



Universidad de las Ciencias
Informáticas

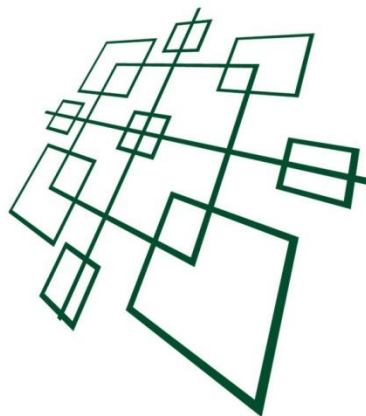
Dirección de Informatización



ANÁLISIS Y DISEÑO DEL MÓDULO REGISTRO Y CONTROL DOCENTE PARA AKADEMOS V2.0

Trabajo de diploma para optar por el título de:

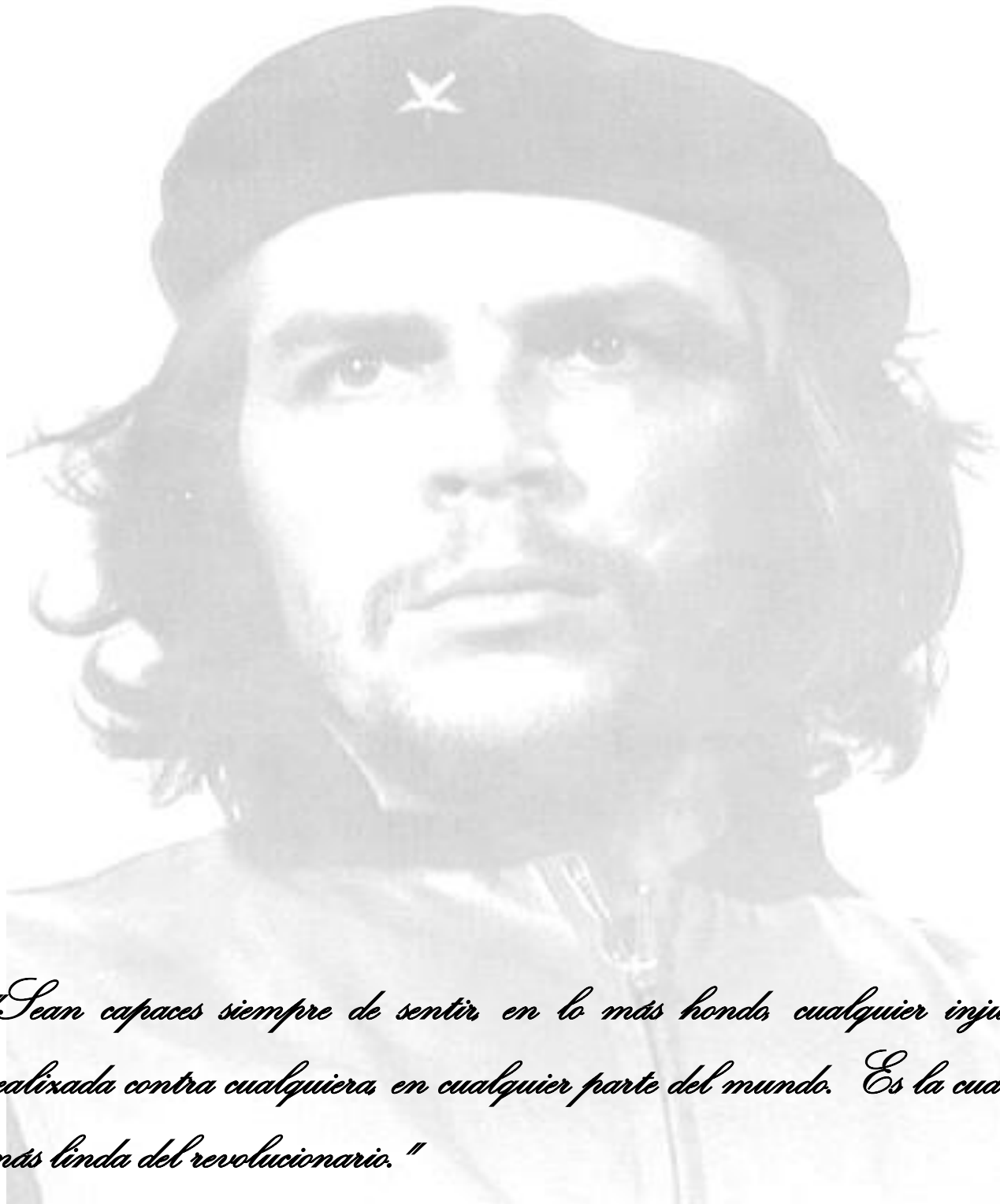
Ingeniero en Ciencias Informáticas



**Autores: Acralys Ferriol Ortiz
Enmanuel Azahares Reyes**

**Tutores: Ing. Geidis Sánchez Michel
Ing. Damián Cervantes Rodón**

Ciudad de La Habana, Junio 2009



"Sean capaces siempre de sentir en lo más hondo cualquier injusticia realizada contra cualquiera, en cualquier parte del mundo. Es la cualidad más linda del revolucionario."

Che

Declaración de autoría

Declaramos que somos los únicos autores del trabajo titulado: Análisis y Diseño del Módulo Registro y Control Docente para Akademos v2.0 y autorizamos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmamos la presente a los ____ días del mes de junio del año 2009.

Acralys Ferriol Ortiz

Firma del Autor

Geidis Sánchez Michel

Firma de la Tutora

Enmanuel Azahares Reyes

Firma del Autor

Damián Cervantes Rodón

Firma del Tutor

Opinión del tutor del trabajo de diploma

Título: Análisis y Diseño del Módulo Registro y Control Docente para Akademos v2.0.

Autores: Acralys Ferriol Ortiz

Enmanuel Azahares Reyes

Los tutores del presente Trabajo de Diploma consideran que durante su ejecución los estudiantes mostraron las cualidades que a continuación se detallan.

Por todo lo anteriormente expresado, consideramos que los estudiantes están aptos para ejercer como Ingenieros en Ciencias Informáticas; y proponemos que se le otorgue al Trabajo de Diploma la calificación de __ puntos.

Geidis Sánchez Michel

Firma

_____ días del mes de Junio del año 2009.

Damián Cervantes Rodón

Firma

_____ días del mes de Junio del año 2009.

A:

mí Papá, mí Mamá,

mí Hermana y mí Familia.

Acrafys

A:

mí Papá, mí Mamá,

mí hermano, mí hermana y mí Familia.

Enmanuel

Agradezco a:

Mi papá, por ser un ejemplo a seguir, por haberme enseñado que la vida es sacrificio y esfuerzo y sobre todo por ser el padre que es.

Mi mamá, por su dedicación e infinito amor y por haber soportado todas mis malcriadeces.

Mi hermana te quiero infinitamente...

Andry, por darme su amor y compartir nuevos e inolvidables momentos de mi vida, acuérdate de mí, si Abril te llega...

Mi abuelo José, mis tíos, mis tías, mis primos y a toda la familia en general, por todo el apoyo que siempre me brindaron, por hacerme sentir tan especial para ustedes.

Mis abuelas Ana y Zenaida, porque sé que estarían orgullosas de mí y mi abuelo Níco también.

Yoya, Normi, Sandra, Meli (de chocolate), Carlín, Tania, Ferni, Pedro, por estar siempre al tanto, tan cerca de mí.

Mis tutores, Damián y Geidis, por toda la confianza y ayuda que me brindaron para que este trabajo resultara lo mejor posible.

Ani, Mora y Pepito por acogerme con tanto cariño y amor, muchas gracias.

Clarisa, Adrianys y Alemán, por su apoyo y preocupación.

Yohana, por estar siempre dispuesta a ayudarme y escucharme tantas veces...

Geykel, mi amigo, por su ayuda incondicional y su apoyo en todo momento.

Darien (Pepe), Frank, Grisel, My Friend (...), Dianly (El Loco), Yaniris (La Tata), Joe, Norges, Robe, Rosmel, Yisel, por los años que compartimos juntos.

Todo aquel que de una forma u otra hizo posible la realización de este sueño.

Acraľys

Agradezco a:

Nuestro Comandante en Jefe, Fidel Castro Ruz, por haberme brindado la posibilidad de estar en una escuela que trabaja en pos de ser una universidad de excelencia.

Los buenos profesores que me han dado clases, José Albert, Joel, Sergio y otros, con los cuales he aprendido mucho no sólo profesional sino espiritualmente.

Mis compañeros de aula y de mi año, con los cuales he pasado muchas vicisitudes.

Mis dos tutores, Damián y Geidis. A Geidis por su disposición y ayuda en todo momento y a Damián por sus revisiones constantes y periódicas a todo el trabajo.

Mi compañera de tesis, Acralys, por su comprensión, preocupación y por haber contribuido al entendimiento entre nosotros, para poder realizar con eficiencia y curiosidad nuestro trabajo de diploma. Empezamos un día a trabajar en la tesis y ahora, finalmente, compartimos la alegría de haber hecho el último trabajo para ser ingenieros.

Mi novia Lenia Martinez Noa que ha sido mi apoyo en estos 2 años, 9 meses y 8 días que cumplimos hoy. Ella ha sido mi apoyo en los momentos difíciles, mi alegría en los momentos íntimos y verdaderamente ha sido mi amiga y mi amor.

Mi hermano Raúl, mi hermana Irina y a toda mi familia por haberme apoyado en mis estudios.

Mi papá Manuel, por haber sido mi ejemplo de determinación y responsabilidad en todo momento, por haberme enseñado cómo aprender a vivir, conocer a las personas, escuchar primero y hablar después. Me estás viendo a mí, estás viendo a mi padre. El hecho de ser como soy se lo debo a él.

Mi mamá Nidia, que ha sido mi guía y mi alma. Realmente me quedo sin palabras. Su ayuda incondicional, sacrificio constante y educación en todo momento, me servirá siempre para vivir toda mi vida, seguro de mí mismo. En cada paso que doy, está mi madre, en cada problema que resuelvo, está su pensamiento y durante toda mi vida, mami estará conmigo.

Enmanuel

Resumen

La Universidad de las Ciencias Informáticas está llamada a ser, por sus características excepcionales, una universidad de excelencia y élite en el desarrollo de software y la informatización de la sociedad cubana. En la actualidad la UCI cuenta con el Sistema Automatizado para la Gestión Académica – “Akademos” que se encarga de optimizar los procesos de gestión académica de la universidad. Este sistema se encuentra actualmente en plena utilización, pero a pesar de las facilidades tecnológicas y de accesibilidad que ofrece, se han detectado algunas dificultades que atentan contra su buen funcionamiento e impiden que el mismo pueda ser utilizado por otras entidades educativas, por lo que en estos momentos se encuentra sometido al desarrollo de una nueva versión. En el presente trabajo se realizó el análisis y diseño del Módulo Registro y Control Docente para la nueva versión del sistema Akademos. La incorporación de este módulo permitirá gestionar toda la información relacionada con el desarrollo docente de los estudiantes y consultar un amplio y sensible universo de información referente al proceso docente educativo como evaluaciones y registro de asistencia. Para el desarrollo del trabajo se determinó utilizar por decisiones arquitectónicas de la dirección del proyecto PHP 5.2.8, HTML, AJAX, JavaScript y UML como lenguajes; como sistema gestor de bases de datos: PostgreSQL 8.3.5; como metodología de desarrollo: RUP; como servidor Web: Apache 2.2.9; como herramienta de modelado: Visual Paradigm 3.0 y como marcos de trabajo: Ext JS 2.2, Zend Framework 1.7.6 y Doctrine 1.1.

Índice

Introducción	1
Capítulo I: Fundamentación teórica	5
1.1 Introducción.....	5
1.2 Los sistemas de gestión académica	5
1.2.1 Los sistemas de gestión académica en el mundo	5
1.2.2 Los sistemas de gestión académica en Cuba	8
1.3 Software libre	14
1.3.1 Tendencias actuales del software libre.....	15
1.4 Herramientas, lenguajes y metodologías a utilizar.....	16
1.4.1 Aplicaciones Web	16
1.4.2 Modelo cliente-servidor	16
1.4.3 Servicios Web	19
1.4.3.1 Servidor Web Apache.....	19
1.4.4 Lenguajes de programación para la Web.....	21
1.4.4.1 Lenguajes utilizados del lado del cliente.....	21
1.4.4.2 Lenguajes utilizados del lado del servidor.....	22
1.4.5 Metodologías de desarrollo del software	23
1.4.6 Lenguaje Unificado de Modelado	25
1.4.7 Herramientas de modelado de software.....	26
1.5 Sistemas para la gestión de base de datos	27
1.6 Marcos de trabajo.....	28
1.7 Entornos integrados de desarrollo	31
1.8 Fundamentación de la metodología y herramienta de modelado a utilizar.....	33
1.9 Fundamentación de las tecnologías, lenguajes y herramientas a utilizar	34
1.10 Conclusiones.....	35
Capítulo II: Descripción de la solución propuesta.....	36
2.1 Introducción.....	36
2.2 Descripción de los procesos vinculados al negocio	36

2.3 Descripción de las reglas del negocio.....	39
2.4 Modelo del negocio	40
2.4.1 Actores del negocio.....	40
2.4.2 Trabajadores del negocio.....	41
2.4.3 Diagrama de los de casos de uso del negocio	41
2.4.4 Descripciones textuales de los casos de uso del negocio	42
2.5 Modelado del sistema.....	49
2.5.1 Requerimientos funcionales	49
2.5.2 Requerimientos no funcionales	52
2.5.3 Actores del sistema.....	55
2.5.4 Diagrama de los casos de uso del sistema	56
2.5.5 Descripciones abreviadas de los casos de uso del sistema	60
2.6 Conclusiones.....	64
Capítulo III: Elaboración de la solución propuesta.....	66
3.1 Introducción.....	66
3.2 Estimación del esfuerzo.....	66
3.3 Análisis.....	72
3.3.1 Diagramas de clases del análisis	73
3.4 Diseño	80
3.4.1 Patrones de diseño	80
3.4.2 Diagrama de clases del diseño	83
3.5 Diseño de la base de datos	98
3.5.1 Modelo lógico.....	99
3.5.2 Modelo físico.....	100
3.6 Diagrama de despliegue.....	101
3.7 Concepción general de la ayuda	101
3.8 Tratamiento de errores	102
3.9 Conclusiones.....	102
Conclusiones	103

