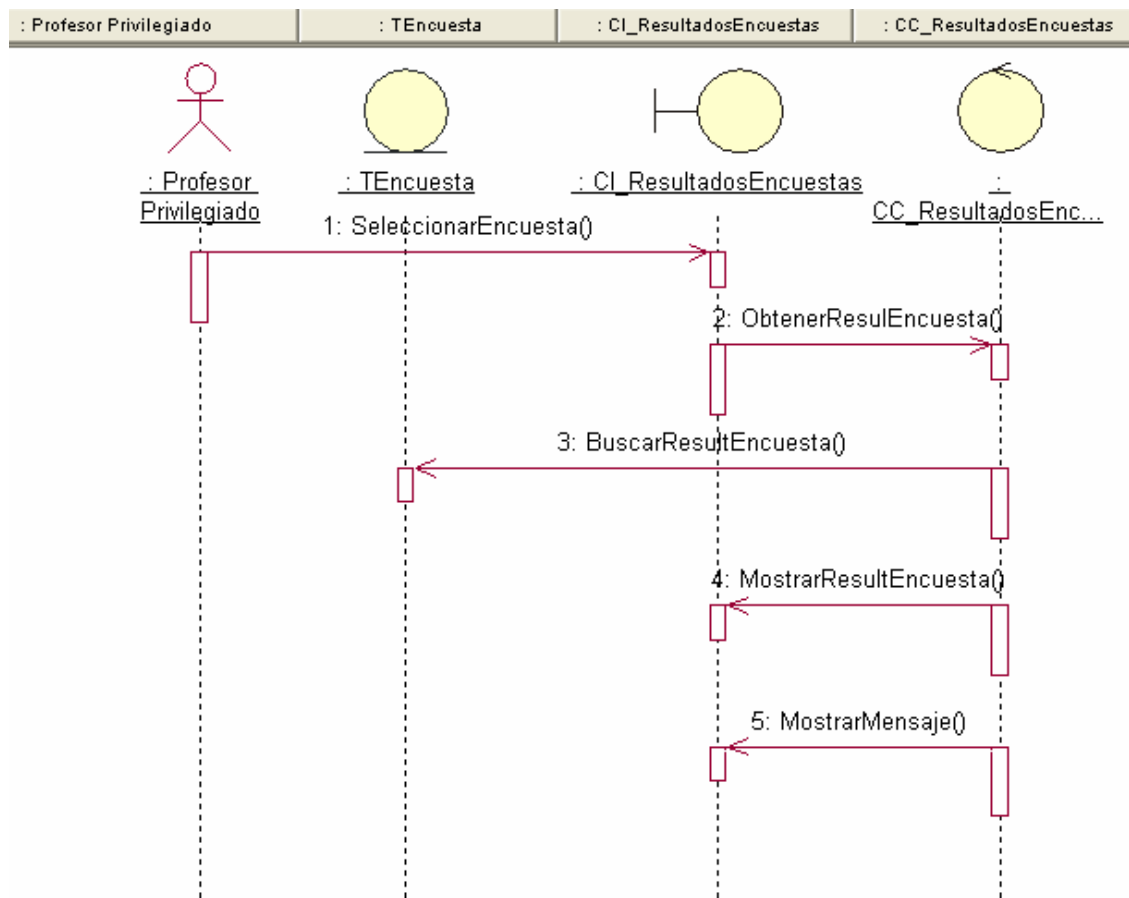
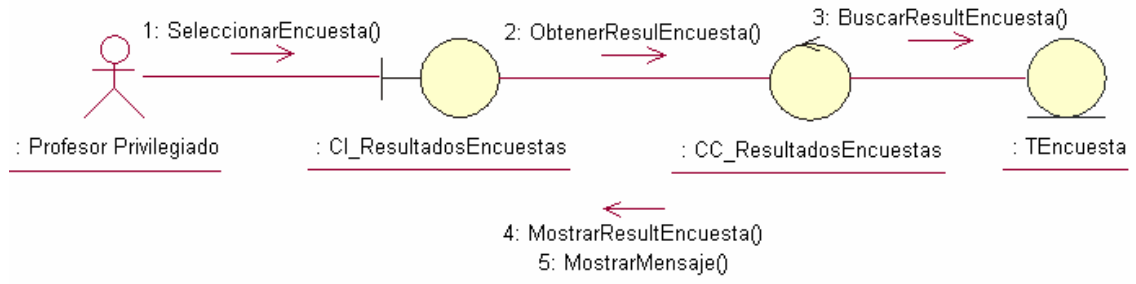
Anexo Caso de uso 8. Crear Encuesta



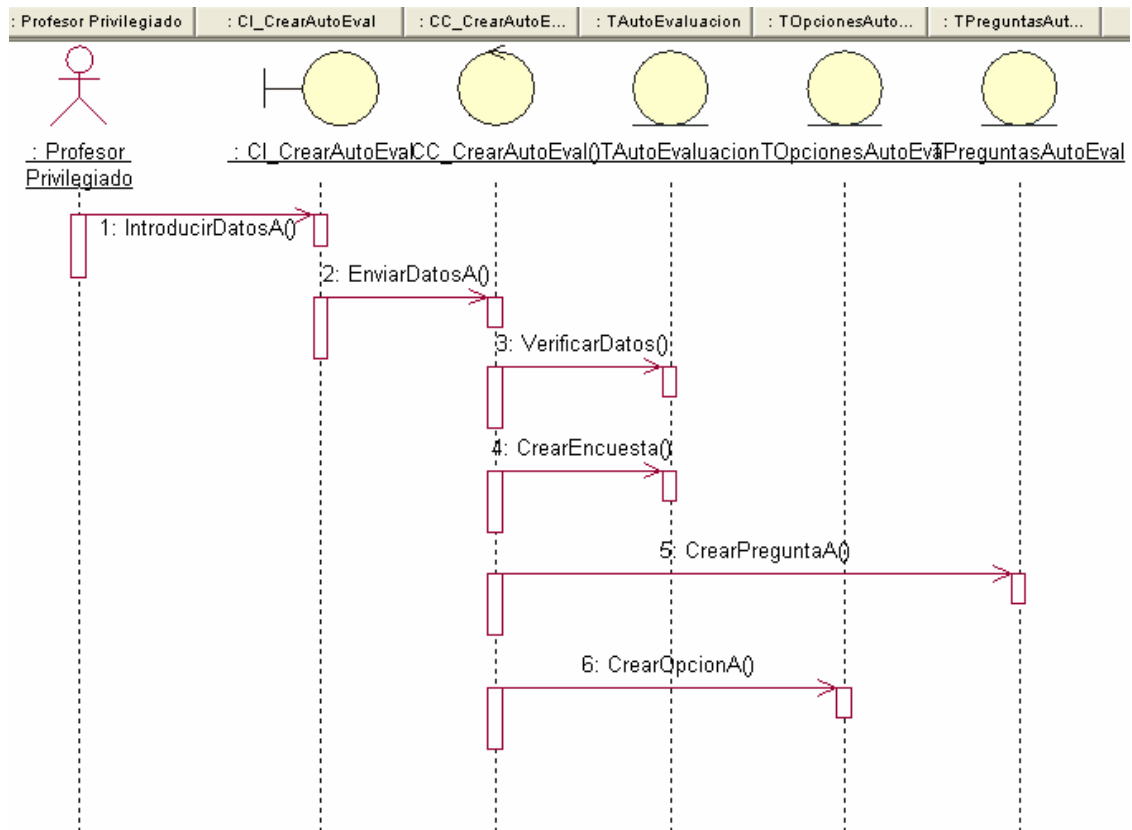
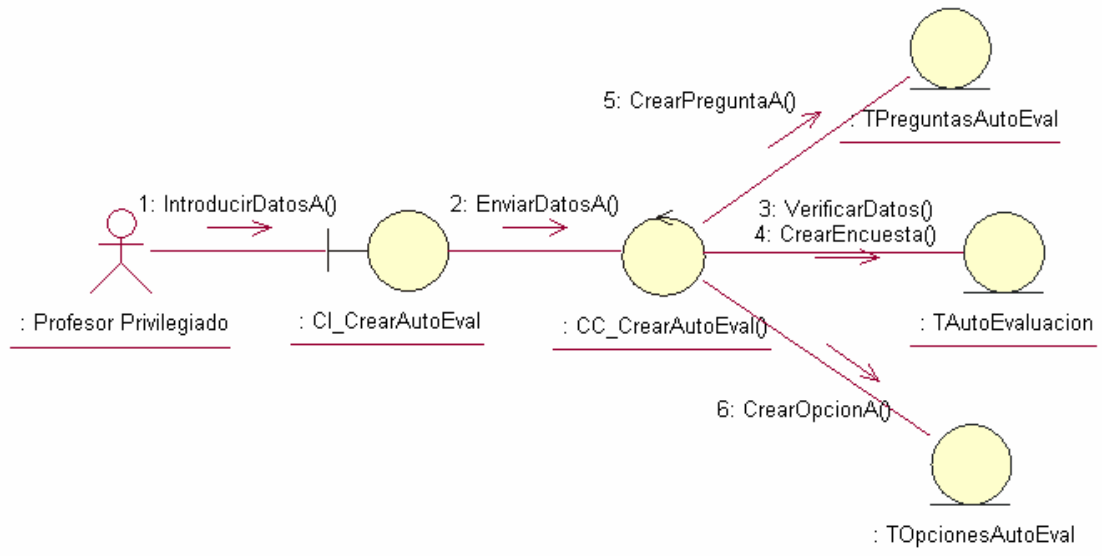
Anexo Caso de Uso 9. Solicitar Encuesta



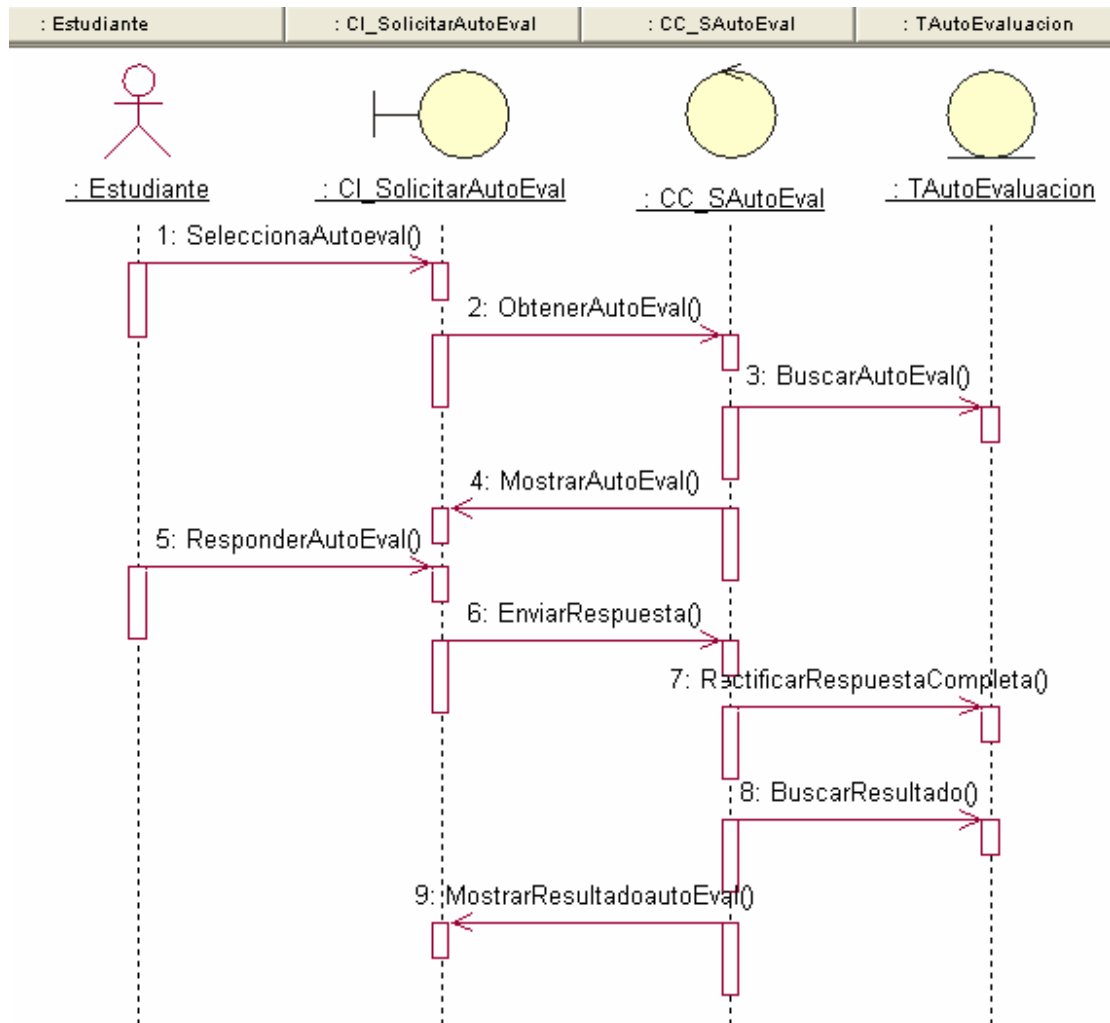
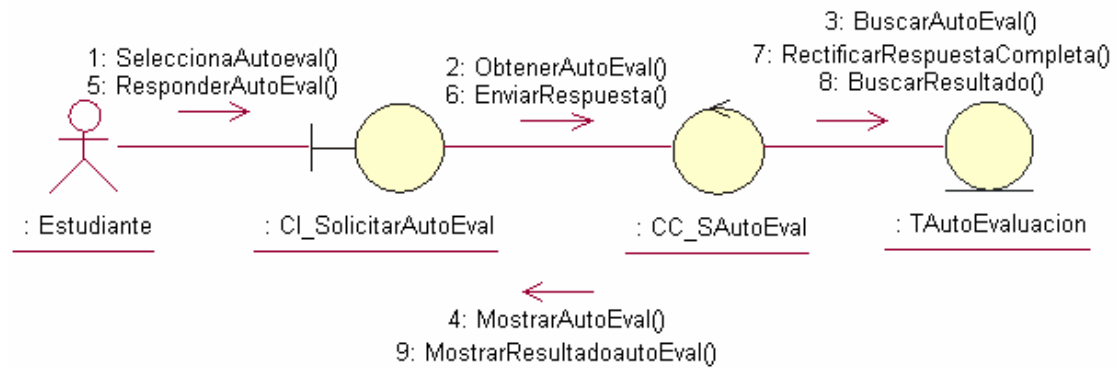
Anexo Caso de Uso 10. Ver Resultados de Encuesta