Recomendaciones	104
Bibliografía referenciada	105
Bibliografía consultada.....	106
Glosario de términos.....	108
Anexos.....	111

Índice de figuras

Figura 1. Modelo cliente-servidor	17
Figura 2. Solicitud de servicios utilizando modelo cliente-servidor	18
Figura 3. Análisis realizado por Netcraft de la explotación de servidores Web.....	20
Figura 4. Esquema de procesos del Módulo Registro y Control Docente.....	37
Figura 5. Diagrama de los casos de uso del negocio	42
Figura 6. Catálogo de actores del sistema	57
Figura 7. Diagrama de paquetes de los casos de uso del sistema.....	57
Figura 8. Diagrama de casos de uso. Paquete “Gestionar Información”	58
Figura 9. Diagrama de casos de uso. Paquete “Generar Información”.....	58
Figura 10. Diagrama de casos de uso. Paquete “Consultar Información”.....	59
Figura 11. Vista de casos de uso arquitectónicamente significativos	60
Figura 12. Diagrama de clases del análisis: CUS “Solicitar Asignatura Adelanto”	73
Figura 13. Diagrama de clases del análisis: CUS “Aprobar_Rechazar Solicitud de Asignatura”	73
Figura 14. Diagrama de clases del análisis: CUS “Asignar Profesor a Grupo Docente”	74
Figura 15. Diagrama de clases del análisis: CUS “Desubicar Profesor de Grupo Docente”	74
Figura 16. Diagrama de clases del análisis: CUS “Consultar Grupo Docente”	75
Figura 17. Diagrama de clases del análisis: CUS “Gestionar Crédito Bonificación”	75
Figura 18. Diagrama de clases del análisis: CUS “Gestionar Crédito Premio”	76
Figura 19. Diagrama de clases del análisis: CUS “Gestionar Crédito por Actividades”	76
Figura 20. Diagrama de clases del análisis: CUS “Gestionar Control Asistencia”	77
Figura 21. Diagrama de clases del análisis: CUS “Generar Acta de Examen Final”	77
Figura 22. Diagrama de clases del análisis: CUS “Gestionar Grupo Docente”	78

Figura 23. Diagrama de clases del análisis: CUS “Mostrar Registro de Asistencia”	78
Figura 24. Diagrama de clases del análisis: CUS “Gestionar Registro Evaluación”	79
Figura 25. Diagrama de clases del análisis: CUS “Gestionar Matrícula en Grupo Docente”.....	79
Figura 26. Diagrama de clases del análisis: CUS “Mostrar Registro de Evaluación”	80
Figura 27. Diagrama de clases Web: CUS “Solicitar Asignatura Adelanto”	84
Figura 28. Diagrama de clases Web: CUS “Aprobar_Rechazar Solicitud de Asignatura”.....	85
Figura 29. Diagrama de clases Web: CUS “Asignar Profesor a Grupo Docente”	86
Figura 30. Diagrama de clases Web: CUS “Desubicar Profesor de Grupo Docente”	87
Figura 31. Diagrama de clases Web: CUS “Consultar Grupo Docente”	88
Figura 32. Diagrama de clases Web: CUS “Gestionar Crédito Bonificación”	89
Figura 33. Diagrama de clases Web: CUS “Gestionar Crédito Premio”.....	90
Figura 34. Diagrama de clases Web: CUS “Gestionar Crédito por Actividades”.....	91
Figura 35. Diagrama de clases Web: CUS “Gestionar Control Asistencia”.....	92
Figura 36. Diagrama de clases Web: CUS “Generar Acta de Examen Final”	93
Figura 37. Diagrama de clases Web: CUS “Gestionar Grupo Docente”	94
Figura 38. Diagrama de clases Web: CUS “Gestionar Matrícula en Grupo Docente”.....	95
Figura 39. Diagrama de clases Web: CUS “Gestionar Registro Evaluación”	96
Figura 40. Diagrama de clases Web: CUS “Mostrar Registro de Asistencia”	97
Figura 41. Diagrama de clases Web: CUS “Mostrar Registro de Evaluación”	98
Figura 42. Modelo lógico.....	99
Figura 43. Modelo físico.....	100
Figura 44. Diagrama de despliegue	101

Índice de tablas

Tabla 2.1. Actores del negocio.....	40
Tabla 2.2. Trabajadores del negocio.....	41
Tabla 2.3. Caso de uso “Solicitar Asignatura de Adelanto”	43
Tabla 2.4. Caso de uso “Cerrar Período”	44
Tabla 2.5. Caso de uso “Controlar Asistencia”	45
Tabla 2.6. Caso de uso “Controlar Evaluación”	45
Tabla 2.7. Caso de uso “Acreditar Premio”	46
Tabla 2.8. Caso de uso “Acreditar Bonificación”	48
Tabla 2.9. Caso de uso “Crear Grupo Docente”	48
Tabla 2.10. Actores del sistema	56
Tabla 2.11. Resumen del caso de uso “Gestionar Grupo Docente”	60
Tabla 2.12. Resumen del caso de uso “Gestionar Matrícula en Grupo Docente”	61
Tabla 2.13. Resumen del caso de uso “Gestionar Control Asistencia”	61
Tabla 2.14. Resumen del caso de uso “Gestionar Registro Evaluación”	61
Tabla 2.15. Resumen del caso de uso “Gestionar Crédito Premio”	62
Tabla 2.16. Resumen del caso de uso “Gestionar Crédito Bonificación”	62
Tabla 2.17. Resumen del caso de uso “Asignar Profesor a Grupo Docente”	62
Tabla 2.18. Resumen del caso de uso “Desubicar Profesor de Grupo Docente”	62
Tabla 2.19. Resumen del caso de uso “Gestionar Crédito por Actividades”	63
Tabla 2.20. Resumen del caso de uso “Consultar Grupo Docente”	63
Tabla 2.22. Resumen del caso de uso “Mostrar Registro de Evaluación”	63
Tabla 2.23. Resumen del caso de uso “Generar Acta de Examen Final”	64

Tabla 2.24. Resumen del caso de uso “Solicitar Asignatura Adelanto”	64
Tabla 2.25. Resumen del caso de uso “Aprobar_Rechazar Solicitud de Asignatura”	64
Tabla 3.1. Factor de peso de los actores sin ajustar	67
Tabla 3.2. Factor de peso de los casos de uso sin ajustar	68
Tabla 3.3. Factor de complejidad técnica.....	69
Tabla 3.4. Factor ambiente	70
Tabla 3.5. Distribución del esfuerzo	72

Introducción

La gestión académica, en un centro educacional, constituye un elemento fundamental. Esta determina el rumbo de la institución y posibilita la planificación, organización, gestión y control del proceso docente, siendo su correcto desenvolvimiento uno de los objetivos fundamentales del centro. La Universidad de las Ciencias Informáticas (UCI), es la casa de altos estudios que reúne mayor potencial de capital humano en la rama de la informática en el país. Entre los objetivos fundamentales de la universidad se encuentra la formación de profesionales comprometidos con la Revolución y altamente calificados como ingenieros, formados a partir de un modelo de formación profesional para y desde la producción.

En este centro de altos estudios, el buen desenvolvimiento del proceso docente educativo representa, sin dudas, el objetivo principal. Dentro de este proceso se encuentra el control docente que constituye una actividad importante dentro de la gestión académica de una universidad, el cual representa la forma de medir la eficiencia de este proceso y el punto de partida para la toma de decisiones por parte de los directivos de la institución. En él intervienen principalmente los resultados académicos alcanzados por los estudiantes.

Actualmente, la UCI cuenta con el Sistema Automatizado para la Gestión Académica – “Akademos”, que tiene como objetivo principal permitir la gestión automatizada de los elementos que intervienen en la labor académica de un centro de estudios, que pueda enfrentar los cambios de forma natural, adaptándose a las nuevas condiciones y formas de hacer con el menor costo.

Este sistema cuenta con un módulo llamado Registro del Profesor que posibilita registrar, por los profesores, las notas de las evaluaciones y asistencia de los estudiantes a los turnos de clases. Estos registros son sólo accesibles por los profesores y personal de secretaría, impidiendo la interacción en línea de los estudiantes con su información académica. Para asignarle un profesor a un grupo docente es necesario ir al módulo Profesor, lo cual hace el trabajo de secretaría más engorroso al tener que buscar funcionalidades dispersas en el sistema; provocando así pérdida de tiempo en la gestión. Solicitar una asignatura de adelanto es complejo, no se permite que los estudiantes hagan sus solicitudes a través del sistema. Estos módulos presentan, además, una dificultad de orden crítico y es que no poseen ninguna

documentación consistente que respalde su desarrollo y que permita la retroalimentación a nuevos miembros del equipo. Esto implica diversas complicaciones a la hora de realizar otras iteraciones o desarrollar nuevas versiones del sistema, igualmente hace difícil su mantenimiento puesto que requiere de mucho más tiempo y mayor cantidad de personas para analizar el funcionamiento de los módulos una vez que hayan sido desplegados. El hecho de que no exista documentación evoluciona en riesgos más complejos respecto a la calidad del desarrollo; como puede ser el obtener una aplicación que no satisfaga los requerimientos reales de los usuarios debido a la imposibilidad de controlar y revisar periódicamente el trabajo que se desarrolla con el objetivo de corregirlo y perfeccionarlo. Los mismos están desarrollados sobre tecnologías propietarias, por lo que no cumplen con la política de migración hacia software libre que se lleva a cabo en el país y en la universidad.

La actual versión del sistema y la relación que se establece entre los módulos anteriormente estudiados satisfacen muchas de las necesidades de la gestión académica de la UCI, pero no se adaptan a las necesidades y expectativas del usuario, por lo que se pretende realizar el análisis y diseño del Módulo Registro y Control Docente, para que en una futura implementación automatice con eficiencia las funcionalidades existentes, que permita a los estudiantes ver sus evaluaciones, las asistencias a clases, que no existan problemas con la información académica de los estudiantes, además de ser adaptable a cualquier centro de estudios.

Teniendo en cuenta lo anteriormente expuesto, el problema científico queda formulado de la siguiente forma: ¿Cómo transformar las necesidades de los clientes a un lenguaje entendible por los desarrolladores, que permita la posterior implementación?

El objeto de estudio de la presente investigación lo constituyen los procesos de control docente en los sistemas de gestión académica. Derivándose que el campo de acción es el proceso de control docente en el Sistema Automatizado para la Gestión Académica – “Akademos”.

Se define como objetivo de la investigación: elaborar el análisis y diseño del Módulo Registro y Control Docente para el proyecto Akademos V2.0. De ahí se derivan los siguientes objetivos específicos:

- Desarrollar el marco teórico de la investigación.

- Identificar los procesos relacionados con el Módulo Registro y Control Docente en la Universidad de las Ciencias Informáticas.
- Definir el análisis y diseño del Módulo Registro y Control Docente.

Para guiar la investigación se plantea la siguiente idea a defender:

La realización del análisis y diseño del Módulo Registro y Control Docente del proyecto Akademos V2.0 facilitará una eficiente implementación del mismo.

Para lograr el cumplimiento de los objetivos se proponen las siguientes tareas investigativas:

- Realizar un estudio del estado del arte de los procesos de gestión académica en otras instituciones similares a la estudiada.
- Fundamentar el uso de metodologías, tecnologías y herramientas a utilizar.
- Entrevistar a los clientes para estudiar los procesos del Módulo Registro y Control Docente.
- Estudiar la arquitectura propuesta para el sistema.

Para llevar a cabo las tareas investigativas se emplearon métodos teóricos y empíricos de la investigación científica. Los métodos teóricos utilizados para cumplir con las tareas a desarrollar son:

Histórico Lógico: este método se utiliza porque posibilita un mejor análisis histórico de los procesos de control docente, es decir, permite analizar la trayectoria de los procesos, desde su desenvolvimiento hasta las conexiones históricas más importantes.

Modelación: este método se usa porque permite crear abstracciones con el objetivo de investigar la realidad. El mismo se pone en práctica en el trabajo al realizar el análisis de la realidad mediante diversos modelos y diagramas que ayudan a comprender mucho más el objeto en su totalidad.

Analítico-Sintético: posibilitó el análisis de los procesos de Registro y Control Docente para determinar con exactitud cómo este funciona. Como resultado, se toman todas las características principales para lograr

modelar un sistema que logre una integración eficaz y una armonía dentro de los procesos que rigen su comportamiento.

Los métodos empíricos utilizados para obtener información sobre el objeto de estudio son:

Observación: este método se usa porque permite investigar los procesos externamente sin tener que llegar a la esencia de los mismos, lo que ayudó al planteamiento del problema científico, además de permitir conocer bien el proceso delimitado como objeto de estudio, lo cual ayuda a tener un conocimiento más detallado de lo que se quiere, lo que hace falta hacer y cómo hay que hacerlo.

Entrevista: se usa este método porque ayuda a obtener información, comprender y precisar bien el problema a resolver, así como a validar la propuesta que se presentará al cliente.

El presente trabajo está fragmentado en 3 capítulos.

En el capítulo 1, fundamentación teórica, se presentan los sistemas de gestión académica utilizados en universidades cubanas y del mundo y los utilizados en la UCI desde sus inicios. Por otra parte se explican las tecnologías actuales que se emplearán en el desarrollo del sistema y se fundamenta la metodología, lenguaje, gestor de base de datos y programas de software utilizados en el mismo.