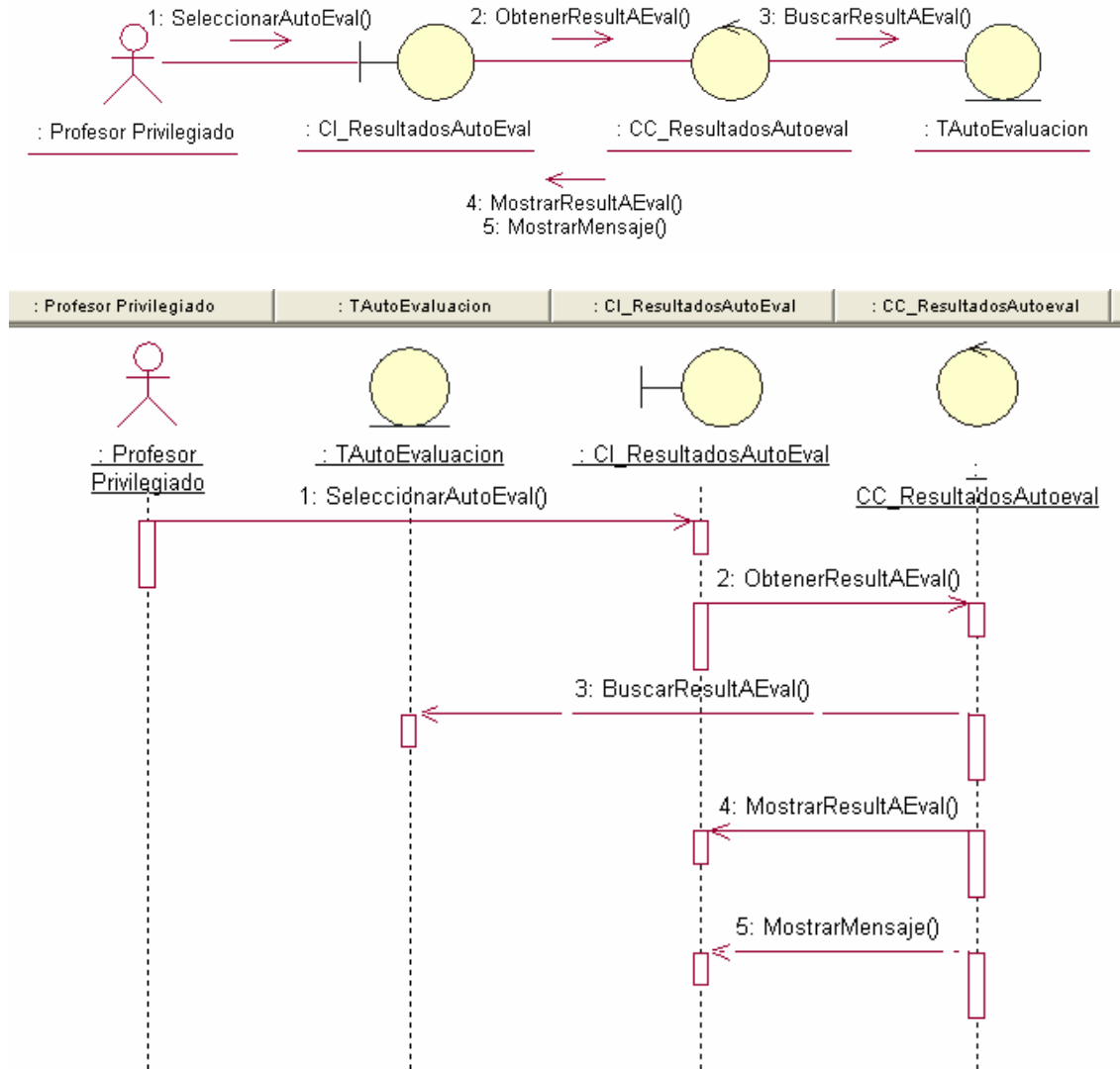
Anexo Caso de Uso 11. Crear Auto evaluación



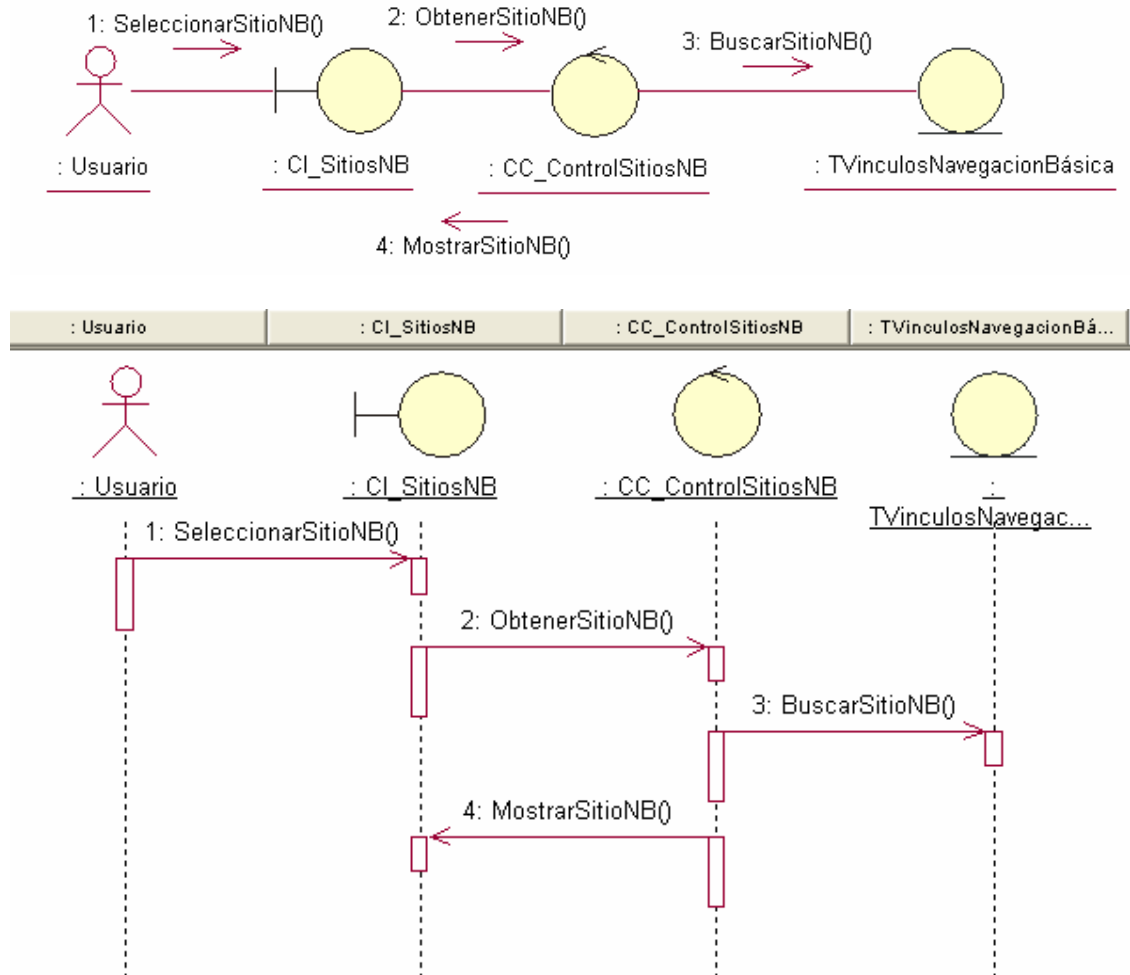
Anexo Caso de Uso 12. Solicitar Auto evaluación



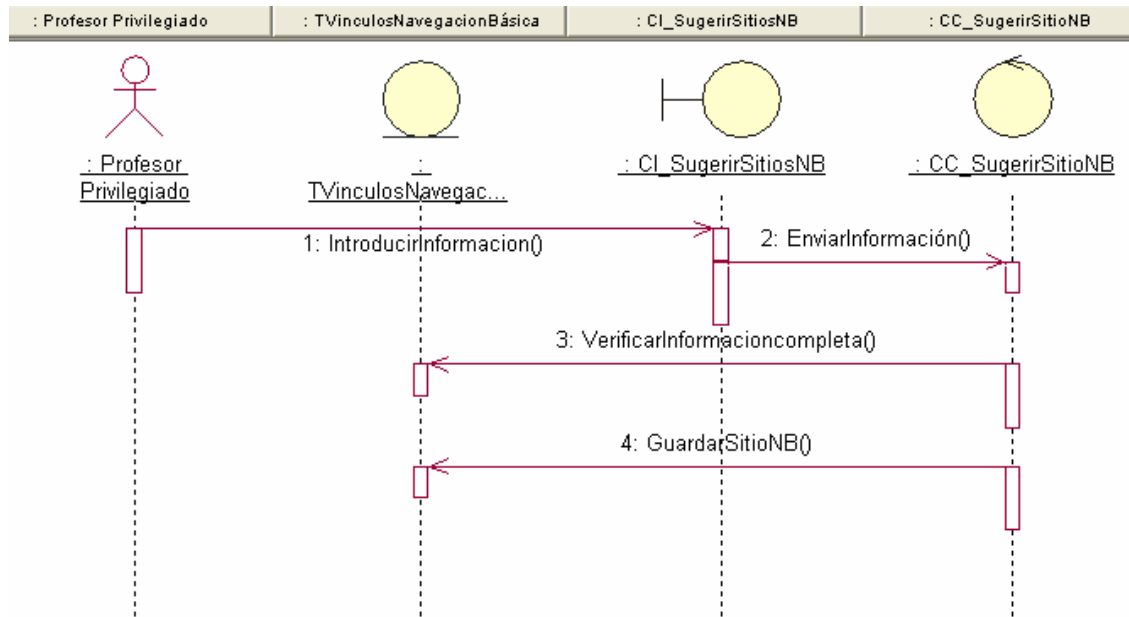
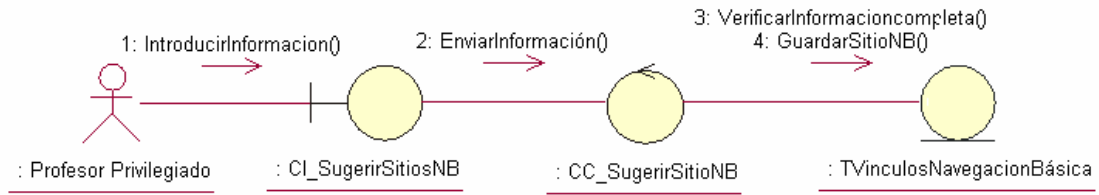
Anexo Caso de Uso 13. Ver Resultado de Auto evaluación



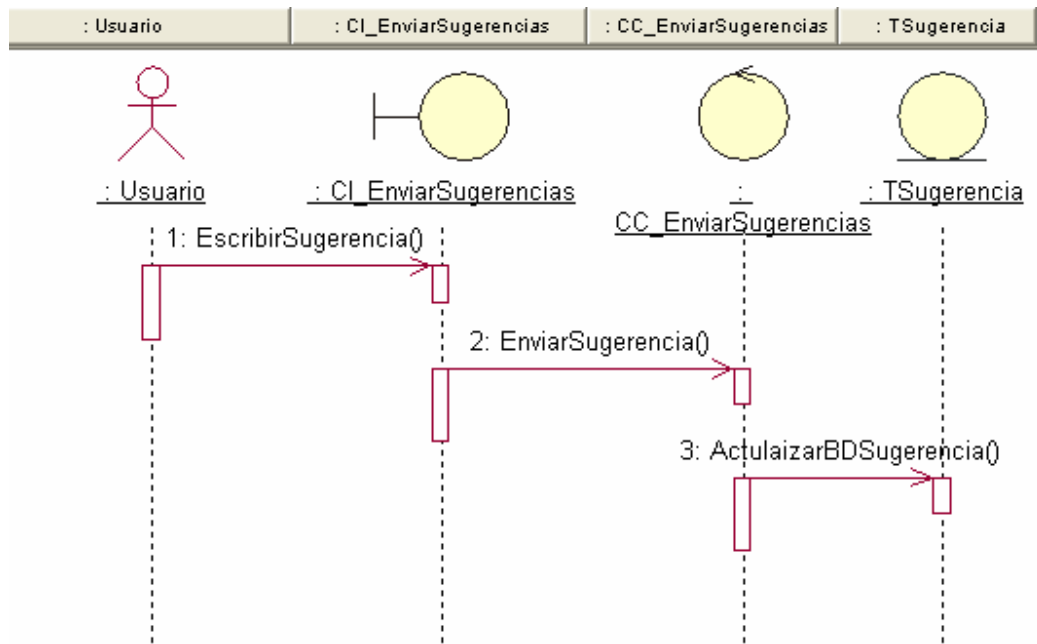
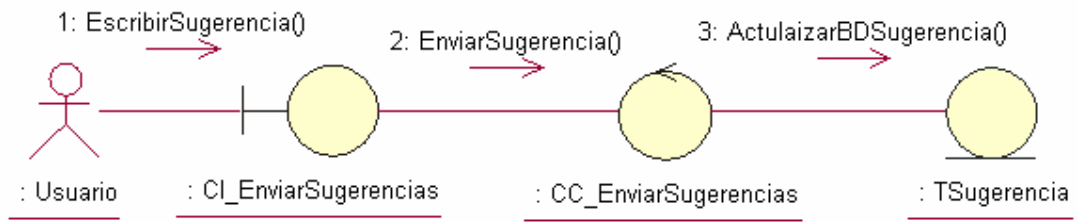
Anexo Caso de Uso 14. Acceder a sitios de Navegación Básica



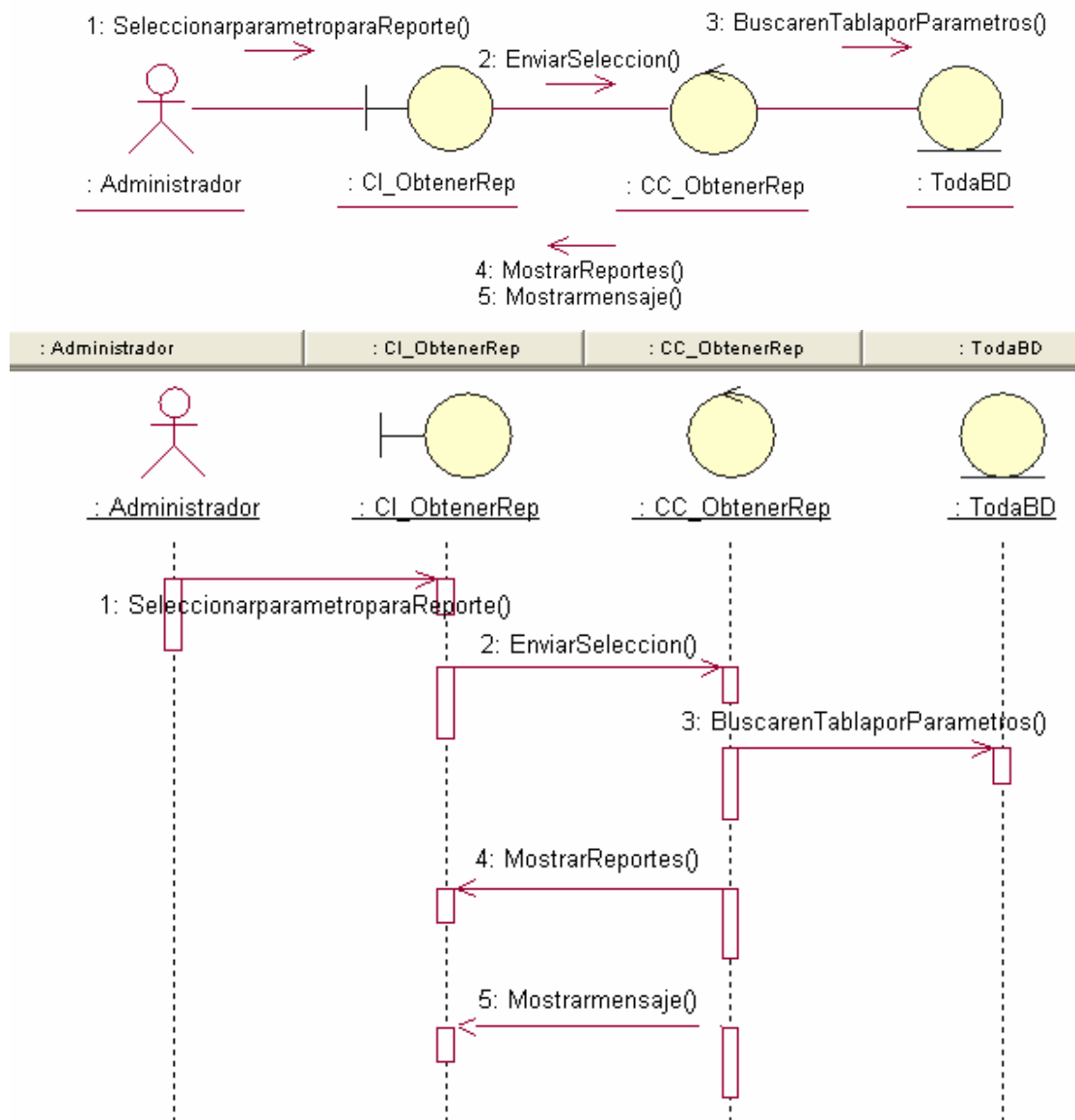
Anexo Caso de Uso 15. Sugerir Sitios de Navegación Básica



Anexo.Caso de Uso 17. Enviar Sugerencias al Administrador del Sitio

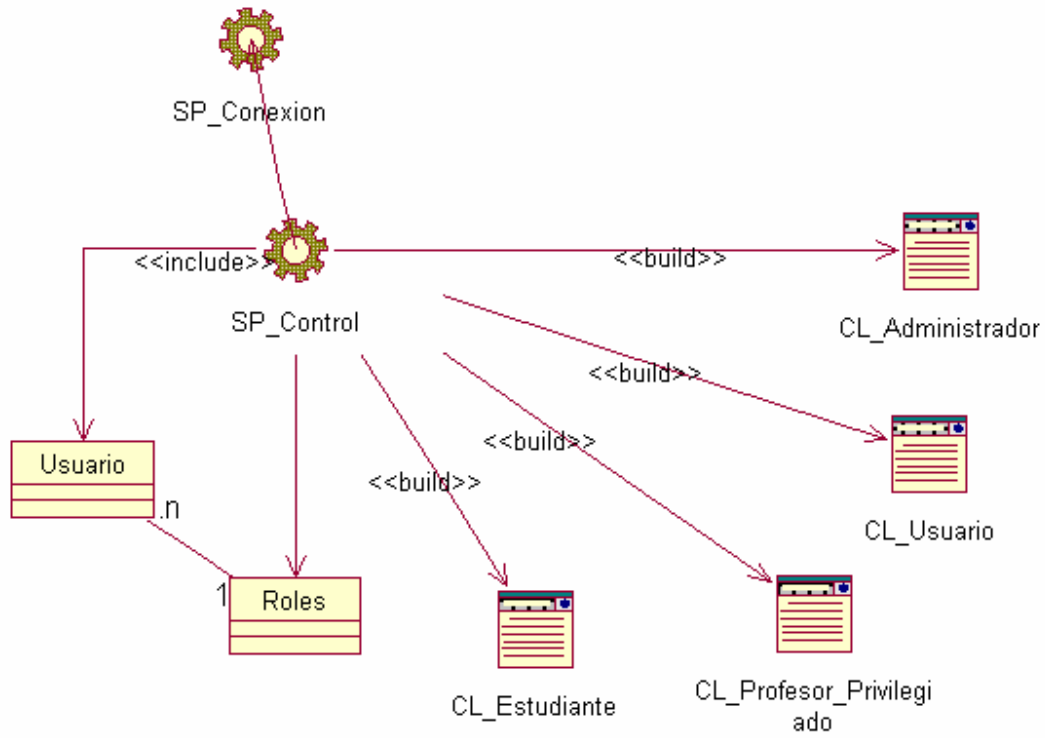


Anexo Caso de Uso 18. Obtener Reportes

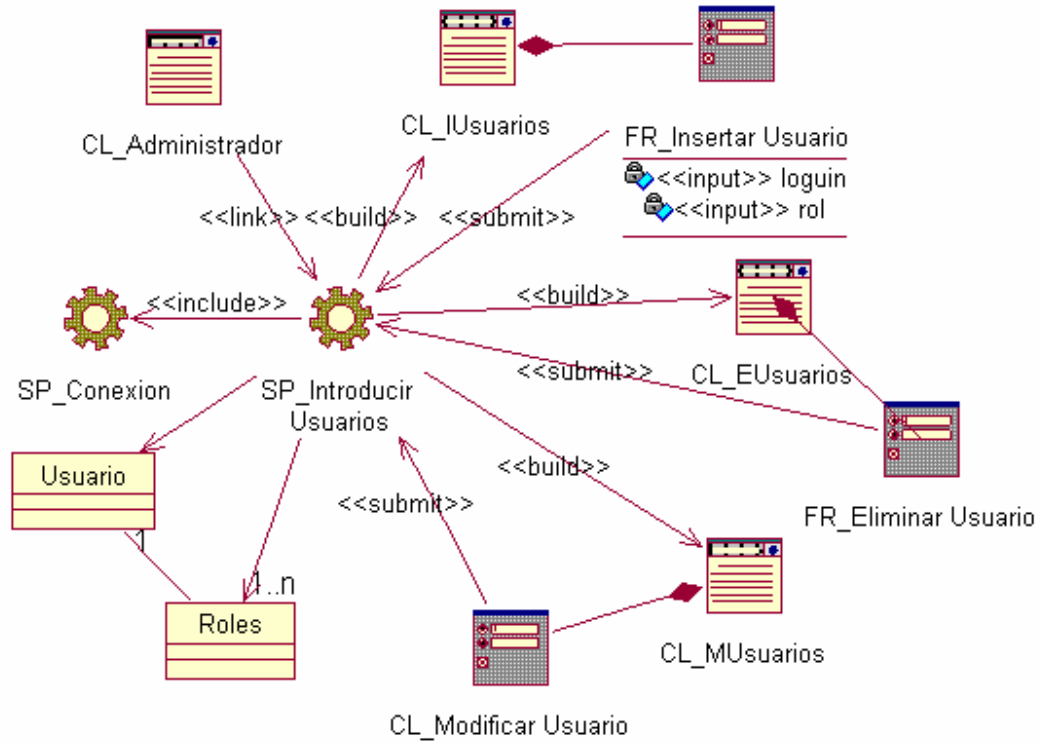


Anexo3 Diseño de Clases Web.