En el capítulo 2, descripción de la solución propuesta, se describen los procesos del negocio que tienen que ver con el objeto de estudio. Se plantean los requisitos funcionales y no funcionales que debe tener el sistema para obtener de él los óptimos resultados, mostrándose en forma de diagramas los casos de usos que de ellos se derivan.

En el capítulo 3, elaboración de la solución propuesta, se realiza una estimación del esfuerzo necesario para la confección del módulo a partir del método de estimación por puntos de casos de uso. Se modelan, además, los diagramas de clases del análisis y del diseño dejando todo listo para la etapa de implementación. Se hace referencia a los patrones utilizados para la elaboración de la propuesta de diseño. Se muestra el modelo lógico y físico de datos, así como el diagrama de despliegue.

Capítulo I: Fundamentación teórica

1.1 Introducción

En este capítulo se realiza una investigación acerca de los sistemas de gestión académica que se utilizan en las diferentes universidades del país y de algunos relevantes en el mundo. Se describen además las características del lenguaje de programación, sistema gestor de base de datos, metodología de desarrollo y herramientas propuestas para dar solución al problema.

1.2 Los sistemas de gestión académica

La gestión académica constituye un elemento de gran importancia para el buen funcionamiento de cualquier centro educacional y tiene como tarea fundamental controlar, organizar y dirigir todas las entidades y actores que intervienen en la labor docente del mismo y que engloba, además, otros subprocesos como la definición de los planes de estudio, el control de la matrícula administrativa, el expediente académico y personal de los estudiantes, así como el control de la asistencia y las evaluaciones de los mismos.

1.2.1 Los sistemas de gestión académica en el mundo

En el transcurso de la investigación se estudiaron varios sistemas en el ámbito internacional con propósitos similares al estudiado en el campo de acción. A continuación se caracterizan algunos sistemas de gestión académica usados hoy en el mundo.

Sistema Integrado de Gestión Académica

Sistema Integrado de Gestión Académica (SIGA) es un sistema modular utilizado por la Universidad de Córdoba (España) para la administración académica y curricular. Es un producto diseñado para interactuar a través de Internet que permite compartir información eficiente y segura. Está compuesto de subsistemas que permiten la operación de los diferentes procesos académicos. Estos subsistemas se dividen en los siguientes módulos: General, Definición de Estudios, Alumnos, Opciones de Alumnos, Gestión de Tutorías, Gestión de Mensajerías, Gestión de Asistencias, Gestión de Calificaciones,

Profesores, Inventario, Ingresos y Gastos, Generador de informes, estadísticas y gráficos (GENFOR), Generador de Horarios y Generador de Diplomas.

El módulo de Alumnos tiene como tarea fundamental la gestión integral de todos los alumnos matriculados en el centro, alumnos antiguos, personas interesadas pero no matriculadas, alumnos que por algún motivo no están en el centro y la plaza está a disposición o libre para ser utilizada por otro estudiante, así como el paso de nivel de un estudiante a otro. Entre las funcionalidades más destacadas se encuentra el paso de alumnos al histórico según criterios, la matriculación automática de alumnos individualmente o en grupos y el análisis del origen del alumnado.

Módulo Gestión de Asistencias: permite controlar la asistencia del alumno a clases, las ausencias, sus causas y si es justificada o no; el motivo y otros datos de interés, incluido el grupo, el turno, la sesión, la duración en horas y minutos de la ausencia parcial.

Módulo Gestión de Calificaciones: posibilita al usuario definir cuántos exámenes querrá hacer y de qué tipo y así hacer medias de exámenes con condiciones. Se generan listados para los expedientes académicos, calificaciones de un alumno/curso, observaciones, diplomas, etc.

En SIGA los estudiantes sí interactúan con el sistema, existen varios módulos destinados a satisfacer las necesidades de estos, como son el módulo de Opciones de Alumnos, donde los estudiantes a través de los submódulos Gestión de Asistencias y Gestión de Calificaciones pueden conocer todos los detalles de su asistencia a clases y evaluaciones. Posee una gran desventaja; su interfaz no es amigable lo cual puede provocar serios problemas de utilización teniendo en cuenta la diversidad de usuarios que interactúan con el sistema (alumnos, profesores, personal del centro, etc.).

Control Escolar GES

“Control Escolar GES es un sistema planeado y desarrollado por la firma mexicana Grupo GES Sistemas Avanzados especializada en el software para la gestión escolar. Tiene la finalidad de automatizar los procesos en una institución educativa y de este modo mejorar la eficiencia en sus áreas de operación interna como lo son: Control Escolar, Control Docente, Académico, Administrativo-Financiero y de cobranza” [1]. Este sistema a diferencia de otros permite el registro de cualquier cantidad de alumnos, haciendo posible el desarrollo incremental de la matrícula de cualquier centro.

Presenta las siguientes funcionalidades relacionadas con el control docente:

Consulta General: esta funcionalidad brinda la oportunidad de generar consultas al expediente del alumno de forma sencilla y ver su estado actual.

Control de Grupos: esta funcionalidad se encarga de controlar los grupos con los que cuenta la institución, el cupo máximo de estudiantes y la cantidad que están inscritos en él, así como el profesor responsable.

Control Escolar GES es un sistema en el cual los estudiantes pueden ver toda su información a través de un expediente, permite llevar el control docente de todos los grupos existentes en el centro y generar reportes asociados a promedios finales, índice de reprobados, etc. pero no posee forma de integrar las funcionalidades al Akademos actual debido que gestiona información que no se adapta al proceso docente de la UCI y del país, tal es el caso de la planeación y la programación de pagos calendarizados, asignación de precio a los libros docentes, obtener presupuestos de ingresos, etc.

Ágora

Ágora es un producto de software estándar de gran calidad para la gestión de centros docentes y academias de todo tipo, desarrollado por la empresa española Kherian Soft. Se adecua a cualquier tipo de centro o formación, ya sea esta de tipo oficial o de carácter libre (academias de enseñanza general, de idiomas, informática, música u oposiciones, etc.), ya se trate de un pequeño centro o una gran empresa de formación con una gestión centralizada de varias sucursales distantes geográficamente. Para ello, la aplicación cuenta con varias versiones y múltiples opciones de organización docente para utilizar en cada caso.

“Ágora aporta soluciones en diversos sentidos, como son la gestión centralizada de datos a través de la cual se gestionan, de forma centralizada, todas las bases de datos de alumnos, profesores, aulas y clases; la generación automática de la documentación permite emitir entregas, recibos y facturas, además de proporcionar listados de soporte de todo tipo” [2]. Gestiona con gran facilidad las excepciones que trastocan la planificación: alumnos que sólo pueden venir a la mitad de las clases proyectadas de su grupo o que no se adaptan a ningún grupo y toman clases de varios, clases que cambian de horario, profesores que están de baja y han de ser sustituidos, etc. Con el control automatizado de la docencia real, el programa calcula cada día la docencia que, en función de los horarios y las matrículas, teóricamente se

habrá impartido, generando los registros correspondientes, permite al usuario verificarlos periódicamente para ajustar, con un simple clic, las incidencias (clase aplazada, no dada, impartida aunque no estaba prevista, etc.). Es especialmente eficaz para calcular los pagos a profesores. Además, calcula de forma automática la disponibilidad horaria de profesores, aulas, plazas en grupos y clases.

Este sistema posee una apariencia compleja y poco amigable debido a la gran cantidad de funcionalidades que presenta y tiene como inconveniente el alto costo de licencia para su uso, mantenimiento y soporte.

1.2.2 Los sistemas de gestión académica en Cuba

En las universidades cubanas se forman los profesionales que contribuyen al perfeccionamiento de la sociedad, transformándola, mejorándola e impulsando su desarrollo, por ello, es necesario lograr una buena organización y control de la gestión docente. En estos momentos, todas las universidades del país y el sistema de educación en general, realizan un arduo trabajo en este sentido, con el objetivo de lograr una mayor eficiencia en la planificación y elevar los resultados. A continuación se describen algunos de los sistemas de gestión académica usados en las universidades cubanas.

GESTACAD

GESTACAD es un sistema informático desarrollado por el Departamento de Informática de la Universidad de Matanzas "Camilo Cienfuegos". Permite actualizar y mantener la información sobre estudiantes y profesores de una universidad y obtener determinados resultados propios del trabajo de las áreas implicadas, aunque el grueso de las informaciones se obtiene mediante el acceso al sitio Web de la universidad.

El sistema es capaz de brindar las siguientes funcionalidades:

- Búsqueda de un alumno: brinda la ubicación según el horario docente.
- Listado de estudiantes por grupo.
- Reportes dinámicos de la información existente: se le da la posibilidad al usuario de seleccionar los campos de datos que desea obtener en el reporte, así como el título de este y las condiciones que debe cumplir la información a mostrar.
- Reporte de notas por asignatura y grupo: examen final, extraordinario, especial, premio, etc.

- Tabla con los resultados docentes de un grupo en un semestre.
- Reporte de los resultados académicos de un estudiante en toda su carrera: Hoja de Rendimiento.
- Actas de exámenes de las diferentes asignaturas.
- Registro de características de un grupo de estudiantes.
- Dar baja a un estudiante.

El sistema consta de los siguientes módulos relacionados con el control docente:

- Un módulo Web para las secretarías docentes: facilita la gestión de los estudiantes mediante la realización de acciones generales comunes en una secretaría docente, así como la obtención de reportes oficiales.
- Un módulo Web para los jefes de departamentos: se incluyen acciones relativas como la asignación de la carga docente y el control sobre los profesores del departamento.
- Un módulo Web para los profesores: permite a los profesores llevar el control docente de sus estudiantes, el control de las evaluaciones, así como reportes relativos a su carga docente.

GESTACAD presenta problemas con la navegabilidad, el ambiente de trabajo resulta en ocasiones restringido e inflexible, aspectos que recrudecen la búsqueda de información; la interfaz requiere ser más agradable, interesante y atractiva a la vista del usuario.

Sistema de Gestión de la Nueva Universidad

“El Sistema de Gestión de la Nueva Universidad (SIGENU) es un software que se ha desarrollado con el fin de ser una herramienta que permita la gestión de toda la información académica vinculada con la educación superior en Cuba. En correspondencia con su carácter nacional y la gran diversidad de sistemas de enseñanza superior con que cuenta la universidad cubana, este sistema ha sido concebido de manera tal que sea capaz de brindar gran seguridad e integridad de la información, y a la vez, ser tan flexible que permita ser adaptado a todos los centros de educación superior del país con sus diversas particularidades y distintas maneras de realizar determinados procedimientos” [3].

SIGENU está compuesto por cinco elementos fundamentales:

Base de datos: lugar en el cual se almacena la información del sistema.

Servidor de aplicaciones: hace posible que la información registrada en la base de datos sea visualizada y actualizada a través de las aplicaciones que son manipuladas por el usuario.

Módulo de Estadísticas: herramienta para los estadísticos, tanto de los Centros de Educación Superior (CES), como del Ministerio de Educación Superior (MES), con el que pueden generar de forma rápida y confiable los reportes de la información estadística oficial.

Módulo Web de Recuperación de Información (Recuperador): permite obtener diversos reportes con los que se puede recuperar toda la información necesaria del sistema.

Aplicación Cliente: aplicación de escritorio que permite la inserción y actualización de toda la información que se registra en el sistema y la obtención de reportes. Consta de los módulos:

- **Codificadores:** contiene toda la información con que debe contar el sistema y que es provista por el MES.
- **Matrícula:** permite realizar el proceso de matrícula a través del cual los estudiantes pasan a ser registrados en el sistema como estudiantes de Educación Superior.
- **Control de estudiantes:** permite buscar un estudiante registrado en el sistema, modificar los datos de un estudiante tanto personales como docentes, ubicar a un estudiante en un grupo o cambiarlo de grupo, realizar el pase de estudiantes a otros años de estudio y definir los que serán repitentes, así como dar baja a un estudiante del centro ya sea por licencia de matrícula, resolución o traslado.
- **Plan de Estudio:** permite la creación de los planes de estudio para las diferentes carreras del centro, así como realizar ajustes y modificaciones a un plan de estudio específico.
- **Evaluaciones:** permite registrar, modificar o eliminar las evaluaciones de los estudiantes registrados en el sistema, así como premios y bonificaciones.
- **Reportes:** permite obtener diversos reportes con los que se puede recuperar toda la información necesaria del sistema.

SIGENU posee la desventaja que es un sistema diseñado a nivel de secretaría, esto quiere decir que otros usuarios que forman parte importante del ámbito académico como los profesores y los estudiantes no pueden interactuar con él.

UCIMAT

UCIMAT fue implementado por profesores de la UCI con vistas a agilizar el censo de estudiantes por determinados criterios, incluye la matrícula y modificación de los datos de los estudiantes, así como reportes generales, pero sólo los que traía por defecto, de lo contrario, había que ir directamente a la base de datos; impidiendo que se puedan realizar otras operaciones importantes entre los que se puede destacar el reingreso, registro de traslado u otros datos requeridos. Fue utilizado durante los dos primeros cursos docentes, siendo de gran utilidad. Paralelamente con este, un poco adentrados en el curso 2002-2003, comienza a prestar servicios **GESTACAD**, cuando la universidad contaba con una matrícula de 2000 estudiantes y sólo un plan de estudio. Tenía muchas funcionalidades que, en el comienzo, fueron de mucha ayuda a la universidad, sin embargo, el sistema no obedecía a la tónica del sistema docente empleado en la UCI, ya que no tenía la posibilidad de contar con más de un plan de estudio y no daba un seguimiento en las evaluaciones de una asignatura, por citar algunos ejemplos, por lo que en el próximo año ya este sistema dejó de tener validez, pues la matrícula creció, se comenzaron a usar dos planes de estudio distintos y era necesario dar una trayectoria en las evaluaciones de los estudiantes.

A partir de estas necesidades surge, en el año 2003, **Akademos**, el Sistema Automatizado para la Gestión Académica, con el propósito de dar respuesta a la necesidad de sustentar y dar soporte en la UCI a toda la labor del personal de secretaría, con la visión de obtener un producto genérico, capaz de ser aplicable y adaptable en cualquier centro que implemente el control docente universitario. Los primeros módulos liberados fueron: Matrícula y Plan de Estudio, posteriormente, se fueron incorporando los demás módulos que actualmente conforman el sistema: Expediente, Profesor, Estudiante, Registro del Profesor y Reportes.

Akademos fue desarrollado tomando en cuenta una serie de principios básicos que determinan su estructura y funcionamiento:

- El dinamismo del proceso de gestión académica constituye la principal fuente de riesgos para un sistema que intente automatizarlo.

- Un sistema que automatice la gestión académica debe lograr que todos los involucrados (directivos, personal de secretaría, profesores y estudiantes) tengan un papel activo en el proceso.
- El plan de estudio es la entidad fundamental del proceso de gestión académica y rige todos sus subprocesos (matrícula, control, planificación, etc.).

Estos principios están sustentados en los siete módulos que componen el sistema y que interactúan coordinadamente entre sí para dar cumplimiento a las tareas del negocio, estos módulos se describen a continuación:

Módulo Plan de Estudio: el plan de estudio es la entidad fundamental del sistema. Este se define en Akademos como una sucesión de niveles, divididos en momentos que los estudiantes deben vencer una vez concluidas las asignaturas, las mismas se ubican en un orden de precedencia dado en los diferentes momentos del plan de estudio. Este módulo también permite la definición de los diferentes perfiles y disciplinas en las que se agrupan las asignaturas.

Módulo Matrícula: es el encargado del control de los estudiantes. Permite no sólo la gestión de la matrícula administrativa, sino también la de los movimientos a que son sometidos los estudiantes en su paso por la universidad y la definición de los estados y las transiciones entre estos. Logra la descentralización del proceso de matrícula, aspecto fundamental en centros como la UCI que reciben cada año a miles de educandos.

Módulo Expediente: gestiona toda la información que constituye el grueso de la historia de la universidad y constituye un repositorio digital de los documentos de los estudiantes. Los expedientes pueden almacenar documentos basados en plantillas, así como otros de libre formato o generados por el propio sistema. Cada vez que se crea un documento este es almacenado junto con otros datos, como la fecha de creación y el autor, para facilitar su posterior localización.

Módulo Profesor: el objetivo de este módulo es permitir la asignación de los profesores a las diferentes estructuras administrativas. Es controlado por el personal de secretaría, que asocia profesores y estructuras a través de diferentes niveles de acceso, de manera que estos puedan mantener actualizado el registro académico de sus estudiantes. Además, permite establecer un balance de la carga docente de los profesores y un mejor control de estos por facultad y grupo.

Módulo Estudiantes: este módulo está diseñado específicamente para que el estudiante pueda consultar sus evaluaciones en las distintas asignaturas vencidas, así como su desempeño en las disímiles materias del período docente en curso. Permite el acceso a su expediente y con esto a documentos como la Hoja de Matrícula y la Hoja de Prematrícula.

Módulo Reportes: la generación de reportes es quizás la tarea fundamental de todo sistema de gestión de datos. Este módulo provee a los usuarios de un conjunto de herramientas, que permiten el diseño y publicación de reportes. Con esta funcionalidad el usuario puede diseñar reportes de todos los datos de los estudiantes, desde las notas hasta la procedencia social, convirtiéndose en una poderosa herramienta para el estudio de la comunidad estudiantil.

Módulo Registro del Profesor: su funcionalidad básica es controlar el registro docente de los estudiantes. Uno de los principios en que se basó el diseño de Akademos persigue que la información sea introducida al sistema por los mismos que la originan, previendo de esta manera la ocurrencia de errores. Son los profesores los que aplican las diferentes formas de control del avance de los estudiantes, de ahí que Akademos les permita directamente introducir los resultados de todas las evaluaciones que apliquen, así como mantener actualizado el registro de asistencia de todos los estudiantes en las distintas asignaturas.

Todos los sistemas estudiados anteriormente presentan características que engloban las generalidades del proceso de control docente, pero algunas de sus funcionalidades no se ajustan a las particularidades del proceso docente de la universidad, además están desarrollados, en su mayoría, sobre tecnologías propietarias, lo que va en contra de las políticas de migración hacia software libre del país y la universidad. Akademos es el sistema que actualmente satisface las necesidades de la gestión académica de la UCI y sus desarrolladores radican en el centro, por lo que es muy conveniente contar con un sistema cuyo equipo de desarrollo se encuentre en la universidad. De ahí que la dirección del proyecto determinó realizar un sistema similar a Akademos donde se integren funcionalidades nuevas y se mejoren las existentes, con el fin de obtener un producto altamente comercializable y que presente flexibilidad para que pueda ser implantado en cualquier centro de estudio, a la par que facilite su adaptación a los cambios que se necesiten introducir.

1.3 Software libre

El software libre es un movimiento tecnológico que ha revolucionado la sociedad. Presenta características especiales que han permitido la experimentación de nuevas formas de desarrollo y mantenimiento de programas, nuevos modelos económicos y nuevas normas legales. “Es un asunto de libertad, no de precio. Para entender el concepto, se debe pensar en “libre” como en “libertad de expresión” [4].

“Software libre es aquel que puede ser distribuido, modificado, copiado y usado; por lo tanto debe venir acompañado del código fuente para hacer efectivas las libertades que lo caracterizan. De modo más preciso, se refiere a cuatro clases de libertad para los usuarios del software: la libertad de usar el programa, con cualquier propósito (libertad 0), la libertad de estudiar cómo funciona el programa y adaptarlo a tus necesidades (libertad 1), la libertad de distribuir copias, con lo que puedes ayudar a tus amigos (libertad 2) y la libertad de mejorar el programa y hacer públicas las mejoras a los demás, de modo que toda la comunidad se beneficie (libertad 3)” [4].

La libertad para utilizar un programa significa que cualquier individuo u organización puede ejecutarlo desde cualquier sistema informático, con cualquier fin y sin la obligación de comunicárselo ni al desarrollador ni a ninguna entidad en concreto. La libertad para redistribuir copias supone incluir las formas binarias o ejecutables del programa y el código fuente tanto de las versiones modificadas, como de las originales, ya que se debe tener la libertad para redistribuir tales formas si se encuentra el modo de hacerlo, pues las libertades para hacer cambios y para publicar las versiones mejoradas requieren de la accesibilidad de código fuente, por supuesto de manera libre, condición necesaria del software libre. Cuando se habla de software libre, se debe evitar utilizar expresiones como “regalar” o “gratis”, ya que se puede caer en el error de interpretarlo como una mera cuestión de precio y no de libertad. Evidentemente, el software libre, es la alternativa para los países subdesarrollados.

Ventajas

- Escrutinio público: el proceso de revisión al que está sometido el código fuente, lleva a un proceso dinámico de corrección de los errores. Cada mejora es socializada libremente, la comunidad puede cambiar la realidad de las innovaciones.

- Seguridad y privacidad: por su carácter abierto, dificulta la introducción de código malicioso, espía o de control remoto, debido a que el código lo revisan muchos usuarios y desarrolladores que pueden detectar posibles puertas traseras.
- Independencia del proveedor: puede ser usado aún después de que haya desaparecido la persona que lo elaboró, dado que cualquier usuario puede continuar desarrollándolo, mejorándolo o adaptándolo.
- Costo: el costo total de propiedad del sistema operativo libre Linux es menos de la mitad que el de Windows. El ahorro proviene de no tener que pagar licencia y de sus menores costos de administración.

Desventajas

Sin embargo, el uso de software libre no está exento de desventajas. Así se plantean las siguientes:

- Dificultad en el intercambio de archivos: esto se da mayormente en los documentos de texto creados con *Microsoft Word*, ya que si se abren con un software libre (*Open Office* o *LaTeX*) da error o se pierden datos.
- Mayores costos de implantación e interoperabilidad: dado que el software constituye "algo nuevo", ello supone afrontar un costo de aprendizaje, instalación, migración, interoperabilidad, etc.
- La posibilidad de una generación más fácil de troyanos: el código fuente también puede ser modificado con intenciones maliciosas. Si el troyano logra confundirse con la versión original puede haber problemas graves. La fuente del programa, de no ser seguro, permitirá que un tercer agente lo manipule.

1.3.1 Tendencias actuales del software libre

“Los expertos afirman que el software libre es hoy en día un recurso potente y maduro, capaz de dar soluciones prácticas y democráticas a las necesidades de la sociedad moderna. Bajo el lema “Innovación y Libertad”, que alude a la aplicación práctica de los sistemas de fuente abierta para conseguir mayores niveles de desarrollo y bienestar. Las principales personalidades en la materia coinciden en señalar que el software libre es clave en la educación digital y su universalización” [5]. Muchas empresas informáticas y compañías a nivel mundial, han apostado por el nuevo modelo en el uso del software libre. Este es el caso de IBM, la Administración del Espacio Aeronáutico Nacional (NASA) en los sistemas de misión crítica, el

Banco del Estado de Río Grande del Sur en su red de cajeros electrónicos, el motor de búsqueda *Google*, el portal *Yahoo*, el Departamento de Energía de Estados Unidos, los gobiernos de la Unión Europea, Australia, China, Venezuela, Brasil y muchos otros, los cuales han confirmado la calidad de los programas que son desarrollados sobre software libre, su gran estabilidad, seguridad y confiabilidad.

1.4 Herramientas, lenguajes y metodologías a utilizar

1.4.1 Aplicaciones Web

Con la introducción de Internet y de la Web, se han abierto infinidad de posibilidades en cuanto al acceso a la información desde cualquier parte del mundo. Esto representa un desafío para los desarrolladores de aplicaciones, ya que los avances en tecnología demandan aplicaciones más rápidas, ligeras, robustas, que permitan utilizar la Web y cuya finalidad sea que el usuario pueda tener una comunicación activa con la información, lo cual posibilita acceder a los datos de modo interactivo.

Ventajas de las aplicaciones Web:

1. Extrapolación y sindicación absoluta: el hecho de que todas las aplicaciones se realicen sobre la Web, va a permitir que entre ellas se pueda compartir toda la información.
2. Ubicuidad: la Web se ha consagrado como el canal de interoperabilidad por excelencia, es decir, las aplicaciones basadas en la Web pueden desarrollarse en cualquier terminal: ordenadores, móviles, TV digital, etc. Esto permite realizar consultas en cualquier parte del mundo donde tenga acceso a Internet y a cualquier hora.
3. Seguridad: la capacidad de seguridad y de protección de los datos será siempre mucho mayor que la mantenida en servidores compartidos o en los mismos ordenadores de gestión diaria. Otorga la flexibilidad de determinar niveles de acceso según la confidencialidad de los datos, así como la posibilidad de realizar transacciones *on-line*.

1.4.2 Modelo cliente-servidor

La utilización y aplicación del modelo cliente-servidor está fuertemente ligada al desarrollo de sistemas Web. Este modelo permite al usuario en una máquina, llamado “el cliente”, requerir algún tipo de servicio

de una máquina a la que está unido, llamado “el servidor”, mediante una red como una Red de Área Local (LAN) o una Red de Área Amplia (WAN).

Un único servidor típicamente sirve a una multitud de clientes, como se muestra en la figura 1, ahorrando a cada uno de ellos el problema de tener la información instalada y almacenada localmente.



Figura 1. Modelo cliente-servidor

En este modelo cuando un cliente desea un servicio, que proporciona cierto servidor, le envía un mensaje solicitando ese servicio: una petición (Ver figura 2). Los procesos clientes y servidores han de seguir un protocolo de comunicación que defina cómo se codifican las peticiones y cómo se sincronizan entre sí los procesos.