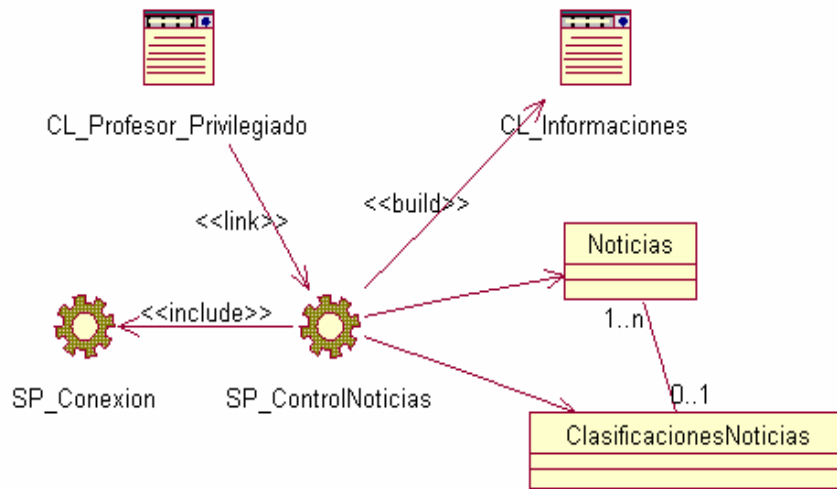
Anexo3.1 Autenticar usuario



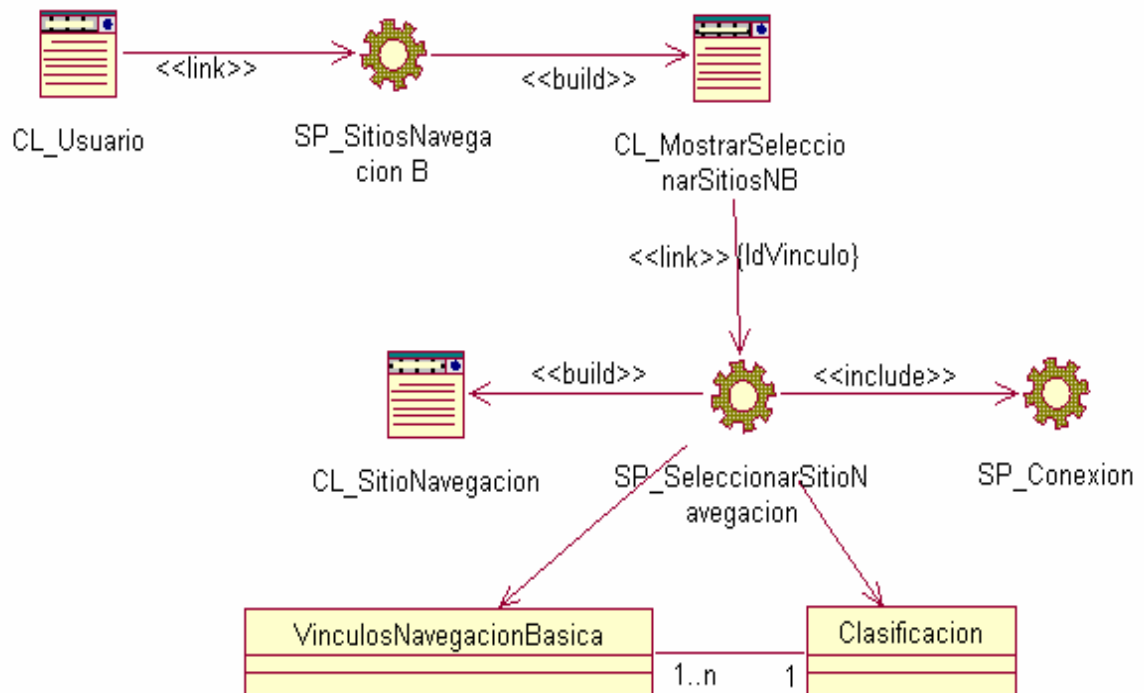
Anexo3.2 Introducir usuarios con roles



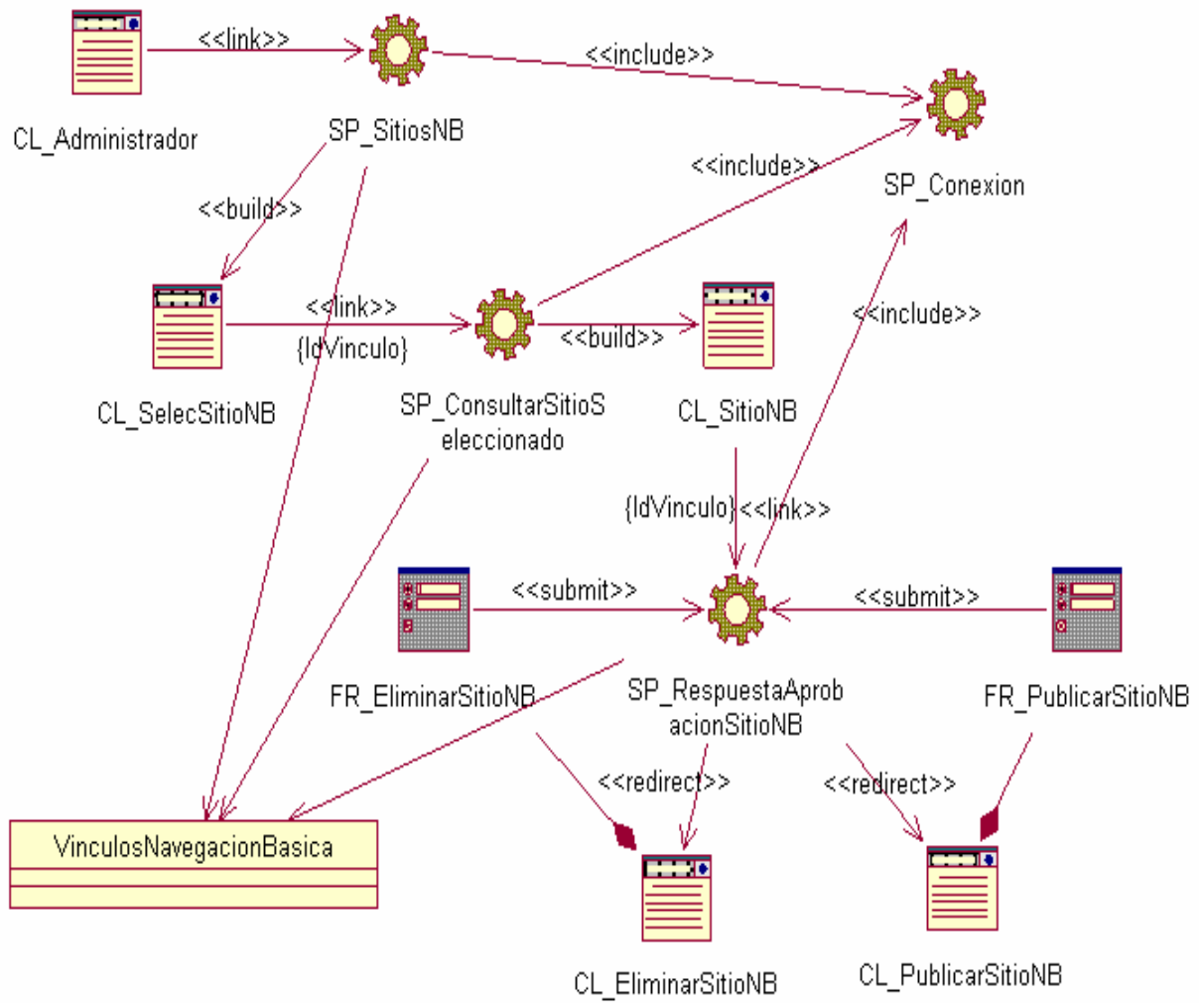
Anexo3.3 Acceder a Informaciones Específicas



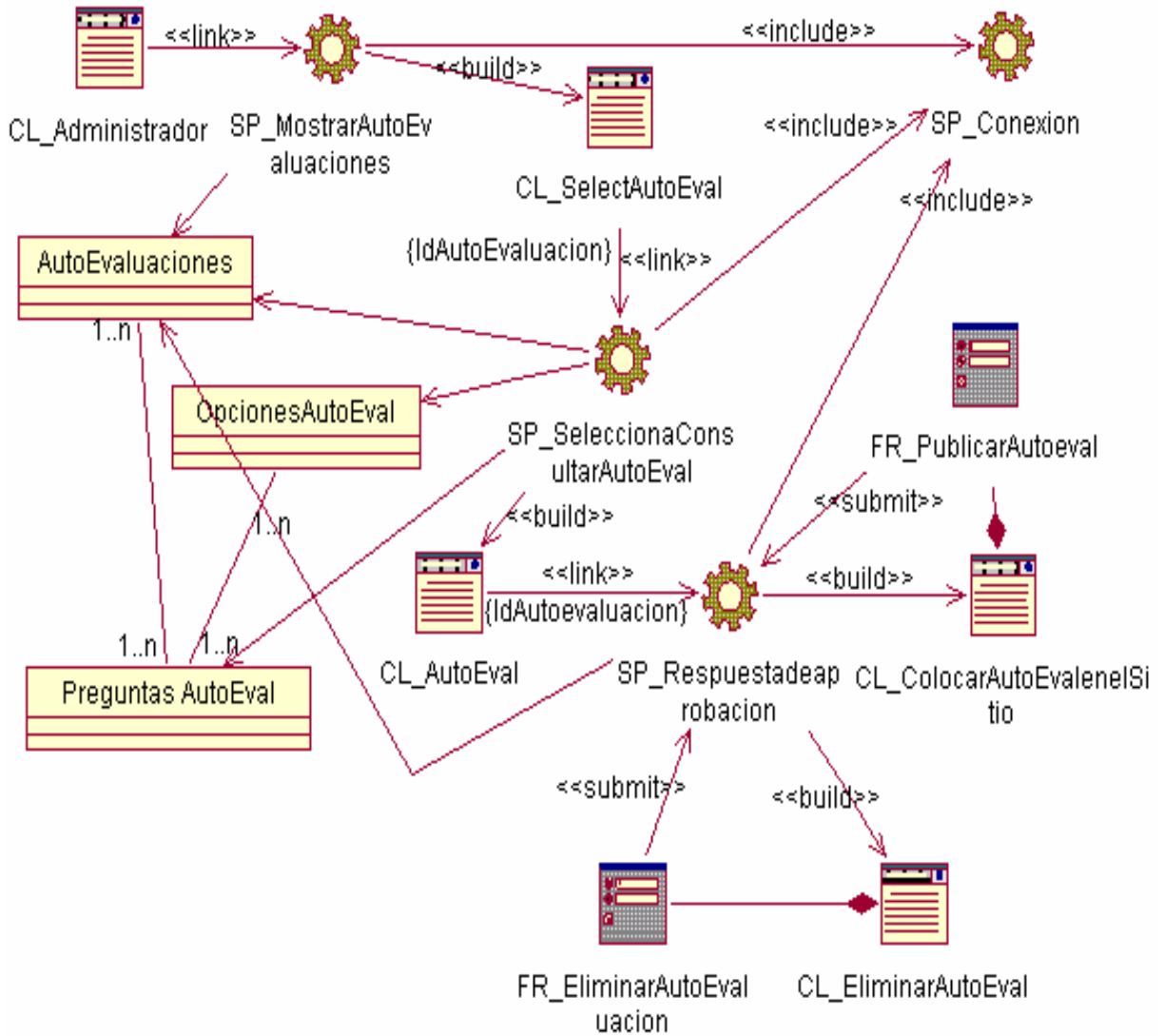
Anexo3.4 Acceder a sitios de Navegación Básica



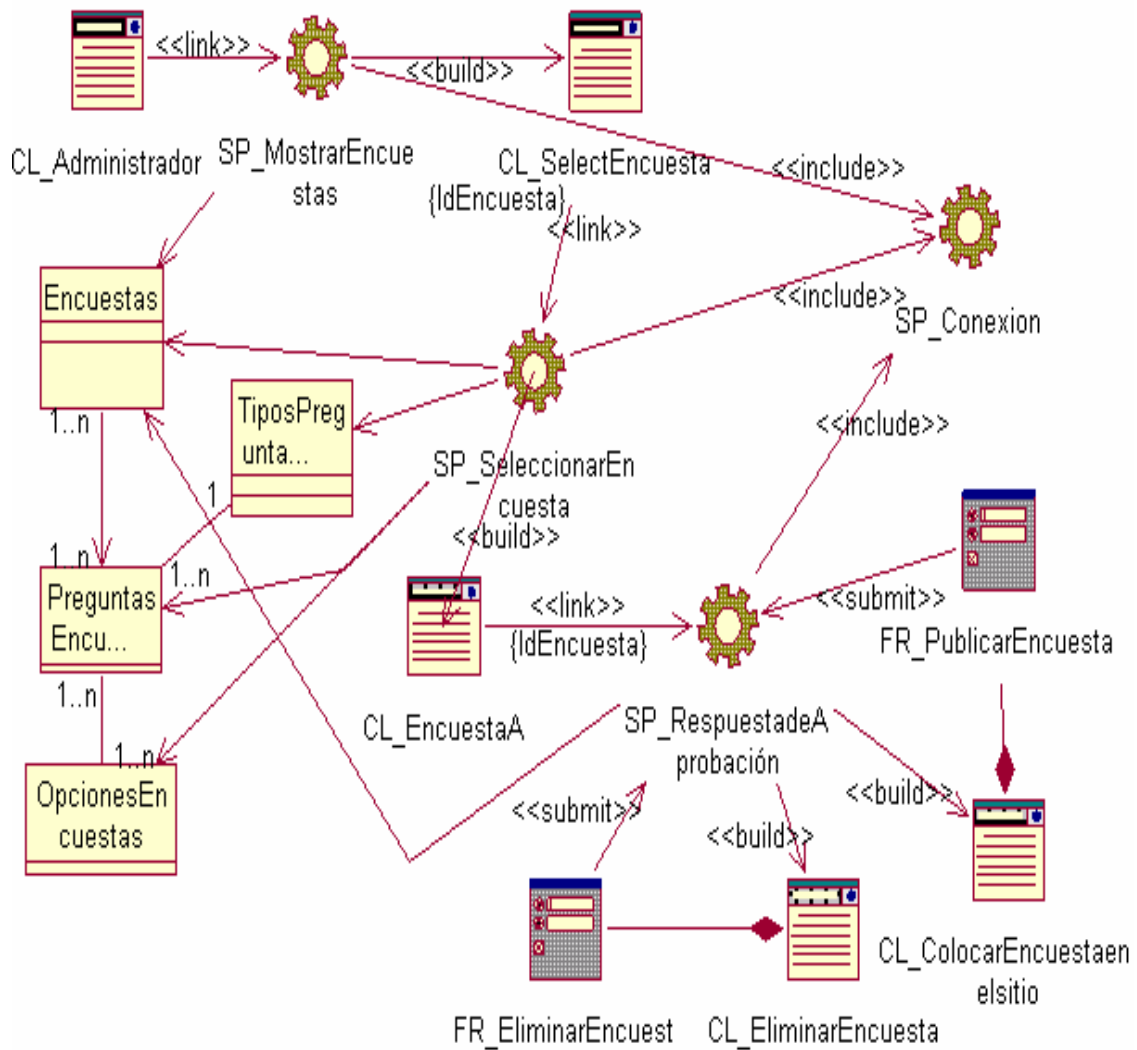
Anexo3.5 Aprobar publicar sitios de Navegación Básica