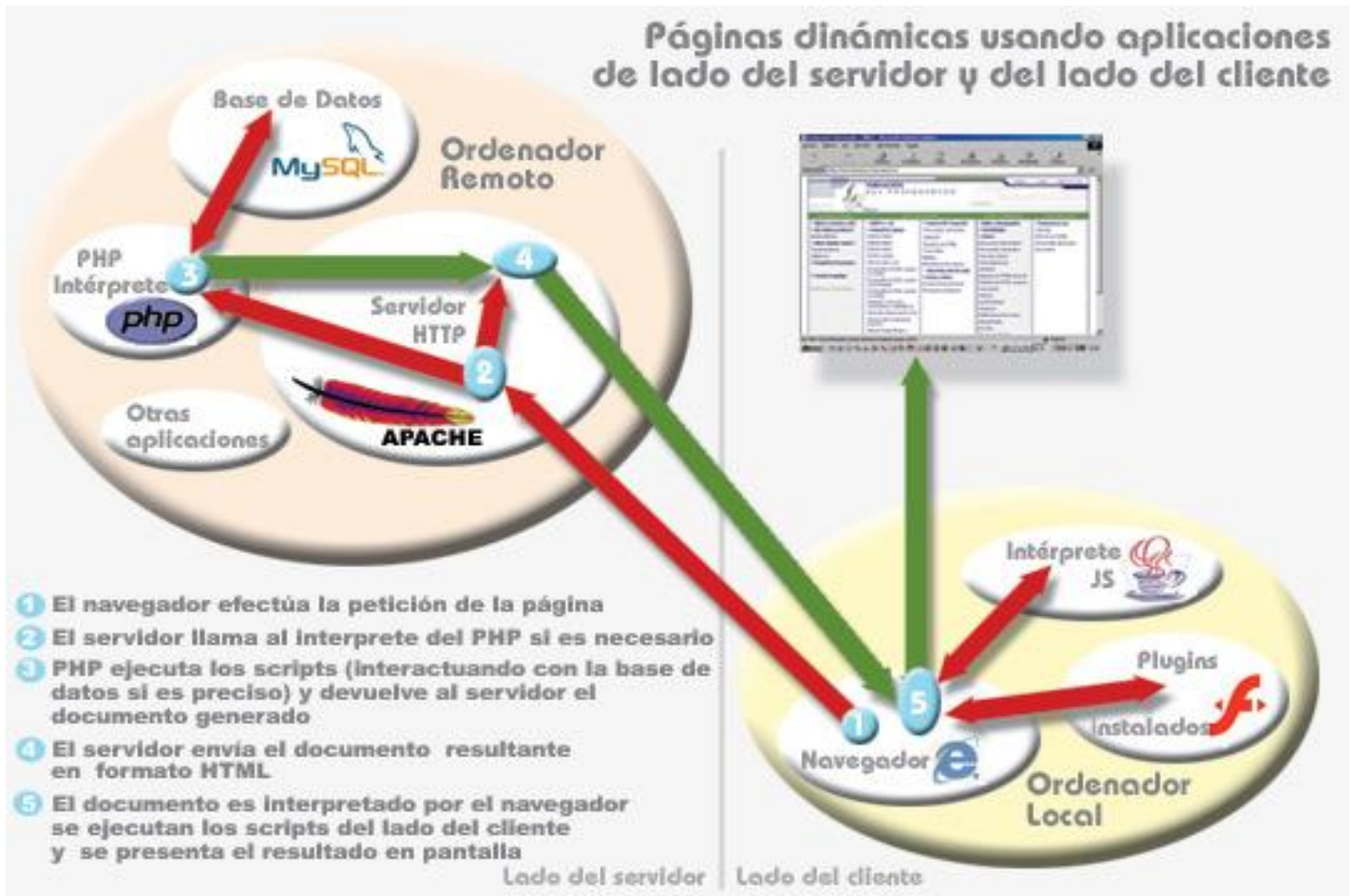


Figura 2. Solicitud de servicios utilizando modelo cliente-servidor

Algunas de las ventajas del modelo cliente-servidor se mencionan a continuación:

1. Se reducen los costos de producción de software y se disminuyen los tiempos requeridos, ya que para la construcción de una nueva aplicación pueden usarse los servidores que estén disponibles, reduciéndose el desarrollo a la elaboración de los procesos del cliente, según los requerimientos deseados.
2. Facilita el suministro de información a los usuarios, proporcionando una mayor consistencia a la información de la organización al contar con un control centralizado de los elementos compartidos. Permite llevar más fácilmente la información a donde se necesita y contribuye a aumentar su

precisión, pues se puede obtener de la fuente (el servidor) y no de una copia en papel o en medio magnético.

3. La habilidad de integrar sistemas heterogéneos es inherente al modelo cliente-servidor, pues los clientes y los servidores pueden existir en múltiples plataformas y tener acceso a datos de cualquier sitio de la red.
4. Favorece la adaptación a cambios en la tecnología, pues facilita la migración de las aplicaciones a otras plataformas y al aislar claramente las diferentes funciones de una aplicación, hace más fácil incorporar nuevas tecnologías en esta.

1.4.3 Servicios Web

Un Servicio Web es un componente de software que se comunica con otras aplicaciones a través de mensajes XML, se definen, además, como una función sin estado, auto-contenida, que acepta una o varias llamadas y devuelve una o varias respuestas mediante una interfaz bien definida que no depende del estado de otras funciones o procesos.

“Los Servicios Web tienen como base un conjunto de estándares abiertos, incluyendo XML, SOAP, WSDL y UDDI, los cuales son controlados por el W3C (*World Wide Web Consortium*)” [6].

1.4.3.1 Servidor Web Apache

Apache se basó originalmente en codificación e ideas basadas en el servidor HTTP más popular de todos, el NCSA httpd 1.3 (principios de 1995). Actualmente por su flexibilidad, rapidez, eficiencia, su constante actualización y adaptación a los nuevos protocolos (HTTP 1.1) es uno de los mejores servidores Web utilizados en Internet, según Netcraft Survey, empresa dedicada a la realización de encuestas a nivel global y estudios sobre el tráfico en Internet. (Ver figura 3)

Fue hecho para proveer un alto grado de calidad y fortaleza para las implementaciones que utilizan el protocolo HTTP, ha sido desde su salida al mercado, uno de los servidores de mayor popularidad, considerado el proyecto punta del movimiento de software libre.

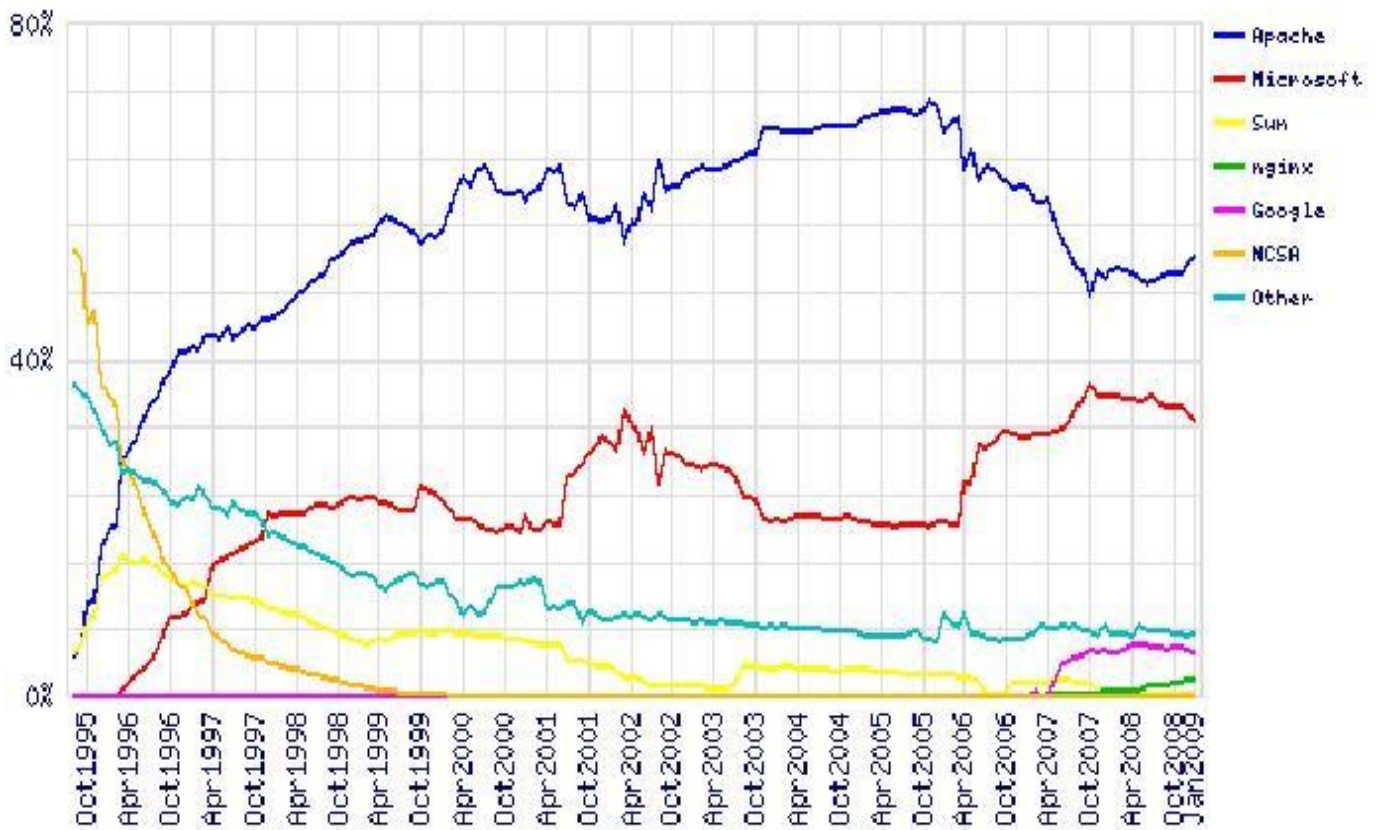


Figura 3. Análisis realizado por Netcraft de la explotación de servidores Web

Las principales características de Apache son:

- Funcionalidad en múltiples plataformas.
- Sencilla administración basada en la configuración de un único archivo.
- Soporte para Interfaz de Entrada Común (CGI de su significado en inglés *Common Gateway Interface*) y *FastCGI*.
- Soporte para el protocolo HTTP.
- Tecnología gratuita de código fuente abierto.

- Soporte de *host* virtuales.
- Servidor *proxy* integrado.
- Soporte para varios lenguajes: PHP, JAVA, Perl y librerías ASP.

1.4.4 Lenguajes de programación para la Web

En la actualidad los lenguajes de programación para la Web se clasifican en dos grupos teniendo en cuenta dónde se implementan respecto al modelo cliente-servidor, nombrándose lenguajes del lado del cliente y lenguajes del lado del servidor.

1.4.4.1 Lenguajes utilizados del lado del cliente

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

JavaScript: lenguaje de programación que ha permitido gran desarrollo en la animación de las páginas Web. La aparición de JavaScript ha resuelto de manera fácil y elegante la mayoría de los problemas con que se enfrenta el diseñador de páginas Web referidos a la programación, debido fundamentalmente, a que sus requerimientos son relativamente sencillos y lo que quizás sea más importante, es un lenguaje cuyos códigos se resuelven en el navegador del cliente, sin tener que ir y venir del cliente al servidor actualizando la información.

AJAX: JavaScript Asíncrono y XML de su significado en inglés *Asynchronous JavaScript And XML*, es un término que engloba a un grupo de tecnologías que trabajan conjuntamente para coordinar los lenguajes del lado del cliente con tecnologías del lado del servidor, uno de los principales objetivos es añadir más funcionalidad, especialmente para crear una experiencia interactiva que no requiera recargar la página

cada vez. Se ejecuta en el cliente, es decir, en el navegador del usuario y mantiene comunicación asíncrona con el servidor en segundo plano. De esta forma es posible realizar cambios sobre la misma página sin necesidad de recargarla. Esto significa aumentar la interactividad, velocidad y usabilidad en la misma.

1.4.4.2 Lenguajes utilizados del lado del servidor

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

Perl: es un lenguaje sencillo, poderoso, flexible y de propósito general. Creado, originalmente, para la manipulación de texto y que ahora es utilizado como una imprescindible herramienta para la creación de programas gráficos, construcción de páginas Web dinámicas, la automatización de tareas y muchas otras aplicaciones. Perl está implementado como un intérprete que su estructura deriva ampliamente del lenguaje C, por lo que sus elementos: escalares, listas, *hashes*, referencias a código, manejadores de archivo, están representados en el intérprete como estructuras de C.

JSP: Páginas de Servidor de Java de su significado en inglés *Java Server Pages*, es un lenguaje que utiliza Java para la generación de páginas Web dinámicas. Es una tecnología abierta, multiplataforma, los servidores Web, plataformas y otros componentes pueden ser fácilmente actualizados o cambiados sin que afecte a las aplicaciones basadas en la tecnología JSP. Es un lenguaje que hace mayor énfasis en los componentes que en los *scripts*, esto hace que sea más fácil revisar el contenido sin que afecte a la lógica o revisar la lógica sin cambiar el contenido. Se puede ejecutar en los sistemas operativos y servidores Web más populares: Apache, Netscape o Microsoft IIS.

PHP: acrónimo recursivo Pre-procesador PHP de Hipertexto de su significado en inglés *PHP Hypertext Pre-Processor*, constituye un lenguaje *script* de alto nivel interpretado del lado del servidor. Es un lenguaje de programación (originario del nombre *PHP Tools* o *Personal Home Page Tools*) que sirve principalmente para proporcionar características dinámicas a una página Web. Al ser ejecutado del lado del servidor, PHP permite acceder a los recursos internos del mismo o a otros externos, como por ejemplo a una base de datos, siendo el resultado normalmente una página HTML con los datos y acciones enviadas al cliente.

Algunas de las características básicas del lenguaje son:

1. “Gran comunidad de desarrolladores, como producto de código abierto, PHP goza de la ayuda de un gran grupo de programadores, que garantiza que los fallos de funcionamiento se encuentren y reparen rápidamente” [7].
2. Puede interactuar con muchos motores de bases de datos: MySQL, PostgreSQL, Oracle, MS SQL Server, Sybase mSQL, Informix, entre otros.
3. Integración con varias bibliotecas externas que permite, entre otras funcionalidades, manejo de gráficos, generar documentos en PDF (documentos de *Acrobat Reader*), calendarios, XML, IMAP, POP, etc.
4. El código se pone al día continuamente con mejoras y extensiones de lenguaje para ampliar las capacidades de PHP.
5. “Con PHP se puede hacer cualquier cosa que se realice con un *script* CGI, como el procesamiento de información en formularios, foros de discusión, manipulación de *cookies* y páginas dinámicas” [7].
6. Permite técnicas de programación orientada a objetos.

1.4.5 Metodologías de desarrollo del software

Todo desarrollo de software es riesgoso y difícil de controlar, pero si no se lleva una metodología de por medio, lo que se obtienen son clientes insatisfechos con el resultado y desarrolladores aún más insatisfechos. Lo más importante antes de elegir la metodología que se usará para la implementación de un software, es determinar el alcance que tendrá y luego de ahí ver cuál es la que más se acomoda a la aplicación.

Las metodologías fueron diseñadas bajo un conjunto de filosofías, fases, procedimientos, reglas, técnicas, herramientas, documentación y aspectos de formación para lograr la calidad, que es el principal objetivo estratégico en las organizaciones. Para el desarrollo del sistema informático propuesto, se estudian las dos corrientes más utilizadas en el mundo, conocidas como metodologías pesadas y metodologías ligeras. De ellas se estudiaron las tres metodologías más utilizadas en la universidad, dos ligeras y una pesada.

Metodologías ligeras

Scrum: define un marco para la gestión de proyectos. Es una metodología de desarrollo de software iterativa e incremental, un modelo de referencia que define un conjunto de prácticas y roles. Esta metodología se conforma por ciclos de desarrollos llamados *sprint*, los cuales se definen en un período de quince a treinta días, en el cual el equipo crea un incremento de software utilizable. Las acciones que forman parte de cada *sprint* vienen de un conjunto de requisitos de alto nivel priorizados que definen el trabajo a realizar. Estos elementos que forman parte del *sprint* se determinan durante una reunión, entonces, el equipo determina la cantidad de trabajo que puede comprometerse a completar durante el siguiente *sprint*.

EXtreme Programming (XP): es una metodología ágil centrada en potenciar las relaciones interpersonales como punto importante para el éxito en el desarrollo del software, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los miembros del equipo de desarrollo y propiciando un ambiente de trabajo adecuado. XP se basa fundamentalmente en la realimentación continua con el cliente y el equipo de desarrollo, simplicidad en la solución implementada y coraje para enfrentar los cambios. Se define especialmente para proyectos con requisitos imprecisos, cambiantes y donde existe un alto riesgo técnico. Entre las características esenciales que presenta XP se encuentran las historias de usuarios, técnica utilizada para especificar los requisitos del software.

Metodología pesada

El Proceso Unificado de Desarrollo (RUP): es un marco de desarrollo de software iterativo e incremental, que está compuesto por cuatro fases denominadas Inicio, Elaboración, Construcción y Transición. Está basado en componentes e interfaces bien definidas y utiliza el Lenguaje Unificado de Modelado (UML) para visualizar, especificar, construir y documentar un sistema de software.

Características de RUP:

- Dirigido por los casos de uso: los casos de usos son una forma abstracta de representar lo que los usuarios finales necesitan, por lo cual ellos son los que deben guiar el proceso de desarrollo. Esto se garantiza con la obtención de los diferentes modelos que son el resultado de los flujos de trabajo que propone RUP.
- Centrado en la arquitectura: la arquitectura de un proyecto muestra una visión común del sistema completo, visión en la que deben estar de acuerdo tanto el equipo de desarrollo como los clientes.

Es por ello que la arquitectura describe los elementos del modelo que son más importantes para la construcción del sistema, así como los cimientos para comprenderlo y desarrollarlo de forma económica.

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

1.4.6 Lenguaje Unificado de Modelado

UML es un grupo de especificaciones de notación orientadas a objeto, las cuales están compuestas por distintos diagramas, que representan las diferentes etapas del desarrollo de un proyecto de software. Es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad y está respaldado por el Grupo de Gestión de Objetos (OMG de su significado en inglés *Object Management Group*). Ofrece un estándar para describir un modelo del sistema, incluyendo aspectos conceptuales tales como procesos de negocio, funciones del sistema, aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y componentes reutilizables. Este lenguaje indica cómo crear y leer los modelos, pero no dice cómo desarrollarlos, esto último es el objetivo de las metodologías de desarrollo.

Los objetivos de UML son muchos, pero se pueden sintetizar sus funciones:

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

1.4.7 Herramientas de modelado de software

En la actualidad existen infinidad de herramientas de modelado, entre las que se encuentran: ArgoUML, BoUML, Umbrello, Visual Paradigm y Rational Rose Enterprise Edition. Las dos últimas son propietarias, pero en el caso de Visual Paradigm permite la generación de código PHP que es el lenguaje propuesto por sus amplias características para el desarrollo de la nueva versión de Akademos.

Herramienta CASE. Rational Rose

Rational Rose es una herramienta para realizar el modelado visual, forma parte de un conjunto más amplio de herramientas que todas juntas abarcan el ciclo de vida del desarrollo de software. Permite completar diferentes disciplinas (flujos fundamentales) de RUP; incluye, a su vez, herramientas de ingeniería inversa y generación de código que facilitan el tránsito hasta el producto final.

Rational Rose, es considerado una de las mejores herramientas para traducir requisitos de alto nivel a una arquitectura basada en componentes. Se encuentra en la avanzada en cuanto al desarrollo de UML por lo que se ha convertido en una de las mejores opciones por la notación estándar que se provee para especificar, visualizar, construir productos de software y sistemas.

Umbrello

Umbrello es una herramienta de modelado libre para crear y editar diagramas UML, que ayuda en el proceso del desarrollo de software. Funciona en varios entornos de escritorios, pero está diseñado principalmente para Entornos de Escritorio K (KDE de su significado en inglés *K Desktop Environment*). Actualmente permite la creación de varios diagramas UML, tales como: diagrama de casos de uso, diagrama de componentes, diagrama de despliegue, diagrama de modelo entidad-relación, diagrama de clases, diagrama de secuencia, diagrama de estados, diagrama de actividades y diagrama de colaboración. Estos diagramas pueden ser agrupados en un sólo fichero XML si están relacionados y organizarlos en diferentes vistas (lógica, de casos de uso, de componentes, etc.), que a su vez pueden contener diagramas o carpetas con las que clasificar aún más estos. Umbrello facilita la creación de estos modelos, además de manualmente, importándolos a partir de código en C++, Java, Python, IDL, Pascal/Delphi, Ada o Perl (haciendo uso de una aplicación externa). Así mismo, permite crear un diagrama y generar el código automáticamente en los lenguajes antes citados.

Visual Paradigm

Visual Paradigm es una herramienta que utiliza el lenguaje de modelado estándar UML, diseñada para desarrollar software que utiliza la programación orientada a objetos. Entre las características más atractivas de Visual Paradigm es que permite generar a partir del modelado código PHP 5, realizar ingeniería tanto directa como inversa y generar la documentación del proyecto automáticamente en varios formatos como Web o PDF y facilitar a través de la herramienta Team Framework el trabajo colaborativo. Es una herramienta multiplataforma que se puede utilizar tanto en Linux como en Windows, por lo que cumple con las políticas de migración a software libre. Tiene una interfaz muy intuitiva y es de fácil aprendizaje para los desarrolladores. Permite la generación automática de diagramas a partir de las descripciones de los casos de uso, por ejemplo diagramas de secuencia, permitiendo la agilidad en el trabajo del analista.

1.5 Sistemas para la gestión de base de datos

Los sistemas que gestionan datos, así como las transacciones entre estos, constituyen una potente herramienta, la cual está estrechamente vinculada a sistemas de software robustos que necesiten de la persistencia, localización y explotación de un volumen amplio de datos. El hecho de contar con un repositorio dinámico de valores hace posible que las prestaciones del software aumenten. Esto permite hacer estudios y valorar los resultados obtenidos en investigaciones propias o de otras investigaciones en las que se almacenen datos en una base de datos.

MySQL

MySQL es un sistema de gestión de bases de datos relacional, licenciado bajo GPL de GNU. Fue creado por la empresa sueca: MySQL AB, que mantiene el derecho de autor del código fuente del servidor SQL, así como también de la marca. Este gestor de bases de datos es probablemente el gestor más usado en el mundo del software libre, debido a su gran rapidez y facilidad de uso. Su amplia aceptación está dada, en parte, a que existen infinidad de librerías y herramientas que permiten su uso a través de gran cantidad de lenguajes de programación, además de su fácil instalación y configuración.

Entre sus principales características se pueden encontrar:

- Aprovechamiento de la potencia de sistemas multiprocesador, gracias a su implementación multi-hilo.
- Soporte de un número significativo de datos para las columnas.
- Posee API's para el soporte de varios lenguajes como: C, C++, Java, PHP, entre otros, lo que le permite gran portabilidad entre sistemas.
- Incorpora múltiples motores de almacenamiento (MyISAM, Margue, InnoDB, Memory/heap, MySQL Cluster, Federated, Archive, Blackhole, etc.), que le posibilita al usuario escoger el que sea más adecuado para cada tabla de la base de datos.

PostgreSQL

PostgreSQL es un sistema de gestión de bases de datos objeto-relacional basado en el proyecto *Ingres*, de la universidad de *Berkeley*, California. Pionero en muchos de los conceptos existentes en el sistema objeto-relacional actual, PostgreSQL incluye características de la orientación a objetos como pueden ser: la herencia, tipos de datos, funciones, restricciones, disparadores, reglas e integridad transaccional, además de otras específicas del gestor, como lo son: un mejor soporte para *sub-selects*, *triggers*, vistas y procedimientos almacenados.

Entre sus características más específicas se encuentran:

- Implementación del estándar SQL92/SQL99.
- Soporte para distintos tipos de datos, como son: datos de tipo fecha, datos sobre redes (MAC, IP), cadenas de bits, etc., así como la creación de tipos de datos propios.
- Al igual que con los tipos de datos, PostgreSQL permite la declaración de funciones propias, así como la definición de dispensadores.
- Soporte en la mayoría de los sistemas operativos más utilizados incluyendo, Linux, varias versiones de UNIX, BeOS y Windows.

1.6 Marcos de trabajo

Los marcos de trabajo presentan una estructura organizada que obliga a los programadores a seguir estándares y a trabajar de manera organizada. El uso de estas aplicaciones ha demostrado que la

organización de la programación influye notablemente en la calidad de las aplicaciones. Son diseñados con el intento de facilitar el desarrollo de software, permitiendo a los diseñadores y programadores pasar más tiempo identificando requerimientos de software que tratando con los tediosos detalles de bajo nivel de proveer un sistema funcional.

Symfony

Symfony es un marco de trabajo diseñado para optimizar el desarrollo de las aplicaciones Web dinámicas. Una de sus características más importantes es que separa la lógica de presentación de la de negocio y estas dos primeras de la de acceso a datos. Proporciona, además, un conjunto de herramientas y clases encaminadas a reducir el tiempo de desarrollo de una aplicación Web y automatizar las tareas comunes, permitiendo al desarrollador dedicarse por completo a los aspectos específicos de cada aplicación.

Symfony fue desarrollado utilizando PHP 5 sobre una estructura completamente libre y licenciado bajo MIT, licencia libre del Instituto de Tecnología de Massachusetts. Ha sido probado en numerosos proyectos reales y se utiliza en sitios Web de comercio electrónico de primer nivel. Symfony es compatible con la mayoría de los gestores de bases de datos: MySQL, PostgreSQL, Oracle y Microsoft SQL Server.

Algunas de sus características más destacadas son:

- Fácil de instalar y configurar en la mayoría de las plataformas existentes.
- Independiente del sistema gestor de bases de datos (SGBD).
- Sencillo de usar en la mayoría de casos, pero lo suficientemente flexible como para adaptarse a los casos más complejos.
- Basado en la premisa de “convenir en vez de configurar”, en la que el desarrollador sólo debe configurar aquello que no es convencional.
- Sigue la mayoría de mejores prácticas y patrones de diseño para la Web.

Zend Framework

Zend Framework es un marco de trabajo para construir sitios Web modernos, robustos y seguros utilizando PHP. Posee una arquitectura flexible orientada a componentes que usa el patrón Modelo-Vista-Controlador (MVC) desacoplado y es acoplable a varias tecnologías de desarrollo, pues se le pueden adicionar múltiples complementos externos. Además es código abierto, lo que ha posibilitado que al

código fuente se le hayan hecho numerosas pruebas y que posea una extrema simplicidad y productividad.

Entre sus principales características se encuentran:

- Trabaja con MVC.
- Cuenta con módulos para manejar archivos PDF, canales RSS, servicios Web, etc.
- Completa documentación y *tests* de alta calidad.
- Posee un buscador compatible con *Lucene*.
- Robustas clases para autenticación y filtrado de entrada.
- Clientes para servicios Web, incluidos *Google Data APIs* y *Strikelron*.
- Permite migrar aplicaciones de forma sencilla.
- Sus componentes se pueden utilizar de forma independiente sin tener que generar dependencias internas.
- Soporte avanzado para i18n.
- Constituye una solución para el acceso a base de datos que balancea el Mapeo Objeto-Relacional (ORM de su significado en inglés *Object Relational Mapping*) con eficiencia y simplicidad.
- Posee un marco de trabajo que incluye objetos de las diferentes bases de datos, por lo que es extremadamente simple para consultar su base de datos, sin tener que escribir ninguna consulta SQL.

Ext JS

Ext JS es una librería JavaScript que permite crear aplicaciones complejas utilizando tecnologías como AJAX, DHTML y DOM, componentes predefinidos, manejador de *layouts* similar al que provee *Java Swing*; gracias a esto provee una experiencia consistente sobre cualquier navegador, evitando el tedioso problema de validar que el código escrito funcione bien en cada uno (Firefox, Safari, etc.). Además provee al desarrollador de una ventana flotante que permite dibujar bordes con sólo moverla o redimensionarla, haciendo que el movimiento sea fluido, siendo esta una de las ventajas principales.

Dispone de un conjunto de componentes *widgets* para incluir dentro de una aplicación Web como: cuadros, áreas de texto, campos para fechas, campos numéricos, combos, *radiobuttons*, *checkboxs*,

editor HTML, árbol de datos, pestañas, barra de herramientas, menús al estilo de Windows, paneles divisibles en secciones, entre otros.

Varios de estos componentes están capacitados para comunicarse con el servidor usando AJAX. También contiene numerosas funcionalidades que permiten añadir interactividad a las páginas HTML, como son los cuadros de diálogos y *quicktips* para mostrar mensajes de validación e información sobre campos individuales.

Doctrine

Doctrine es un ORM para PHP, es decir, una interfaz capaz de traducir la lógica de los objetos a la lógica relacional lo que permite acceder de forma efectiva a una base de datos desde un contexto orientado a objetos. Brinda la posibilidad de exportar una base de datos existente a sus clases correspondientes y también a la inversa, es decir convertir clases (convenientemente creadas siguiendo las pautas del ORM) a tablas de una base de datos.