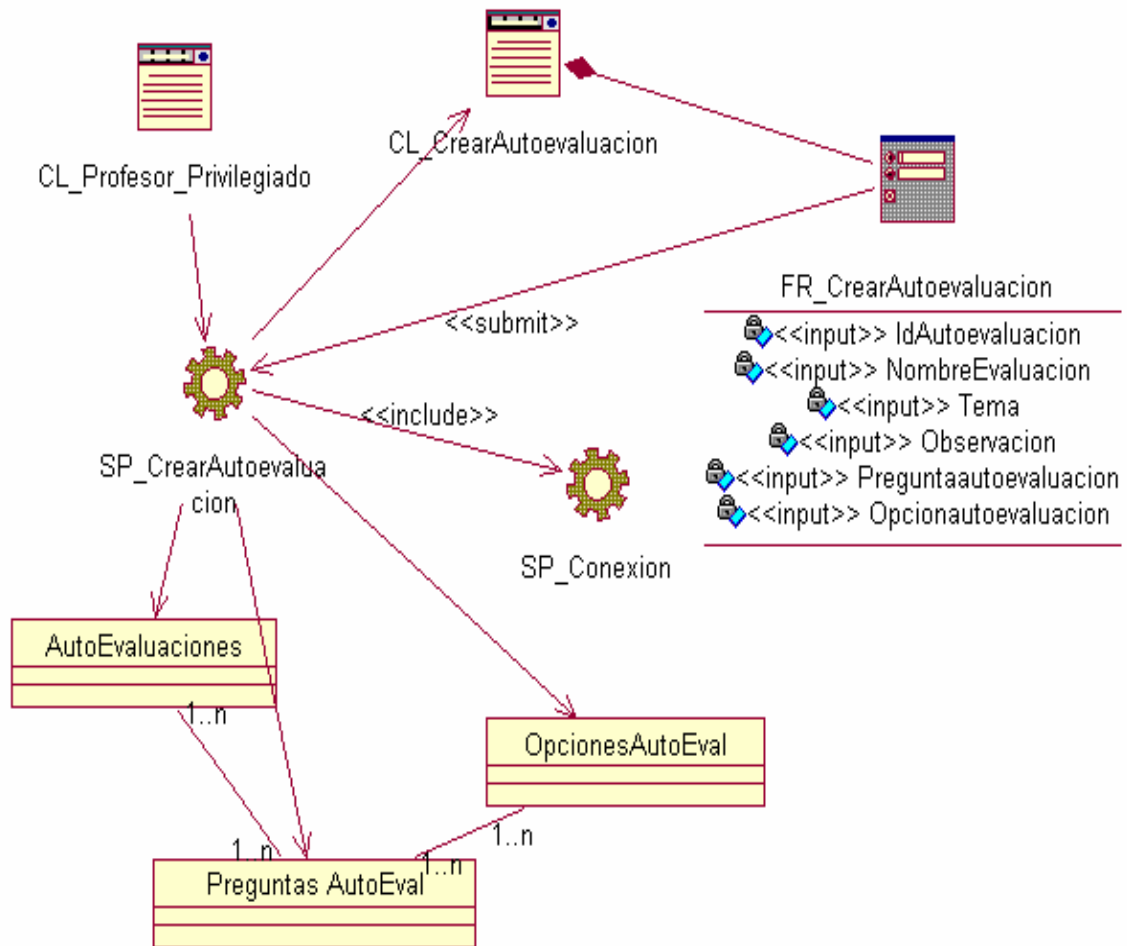
Anexo3.6 Aprobar solicitud de Auto-Evaluación



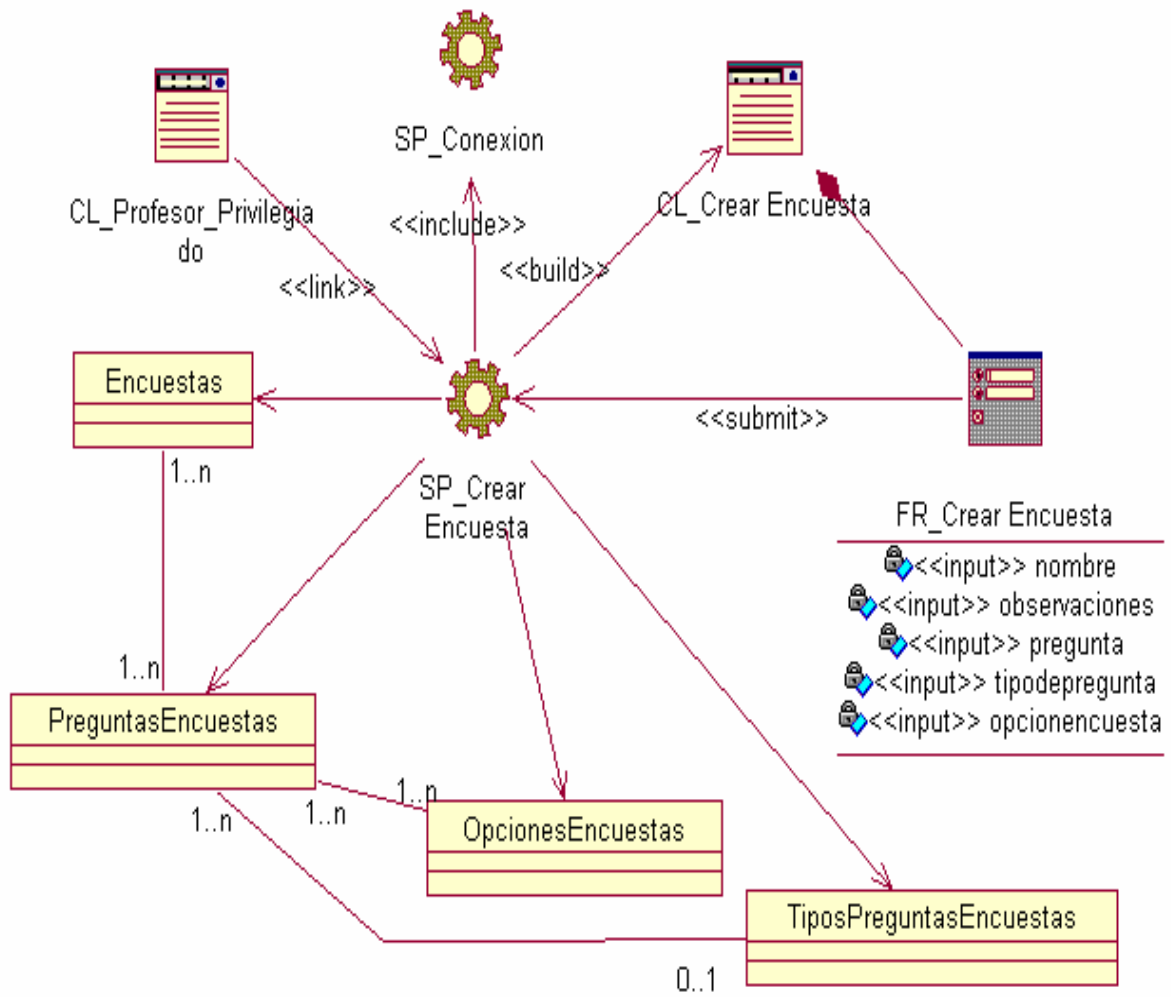
Anexo3.7 Aprobar solicitud de Encuesta



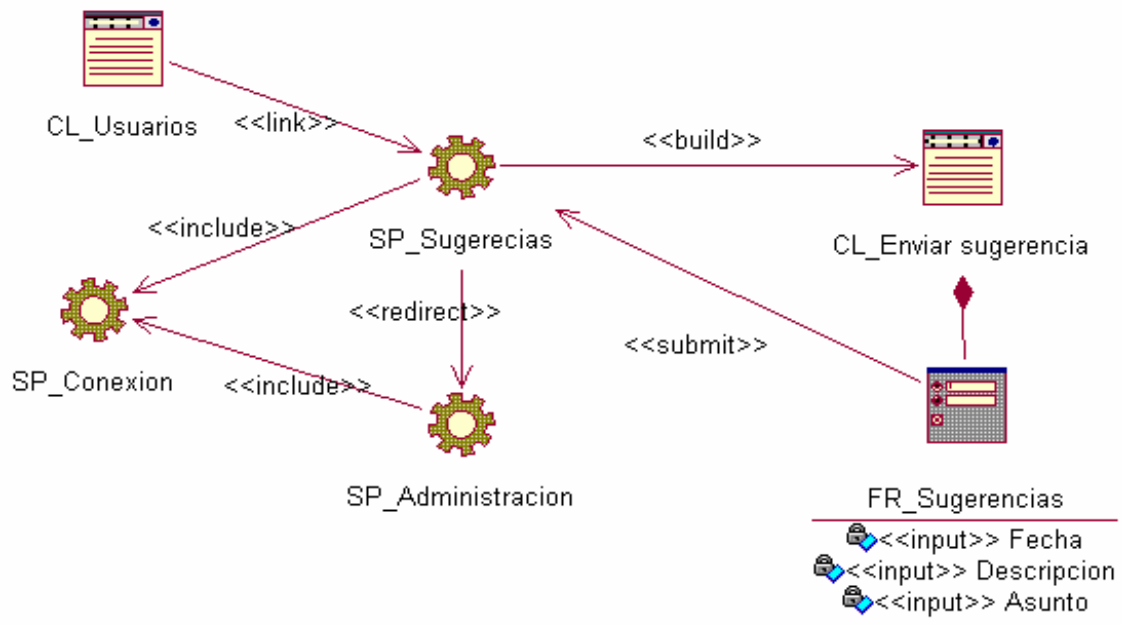
Anexo3.8 Crear Auto-Evaluación



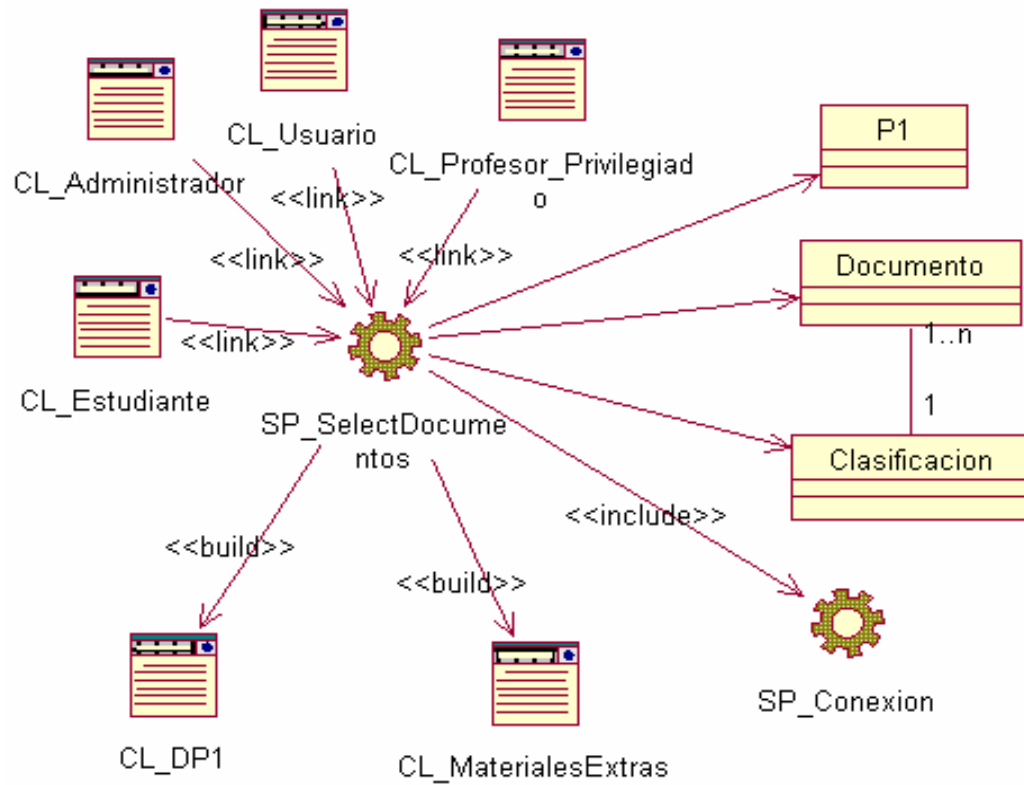
Anexo 3.9 Crear Encuesta



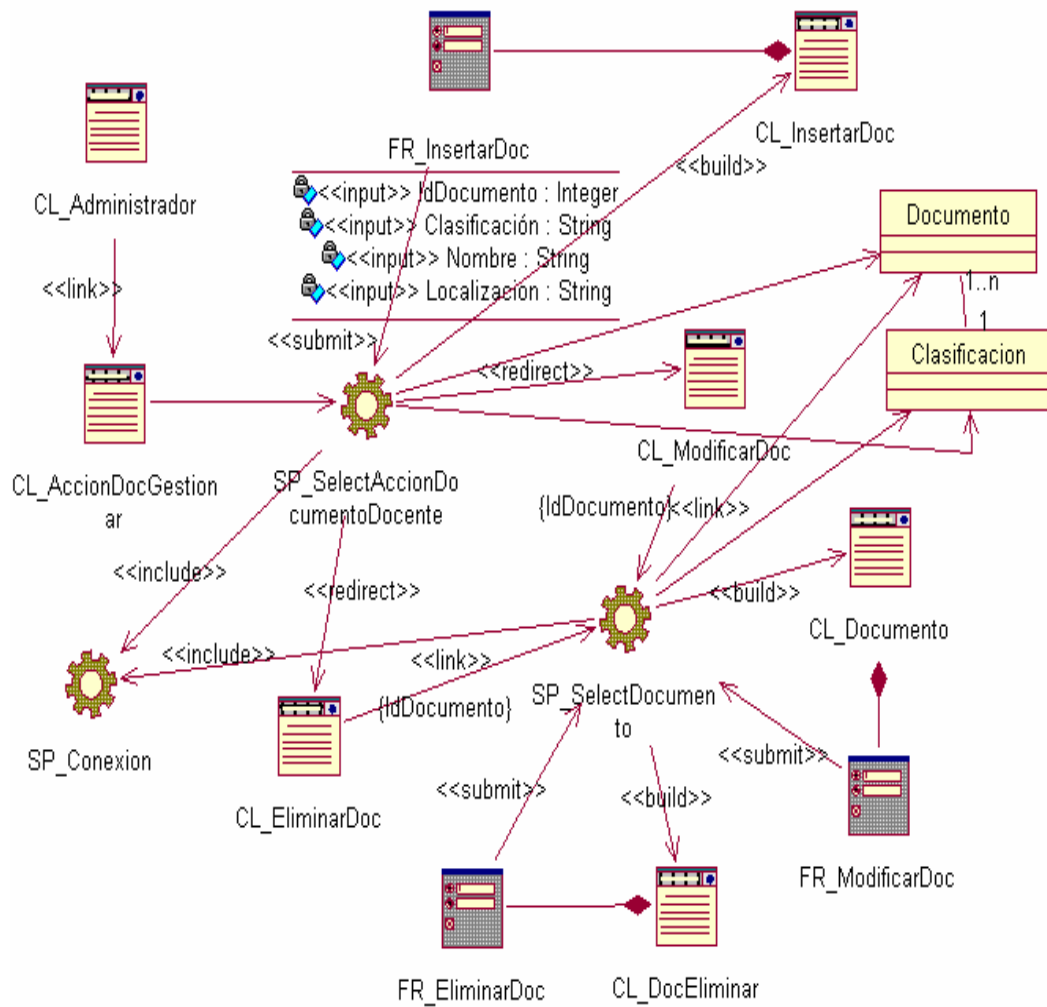
Anexo 3.10 Enviar sugerencias al administrador