1.7 Entornos integrados de desarrollo

Un Entorno de Integrado Desarrollo (IDE de su significado en inglés *Integrated Development Environment*), es un conjunto de herramientas que ha sido creado como un programa de aplicación, es decir, consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica. Un IDE provee un marco de trabajo amigable para la mayoría de los lenguajes de programación y en algunos casos puede funcionar como un sistema en tiempo de ejecución en donde se permite utilizar el lenguaje de programación en forma interactiva, sin necesidad de trabajo orientado a archivos de texto.

Entre los entornos integrados de desarrollo se encuentran los siguientes:

- Eclipse
- VisualStudio.NET
- NetBeans
- SharpDevelop
- Zend Studio

Eclipse

Eclipse es un IDE desarrollado originalmente por IBM de código abierto y multiplataforma para desarrollar aplicaciones robustas. El IDE de Eclipse emplea varios complementos para proporcionar todas sus funcionalidades a diferencia de otros entornos donde estas están incluidas en el propio IDE por defecto, las necesite el usuario o no. Provee al programador marcos de trabajo para desarrollar aplicaciones gráficas, definición y manipulación de modelos de software, aplicaciones Web, etc. Por ejemplo, el Entorno de Edición de Gráficos (GEF de su significado en inglés *Graphic Editing Framework*) es un complemento de Eclipse para el desarrollo de editores visuales que pueden ir desde procesadores de texto "lo que ves es lo que obtienes" (*wysiwyg* de su significado en inglés "What You See Is What You Get") hasta editores de diagramas UML, interfaces gráficas para el usuario, etc.

Entre las características principales de Eclipse se encuentran:

- Editor de texto.
- Compilación en tiempo real.
- Pruebas unitarias con *Junit*.
- Control de versiones con CVS (*Concurrent Versions System*).
- Integración con Ant.
- Asistentes *wizards* para la creación de proyectos, clases, etc.
- Refactorización de código.
- Mediante varios complementos es posible añadir control de versiones con Subversion e integración con Hibernate.

NetBeans

NetBeans IDE es una herramienta para programadores de código abierto escrito en Java usando la plataforma NetBeans. Soporta el desarrollo de todos los tipos de aplicación Java, tales como: JEE (*Java Platform, Enterprise Edition*), Web, EJB (*Enterprise JavaBeans*) y aplicaciones móviles. Las funciones de este IDE son provistas por módulos donde cada uno provee una función bien definida: soporte de Java, edición o soporte para el sistema de control de versiones. NetBeans contiene todos los módulos necesarios para el desarrollo de aplicaciones Java en una sola descarga, facilitándole al usuario comenzar a trabajar inmediatamente.

La plataforma NetBeans es una base modular y extensible usada como una estructura de integración para crear aplicaciones de escritorio grandes. Ofrece servicios comunes permitiéndole al desarrollador enfocarse en la lógica específica de su aplicación.

Entre las características de la plataforma se encuentran:

- Administración de las interfaces de usuario.
- Administración de las configuraciones del usuario.
- Administración del almacenamiento.
- Administración de ventanas.
- Marco de trabajo basado en asistentes.

Zend Studio

Zend Studio o *Zend Development Environment* es un IDE orientado a desarrollar aplicaciones Web en lenguaje PHP. Está escrito en Java y consta de dos partes en las que se dividen las funcionalidades de parte del cliente y las del servidor. Las dos partes se instalan por separado, la del cliente contiene la interfaz de edición y la ayuda. Permite, además, hacer depuraciones simples de *scripts*, aunque para disfrutar de toda la potencia de la herramienta de depuración habrá que disponer de la parte del servidor, que instala Apache y el módulo PHP o en caso de que estén instalados, se configuran para trabajar juntos.

1.8 Fundamentación de la metodología y herramienta de modelado a utilizar

Se considera a RUP dentro de las metodologías estudiadas como la más óptima para modelar el sistema debido a las grandes potencialidades que presenta para el análisis, implementación y documentación de sistemas orientados a objetos. Está diseñado para adaptarse a las necesidades del proyecto o institución que lo esté utilizando. Otra de las características favorables que ofrece RUP es que permite mantener desde sus propios inicios una documentación profunda y detallada de todo el proceso en sentido general, planificar todas las etapas del desarrollo lo que es de gran importancia para el trabajo con equipos grandes y para la realización de proyectos complejos. Su característica de ser iterativo e incremental permitirá ir perfeccionando el software en cualquier momento del desarrollo teniendo en cuenta la variabilidad de las necesidades del cliente.

La herramienta para el modelado de dicho sistema, es el Visual Paradigm, ya que a pesar de ser un software privativo, la UCI adquirió la licencia y lo tiene como política de herramienta CASE a utilizar. Por sus características, cubre todo el ciclo de vida de un proyecto, permite establecer una trazabilidad real entre el modelo (análisis y diseño) y el código ejecutable, posee una herramienta llamada Team Framework que facilita el trabajo colaborativo en el que todos los agentes tienen sus propias vistas de información y utilizan un lenguaje común para comprender y comunicar la estructura y funcionalidad del sistema, permite generar a partir del modelado código PHP 5 que es el lenguaje de programación propuesto para la elaboración del sistema.

1.9 Fundamentación de las tecnologías, lenguajes y herramientas a utilizar

Hasta ahora se han analizado varios aspectos que son importantes a la hora de fundamentar cuál o cuáles son las tecnologías que se utilizarán en el desarrollo de la nueva versión de Akademos.

Este sistema forma parte de los proyectos de la universidad que se rigen por los lineamientos generales que son especificados por la Dirección de Informatización, la cual se encarga entre otras cosas, de estandarizar el uso de las tecnologías en los sistemas que son desarrollados.

El lenguaje a utilizar en el desarrollo de la propuesta de solución es PHP por sus sobradas ventajas sobre Perl o JSP en cuanto a facilidad a la hora de utilizarlo y por ser mucho más liviano. Es uno de los lenguajes para la Web más popular, tiene una comunidad muy grande de desarrolladores, el precio para utilizarlo es cero, es código abierto, por lo que si se desea agregar o modificar algo para obtener un funcionamiento de acuerdo a las necesidades de cualquier usuario puede hacerse con total libertad.

Para un mejor rendimiento y velocidad en los tiempos de respuesta de la aplicación se definió utilizar AJAX y para las validaciones del lado del cliente el lenguaje JavaScript.

Como SGBD de software libre se decidió utilizar PostgreSQL, ya que es el más recomendado para sistemas que tengan alta concurrencia y que manejen gran cantidad de información como es el caso de la aplicación en cuestión. Otra razón por la cual se seleccionó este SGBD es que facilita el trabajo con procedimientos almacenados y consultas, además que brinda una estabilidad muy alta y una gran seguridad de los datos.

Como servidor Web se usará Apache debido a que es el servidor más utilizado a nivel mundial y provee un alto grado de calidad y fortaleza para las implementaciones que utilizan el protocolo HTTP. Además, soporta el lenguaje PHP que es el que será usado en la implementación de Akademos.

Se usará NetBeans como IDE ya que es un producto de código abierto que ayuda a los equipos de desarrollo a utilizar las mejores prácticas y estándares de la industria del software para la productividad general del grupo, incluye estándares de la industria de software ampliamente aceptados, además de que se puede generar informes en HTML.

Para programar la capa de presentación se usará el Ext JS pues permite crear interfaces de usuarios muy amigables y funcionales de una manera intuitiva. Se utilizará el marco de trabajo Zend Framework para programar la capa de negocio, debido a que facilita diseñar su estructura de componentes con pocas dependencias. Esto trae consigo que los desarrolladores utilicen componentes individualmente los cuales forman una potente y extensible aplicación Web al ser combinados. Se utilizará como capa de abstracción de base de datos el marco de trabajo Doctrine pues presenta un gran rendimiento en ejecución y permite escribir de forma concisa consultas muy complejas.

1.10 Conclusiones

En este capítulo se detallaron las condiciones que rodean al objeto de estudio mediante los diferentes conceptos y definiciones planteadas, esclareciendo la lógica de negocio.

Se realizó un estudio detallado de varios sistemas que actualmente brindan un grupo de servicios de gestión académica en el ámbito nacional e internacional, el cual aportó un conjunto de conocimientos básicos sobre el estado del arte, así como posibles ideas y mejoras a desarrollar en la nueva solución.

Como lenguajes se usarán PHP 5.2.8, HTML, AJAX, JavaScript y UML; como SGBD: PostgreSQL 8.3.5; como metodología a utilizar: RUP; como servidor Web: Apache 2.2.9; como herramienta de modelado: Visual Paradigm 3.0; como marcos de trabajo: Ext JS 2.2, Zend Framework 1.7.6 y Doctrine 1.1.

Capítulo II: Descripción de la solución propuesta

2.1 Introducción

En este capítulo se realiza la descripción de los procesos del negocio que tienen que ver con el objeto de estudio. Se detallan las necesidades reales de los clientes y se presenta la propuesta de solución a través de un estudio del modelo de casos de uso del negocio, las reglas del negocio y la identificación de actores y trabajadores. Por último se presenta una propuesta del software a desarrollar, especificando los requerimientos funcionales y no funcionales y a partir de ellos la definición de los casos de uso del sistema y sus descripciones.

2.2 Descripción de los procesos vinculados al negocio

El sistema Akademos fue desarrollado para satisfacer las necesidades de diferentes direcciones y áreas de la UCI, fundamentalmente de la Secretaría General y la Vicerrectoría de Formación. Este sistema abarca todos los procesos involucrados en la gestión docente de un centro de estudios superiores, reportando eficiencia en la gestión académica de la UCI, lo cual es de vital importancia para el buen funcionamiento.

De forma general, Akademos provee las siguientes funcionalidades:

- Posibilita al personal de secretaría de un centro de estudios superiores, mantener actualizada la información asociada a la matrícula, tanto administrativa como académica del centro.
- Permite mantener una versión digital de los expedientes de los estudiantes de un centro de estudios superiores, garantizando la captación de la información.
- Permite a los directivos docentes la gestión efectiva de los planes de estudio presentes en un centro de estudios superiores y cuyas definiciones sirven de base a los restantes módulos del sistema.
- Los usuarios pueden crear de forma rápida reportes asociados a la información manejada por el sistema de gestión académica, así como unificar la generación de todos los reportes que necesiten el resto de los subsistemas.

- Posibilita, a todos los involucrados en el ciclo académico, llevar eficientemente el control directo del rendimiento de los estudiantes de forma tal que sirva de apoyo a la toma de decisiones de los directivos académicos.
- Posibilita el almacenamiento de la documentación generada por la gestión académica, viabilizando la construcción del historial de los centros de estudios.

Estas funcionalidades están sustentadas en un grupo de procesos que describen la gestión académica de la UCI. Dentro del control docente se encuentran identificados los siguientes subprocesos (Ver figura 4).

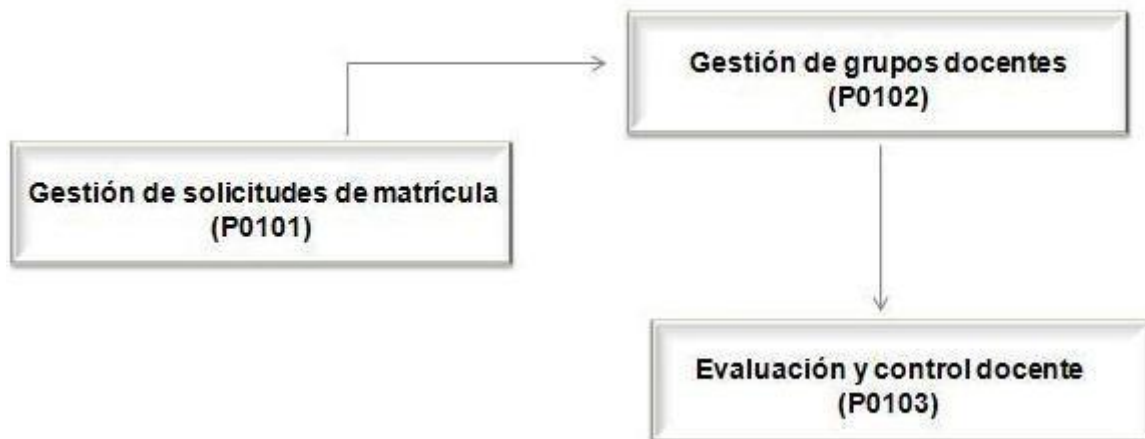


Figura 4. Esquema de procesos del Módulo Registro y Control Docente

Gestión de solicitudes de matrícula (P0101): el proceso lo inicializa el estudiante cuando le solicita a la Vicedecana de Formación de su facultad adelantar una asignatura. La Vicedecana de Formación chequea que el estudiante tenga vencidas las asignaturas que debe haber cursado antes de recibir la solicitada y la aprueba o la rechaza. Una vez aprobada la solicitud son las secretarías docentes de las facultades, las encargadas de matricular al estudiante en un grupo docente y llevar el control de los que se encuentran adelantando.

Gestión de grupos docentes (P0102): el proceso lo inicializa la Secretaria General de la universidad cuando le indica a las secretarías docentes de las facultades que se va a realizar el cierre de semestre, o sea, se va a llevar a cabo el cambio de período. Las secretarías docentes inmediatamente revisan que todos los estudiantes tengan sus notas pasadas ya sea en asignaturas básicas u optativas y mueven de grupo docente a los estudiantes que ellas estimen pertinente. Piden a los profesores las actas finales de las asignaturas, las cuales son archivadas en secretaría, obteniendo como principales artefactos las actas de examen final de las asignaturas y el informe de promoción y retención. Una vez cambiado el período tienen un tiempo determinado para generar los grupos docentes que corresponden al semestre, en dependencia de los planes de estudio y las asignaturas que estos tengan asociadas y matricular o eliminar estudiantes de acuerdo a las necesidades. Es tarea de las secretarías docentes una vez concluido el curso escolar desubicar de los grupos docentes a los estudiantes que suspendieron alguna asignatura y ubicar a aquellos que el Decano otorga licencia especial.