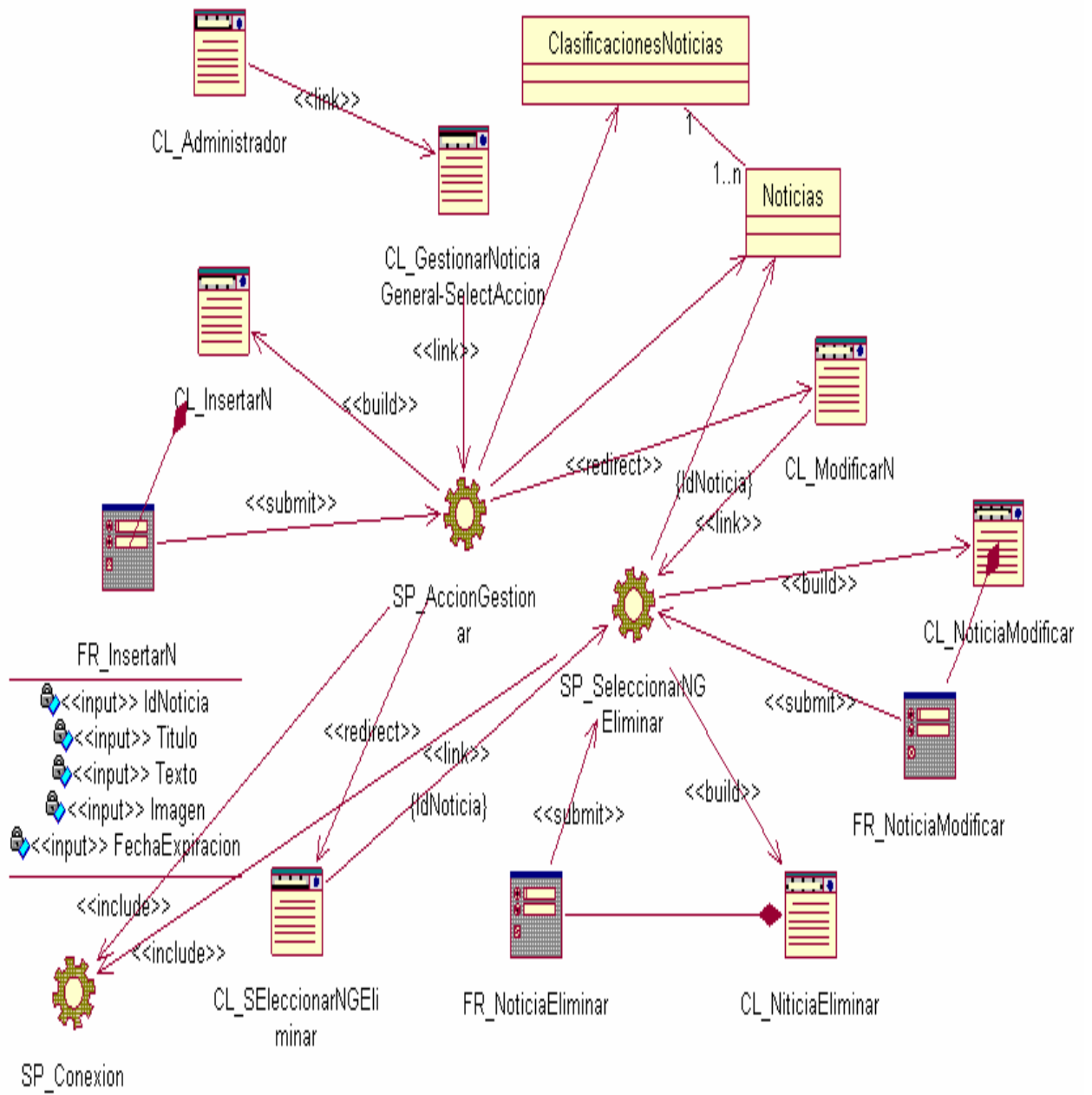
Anexo 3.11 Manipular Documentos Docentes



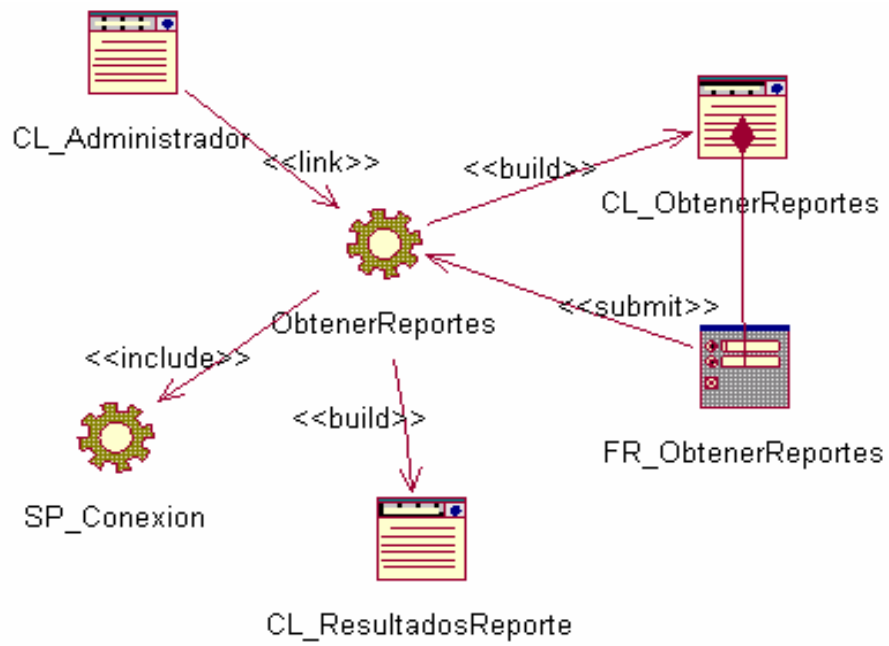
Anexo 3.12 Gestionar Documentos Docentes



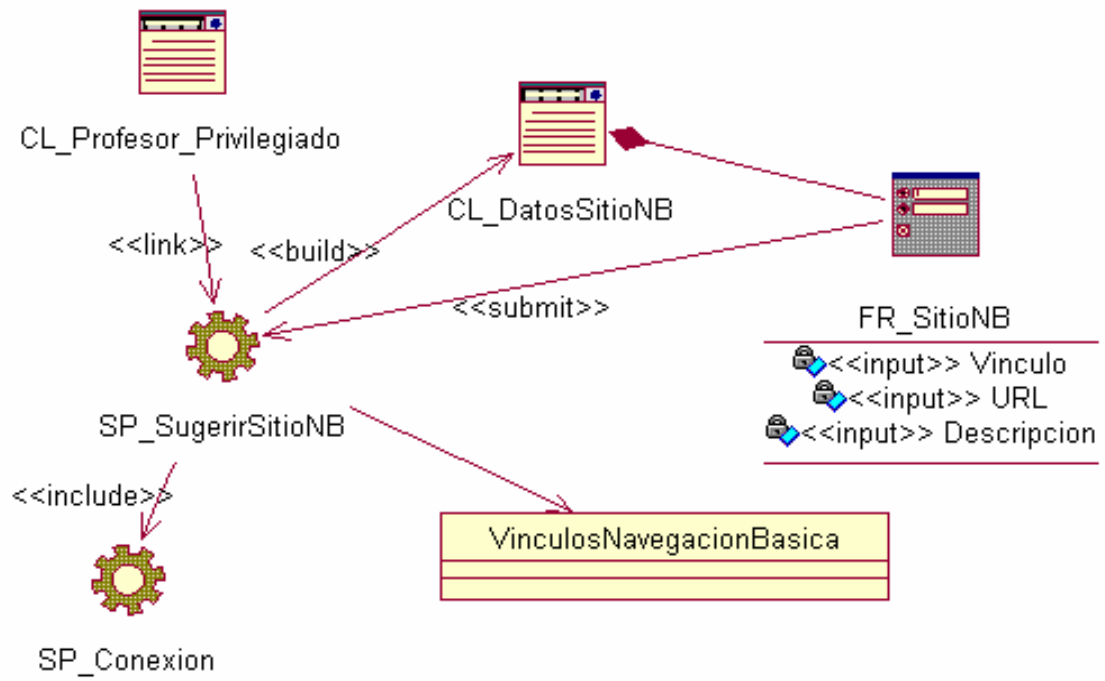
Anexo 3.13 Gestionar Noticias



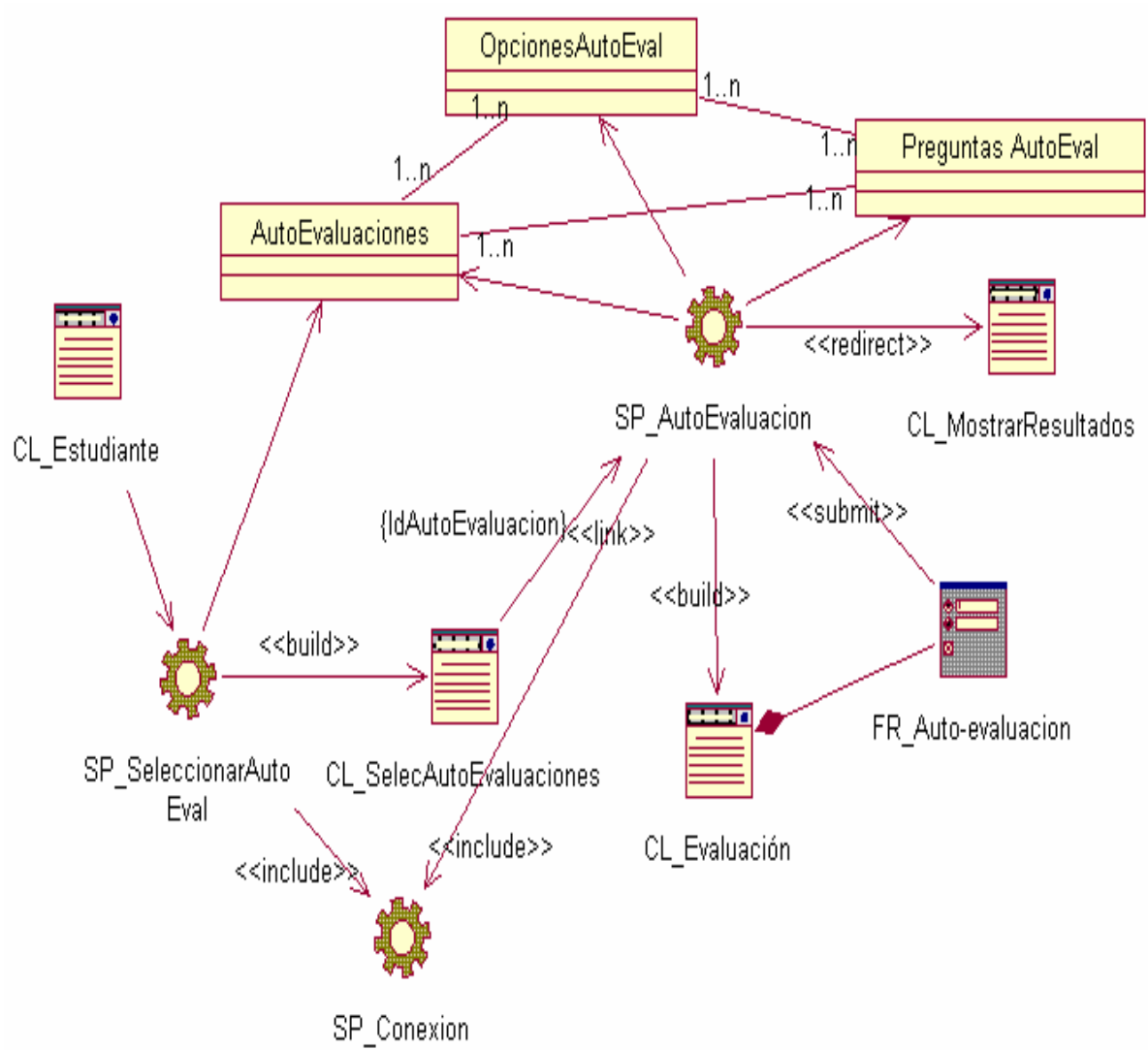
Anexo 3.14 Obtener Reportes



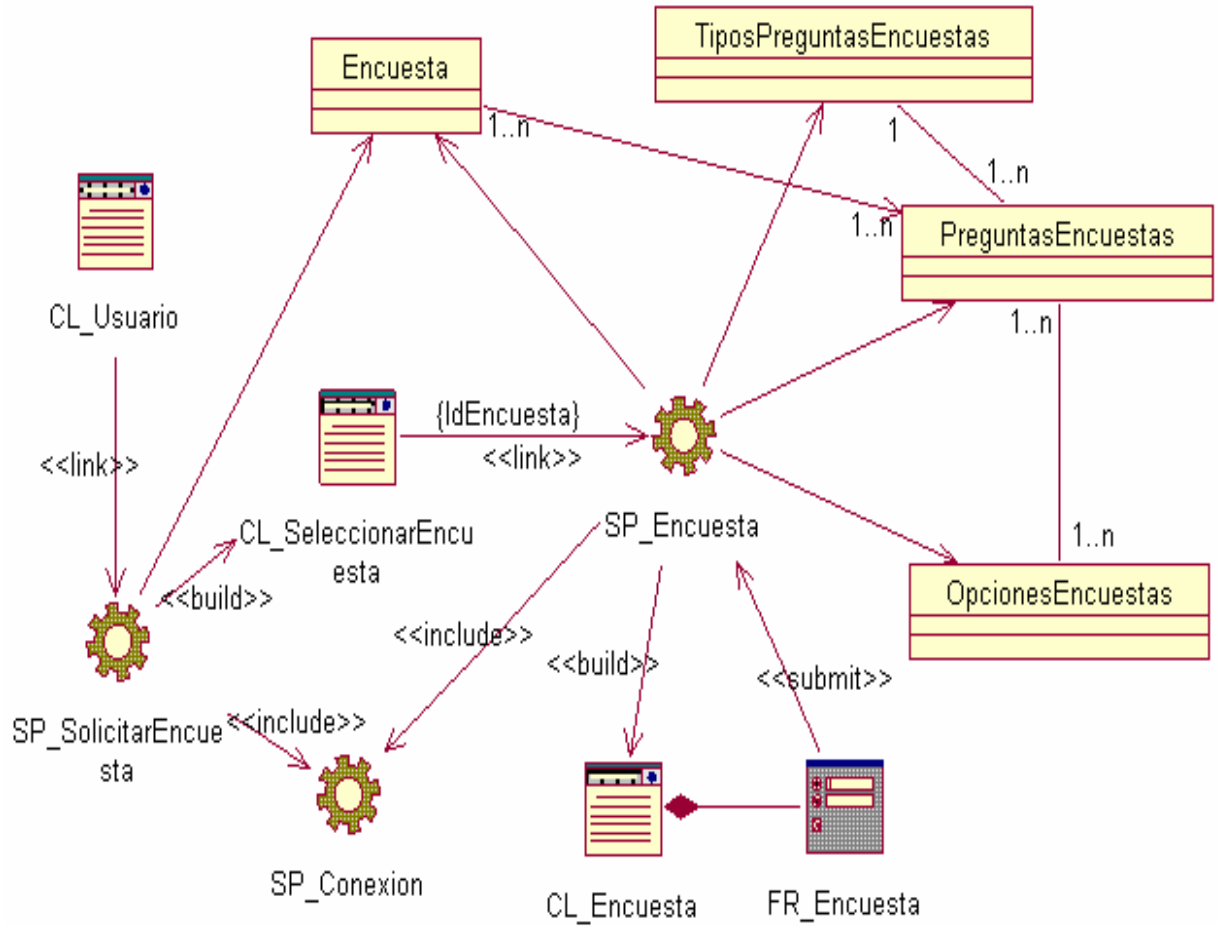
Anexo 3.15 Sugerir Sitios de Navegación Básica



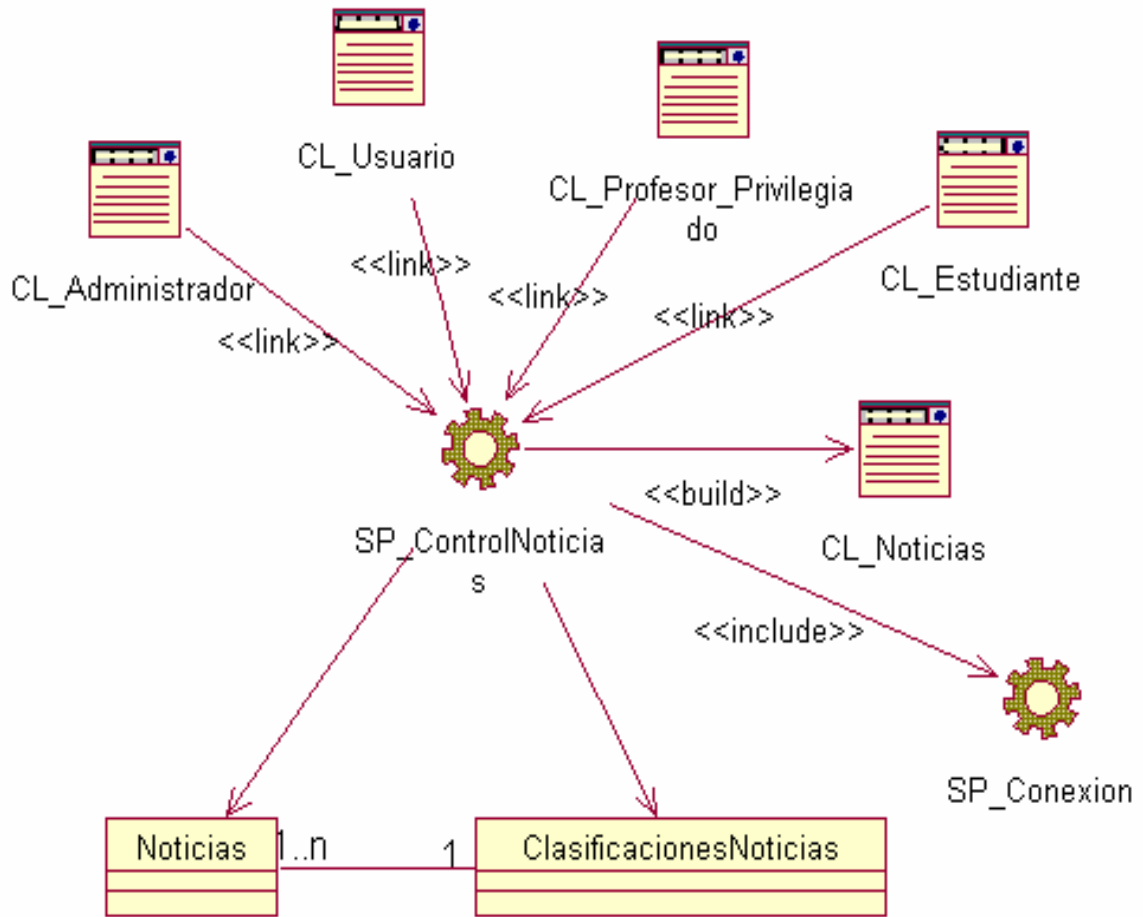
Anexo 3.16 Solicitar Auto-Evaluación



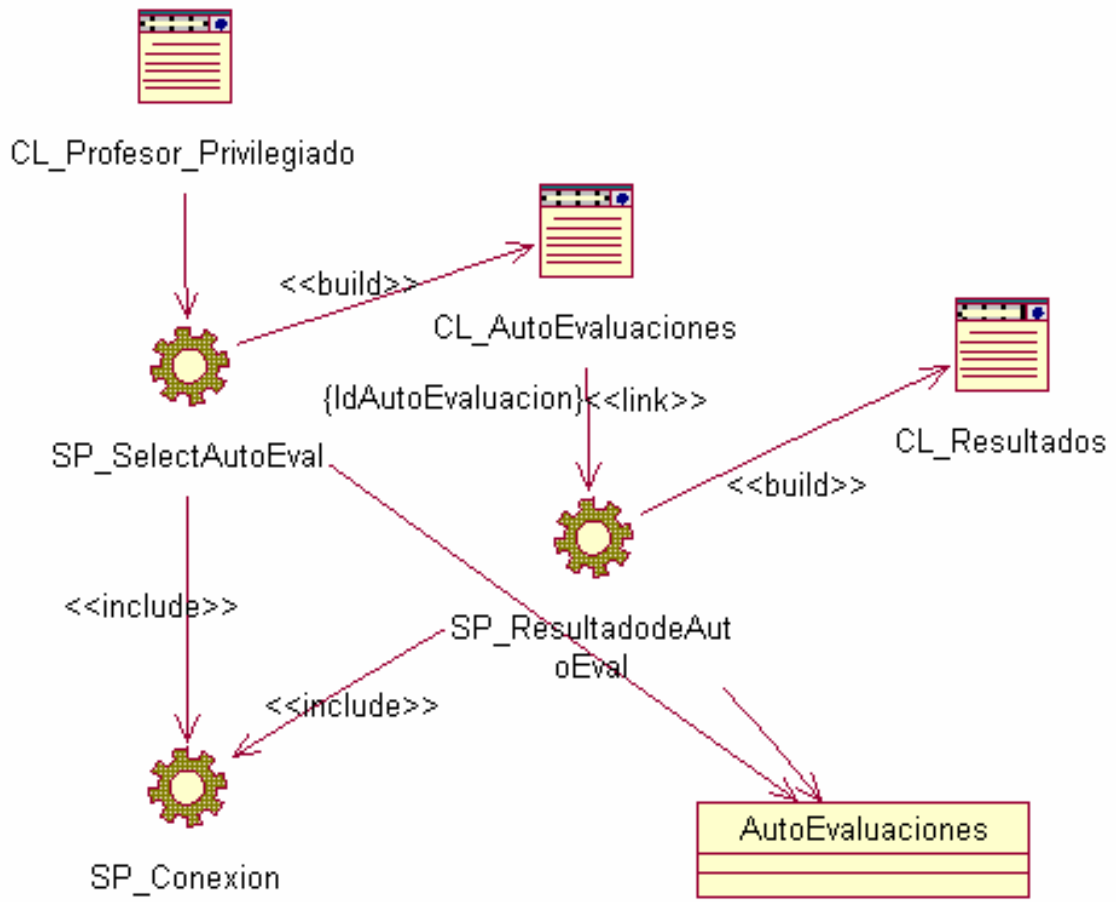
Anexo 3.17 Solicitar Encuesta



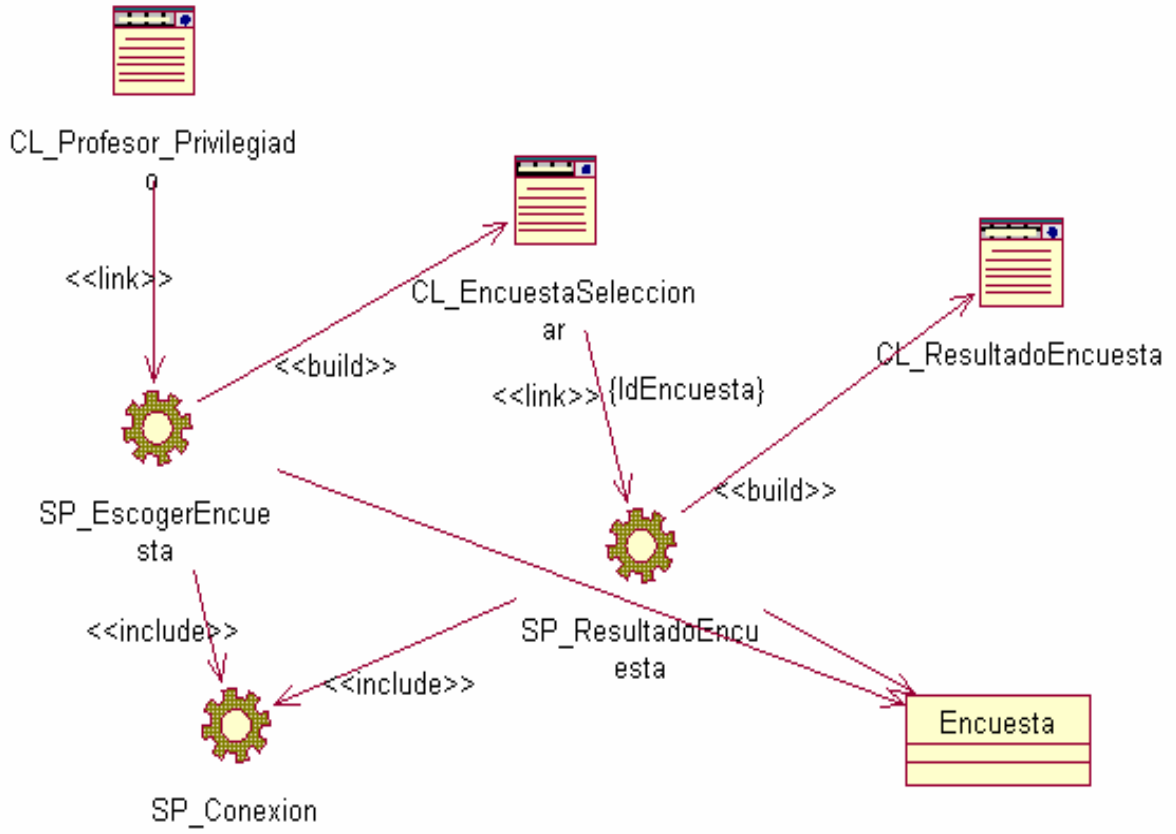
Anexo 3.18 Ver Noticias Generales



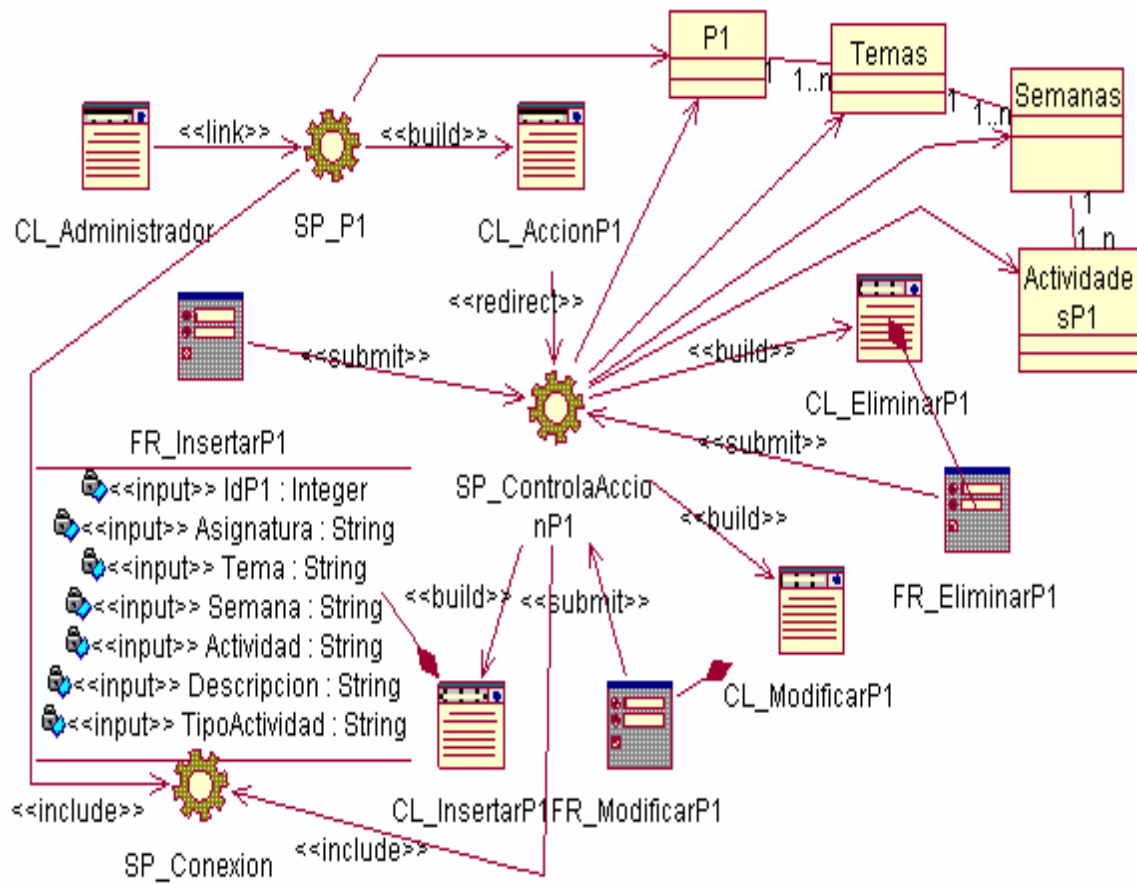
Anexo 3.19 Ver Resultado Auto-Evaluaciones



Anexo 3.20 Ver Resultado de Encuestas



Anexo 3.21 Gestionar P1



Anexo 4 Modelo de Datos

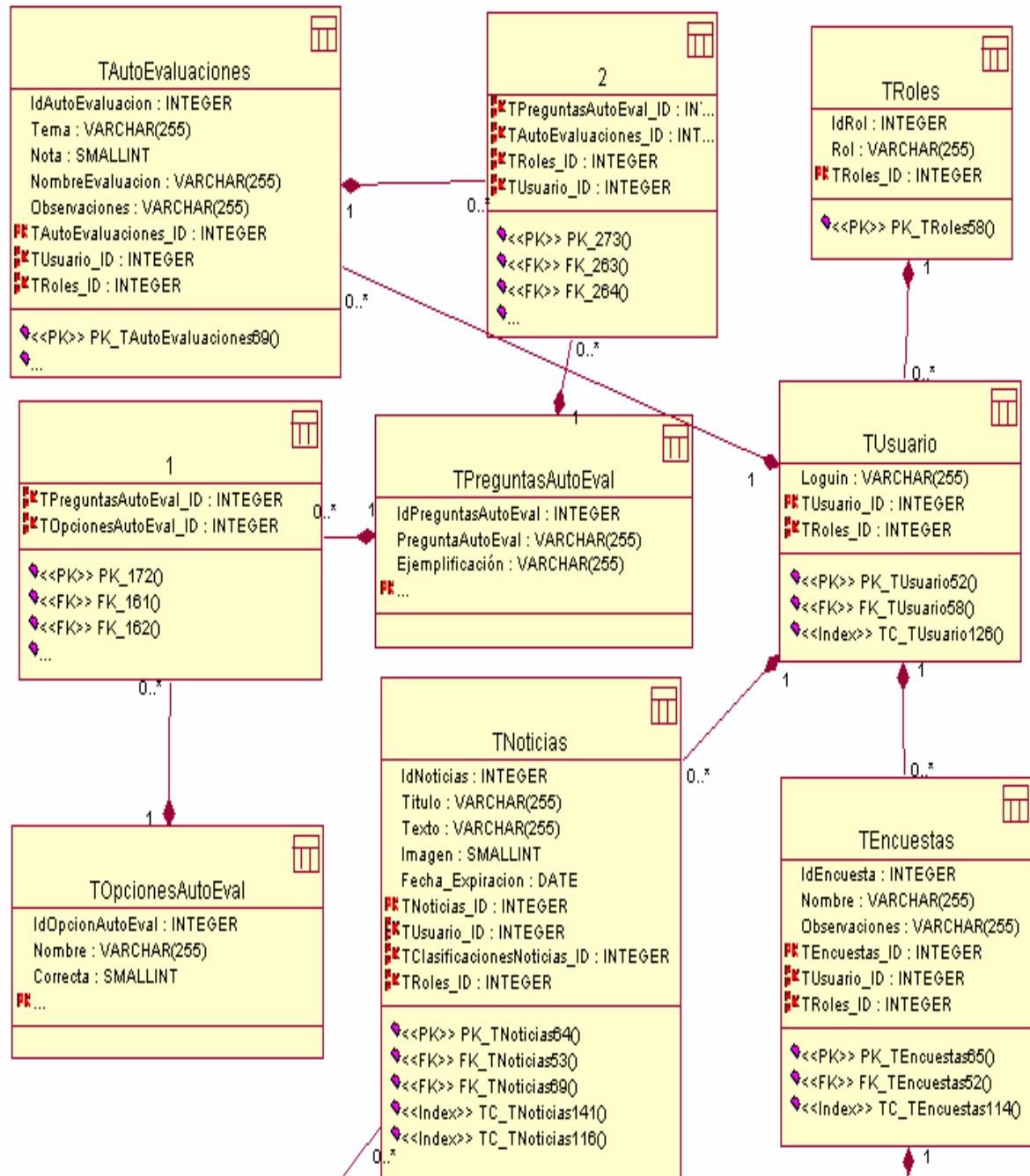


Figura 4.1

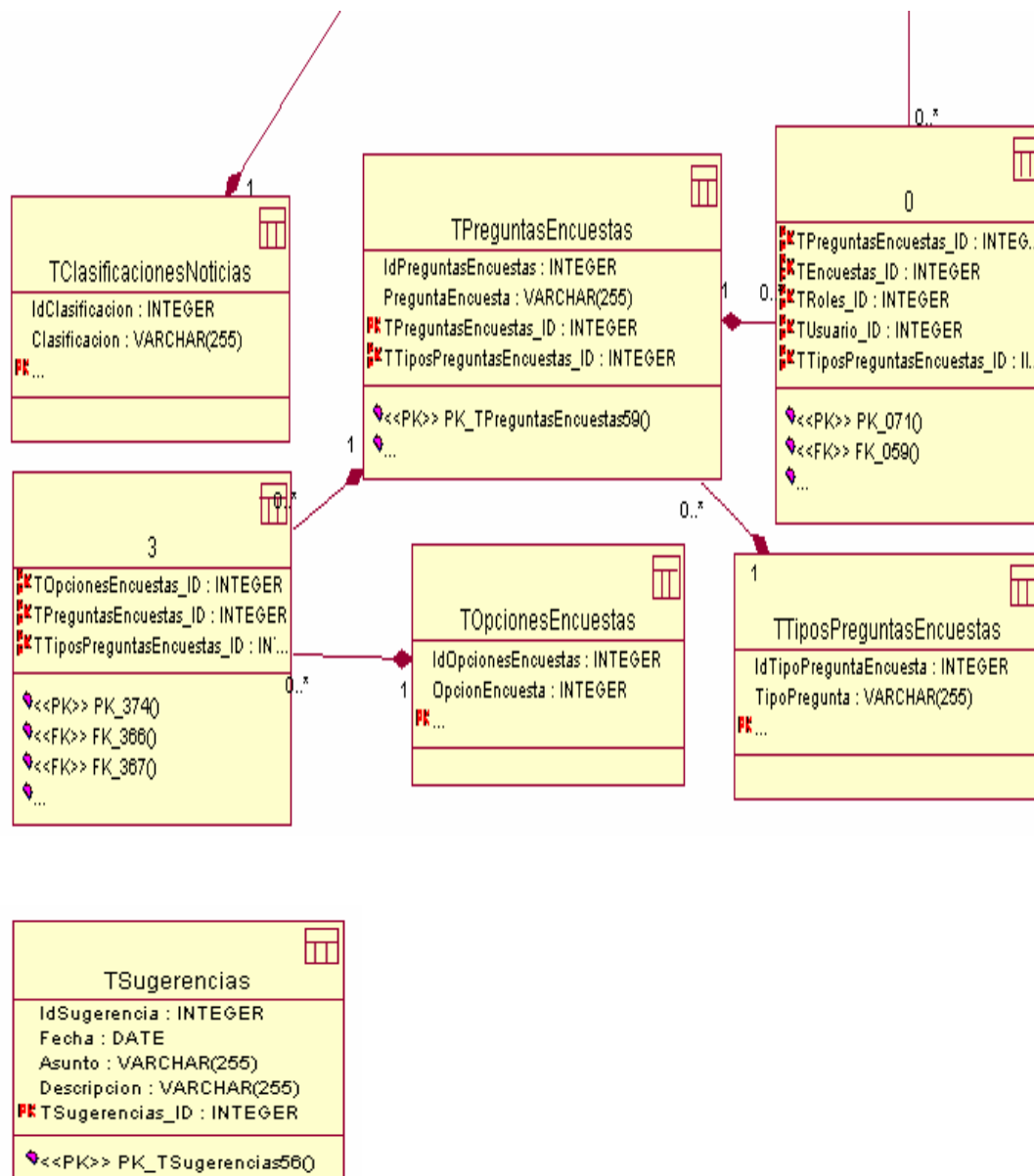


Figura 4.2

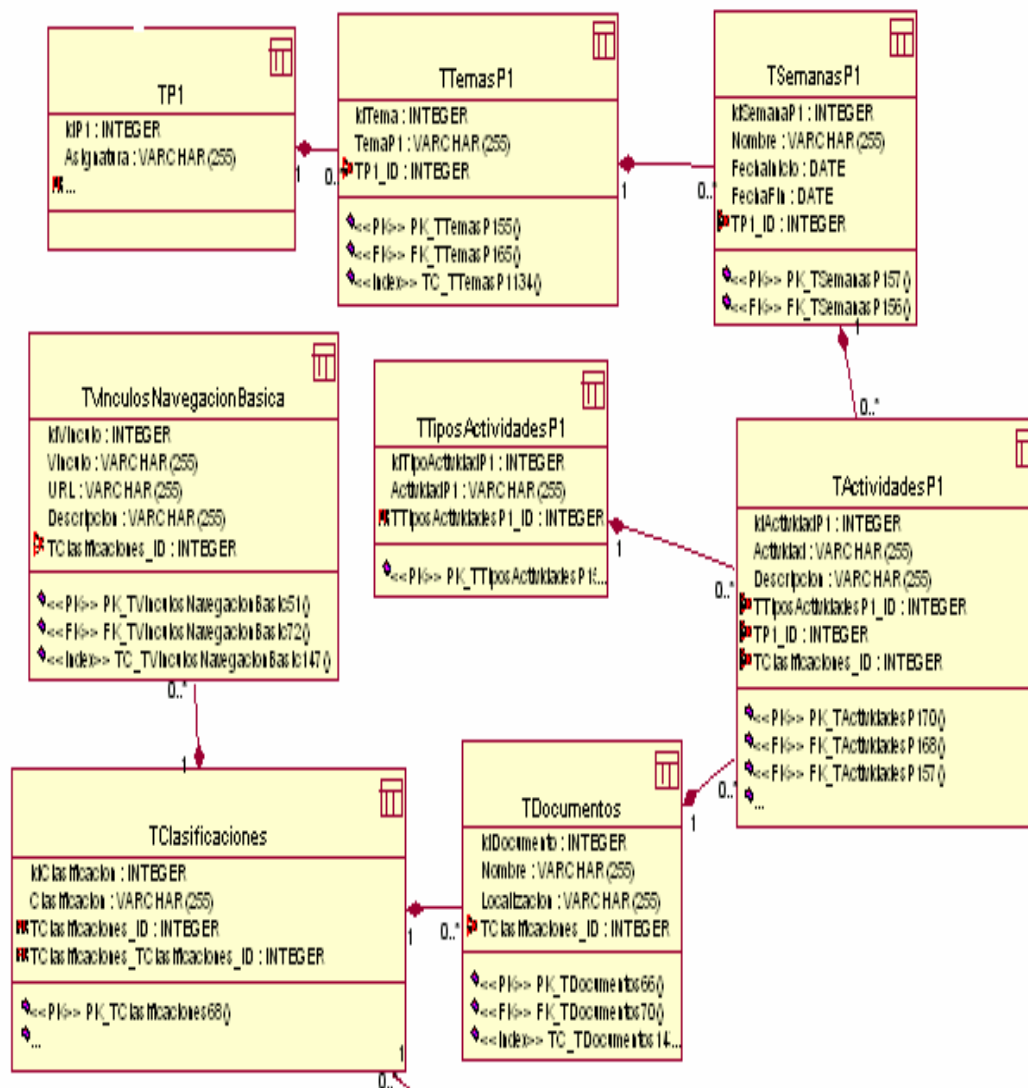


Figura 4.3