Evaluación y control docente (P0103): este proceso se divide en varios subprocesos que siguen determinado flujo de trabajo. La función fundamental comienza cuando al estudiante se le registra una evaluación, una asistencia o cuando solicita participar en un examen de premio o que se le acredite una bonificación. El profesor es el encargado de registrar las evaluaciones de los estudiantes que son matrícula de su grupo, así como la asistencia a los turnos de clases y en el caso de los injustificados especificar el argumento.

Los exámenes de premios son lanzados mediante una convocatoria por la Vicerrectora de Formación a nivel central. El estudiante se dirige al profesor de la asignatura en la cual fue lanzada la convocatoria y este confecciona un listado con los estudiantes que desean presentarse y lo entrega a la secretaria docente de su facultad. La secretaria es la encargada de verificar que el estudiante haya aprobado todas las asignaturas del semestre en ordinario y cerrado con cinco puntos la asignatura a la cual quiere presentarse a examen de premio. Una vez revisado estos aspectos el listado de los aprobados es enviado y los estudiantes se presentan al examen. Los tres primeros lugares son a los que se les acredita una nota; esto se hace mediante la revisión de la resolución ministerial que especifica la calificación que debe ser otorgada al estudiante de acuerdo al lugar obtenido. Esta nota es archivada en el Acta de Examen de Premio y se adjunta al Acta de Examen Final de la asignatura.

En el caso de las bonificaciones, el estudiante se presenta en secretaría docente con un documento que avale su participación en un evento. La secretaria verifica la autenticidad del documento y aprueba si es válido o no. Se revisa la resolución para ver el valor de la nota que se le registra al estudiante, la cual suma a su promedio general.

2.3 Descripción de las reglas del negocio

“Las reglas de negocio describen políticas que deben cumplirse o condiciones que deben satisfacerse, puesto que regulan algún aspecto del negocio. El proceso de especificación implica que hay que “identificarlas” dentro del negocio, “evaluar” si son relevantes dentro del campo de acción que se está modelando e “implementarlas” en la propuesta de solución “[8].

Se identificaron las siguientes reglas que deben seguirse en el análisis y diseño de la aplicación, a fin de respetar y garantizar las restricciones que existen en el negocio:

1. Un estudiante recibe todas las asignaturas que le corresponde recibir en el semestre del año en que se encuentre ubicado, excepto las que haya cursado anteriormente.
2. Para poder adelantar una asignatura debe solicitarlo y haber vencido todas las asignaturas que la preceden.
3. Las notas finales obtenidas por los estudiantes, en cada una de las asignaturas cursadas, son archivadas permanentemente.
4. Los estudiantes suspensos pueden ser reubicados nuevamente en otro grupo, sólo si les fue otorgada la repitencia.
5. Los grupos docentes sólo pueden ser creados por las secretarías docentes.
6. El profesor sólo puede ingresar la nota de una evaluación y la asistencia a los estudiantes del grupo que le imparte clases.
7. Para crear un grupo docente los estudiantes tienen que estar en estado de matriculados.
8. Para acreditar la nota de un examen de premio el estudiante tiene que haber vencido todas las asignaturas del semestre en ordinario y terminar con cinco puntos la asignatura que quiere realizar el examen de premio.

9. Un estudiante no puede pertenecer a dos grupos docentes de la misma asignatura.
10. Los estudiantes que se incluyan en el registro de una asignatura, son los que pertenecen a ese grupo docente.
11. Para crear un grupo docente tiene que tener al menos un estudiante.
12. La cantidad de estudiantes incluidos en un grupo docente es la matrícula de ese grupo docente.

2.4 Modelo del negocio

El modelo del negocio se realiza con el propósito de comprender las características y actividades que se llevan a cabo en el contexto a automatizar. En él, se modelan los clientes y trabajadores del negocio y su interacción con los procesos y objetos que lo componen.

2.4.1 Actores del negocio

“Los actores de un negocio representan el rol que juega un determinado individuo, entidad, organización, máquina o sistema de información externos al interactuar con el negocio para beneficiarse de sus resultados”[8]. A continuación se muestran los actores del negocio en la siguiente tabla.

Tabla 2.1. Actores del negocio

Actores del negocio	Justificación
Estudiante	Se beneficia con las actividades que son salidas de los procesos Gestión de solicitudes de matrícula (P0101) y Evaluación y control docente (P0103). En el proceso Gestión de solicitudes de matrícula (P0101) es quien solicita la matrícula en una asignatura de adelanto. En el proceso Evaluación y control docente (P0103) es a quien se le registra una evaluación, una asistencia, se le acredita una nota de examen de premio y se le acredita una bonificación.
Secretaria General	Interviene en el proceso Gestión de grupos docentes (P0102), pues inicializa las actividades que constituyen salidas a este proceso. Se beneficia con la realización de este.
Vicerrectora de Formación	Interviene en el proceso Evaluación y control docente (P0103), pues es quien lanza la convocatoria para realizar el examen de premio de una asignatura.

2.4.2 Trabajadores del negocio

“Un trabajador del negocio representa un rol que juega una persona (o grupo de personas), una máquina o un sistema automatizado actuando en el negocio” [8]. Se encuentran relacionados con las entidades del negocio, relaciones que se expresan en el diagrama de objetos del negocio que se modela en el Anexo 1.

Tabla 2.2. Trabajadores del negocio

Trabajadores del negocio	Justificación
Vicedecana de Formación	Es responsable de chequear que los estudiantes que soliciten adelantar una asignatura tengan vencidas las asignaturas que deben haber cursado antes de recibir la solicitada y aprobar o rechazar la solicitud.
Secretaria Docente	Es la encargada de matricular un estudiante en un grupo docente y llevar el control de los que se encuentran adelantando asignaturas. Es responsable de revisar que todos los estudiantes tengan sus notas pasadas ya sea en asignaturas básicas u optativas, generar los grupos docentes correspondientes a un semestre, matricular o eliminar estudiantes de un grupo docente, desubicar los estudiantes que suspendieron alguna asignatura culminado el curso escolar y ubicar a aquellos que le fue otorgada licencia especial. Verifica las condiciones para que un estudiante pueda presentarse a examen de premio, verifica la autenticidad del documento para acreditar las bonificaciones y pasa la nota de ambos conceptos.
Profesor	Es el encargado de pasar las notas de los estudiantes y la asistencia de estos en el registro de evaluación y de asistencia respectivamente.

2.4.3 Diagrama de los de casos de uso del negocio

“Un caso de uso del negocio representa un proceso de negocio, por lo que se corresponde con una secuencia de acciones que producen un resultado observable para ciertos actores del negocio” [8]. A continuación (Ver figura 5) se muestran los distintos casos de uso que fueron identificados en el negocio. Sus diagramas de actividades se encuentran modelados en el Anexo 2.

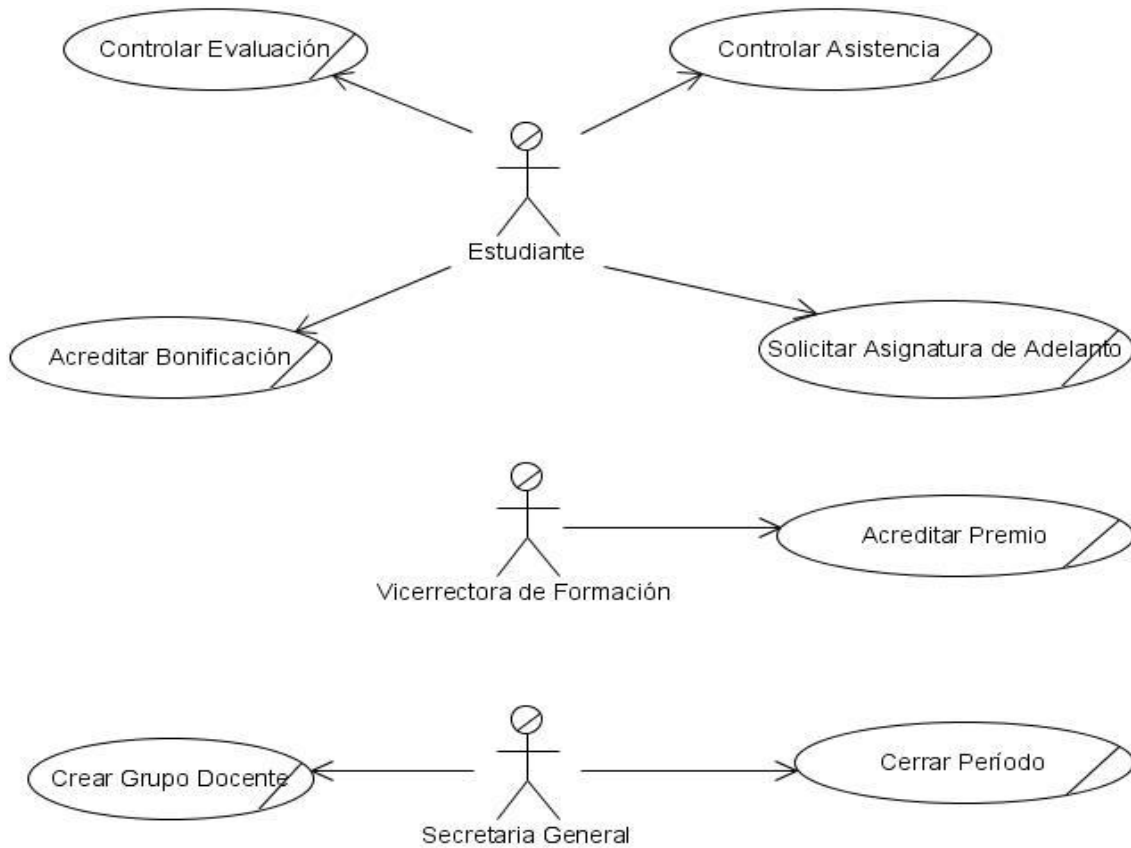


Figura 5. Diagrama de los casos de uso del negocio

2.4.4 Descripciones textuales de los casos de uso del negocio

Los casos de uso del negocio describen las actividades que se llevan a cabo en el negocio a partir de una secuencia de eventos donde se especifica la interacción de los actores del negocio con los procesos del mismo. A continuación se describen detalladamente los casos de uso del negocio.

Tabla 2.3. Caso de uso “Solicitar Asignatura de Adelanto”

Caso de Uso:	Solicitar Asignatura de Adelanto	
Actores:	Estudiante (Inicia)	
Trabajadores:	Secretaria Docente, Vicedecana de Formación	
Resumen:	El estudiante se dirige a la Vicedecana de Formación de su facultad para solicitar adelantar una asignatura. La Vicedecana de Formación chequea que el estudiante tenga vencidas las asignaturas que debe haber cursado antes de recibir la solicitada y culmina cuando la solicitud es aprobada o rechazada y las secretarias docentes matriculan al estudiante en un grupo docente.	
Precondiciones:	El estudiante debe haber vencido las asignaturas que preceden a la solicitada.	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Negocio	
1. El estudiante solicita adelantar una asignatura.	2. La Vicedecana de Formación chequea que el estudiante tenga vencidas las asignaturas que preceden a la solicitada. 3. La Vicedecana de Formación obtiene los resultados de la verificación del estudiante. 4. La Vicedecana de Formación informa al estudiante que puede adelantar la asignatura. 5. La Vicedecana de Formación le indica a las secretarias docentes que la solicitud fue aprobada. 6. Las secretarias docentes ubican al estudiante en un grupo docente.	
Flujos Alternos		
Acción del Actor	Respuesta del Negocio	
	4. La Vicedecana de Formación informa al estudiante que no puede adelantar la asignatura.	
Poscondiciones	-	

Tabla 2.4. Caso de uso “Cerrar Período”

Caso de Uso:	Cerrar Período	
Actores:	Secretaria General (Inicia)	
Trabajadores:	Secretaria Docente, Profesor	
Resumen:	La Secretaria General indica a las secretarias docentes de las facultades que se va a realizar el cierre de semestre. Las secretarias pasan las notas de los estudiantes que aún tengan alguna nota pendiente, piden a los profesores las actas finales de las asignaturas, las cuales son archivadas en secretaría, obteniendo como principales artefactos las Actas de Examen Final y el Informe de Promoción y Retención.	
Precondiciones:	-	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Negocio	
1. La Secretaria General indica que se va a llevar a cabo el cierre de semestre.	2. Las secretarias docentes pasan las notas de los estudiantes que aún tengan notas pendientes. 3. La secretaria docente solicita el acta de evaluación al profesor. 4. El profesor consulta el registro de evaluación para obtener las notas finales de los estudiantes. 5. El profesor crea el Acta de Examen Final poniendo las notas finales de los estudiantes. 6. El profesor pasa el Acta de Examen Final a la secretaria docente. 7. La secretaria docente archiva las Actas de Examen Final para realizar el cierre.	
	8. La secretaria docente obtiene el Informe de Promoción y Retención.	
Flujos Alternos		
Acción del Actor	Respuesta del Negocio	
Poscondiciones	-	

Tabla 2.5. Caso de uso “Controlar Asistencia”

Caso de Uso:	Controlar Asistencia	
Actores:	Estudiante (Inicia)	
Trabajadores:	Profesor	
Resumen:	El estudiante se presenta al aula y el profesor registra la asistencia, contabilizando la presencia de cada estudiante, así como los que no están presentes, de los cuales reconoce el motivo de la ausencia.	
Precondiciones:	-	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Negocio	
1. El estudiante se presenta en el aula.	2. El profesor busca el registro de asistencia. 3. El profesor materializa en el registro de asistencia el tipo de actividad docente. 4. El profesor localiza al estudiante. 5. El profesor pregunta por el estudiante.	
6. El estudiante responde.	7. El profesor reconoce la asistencia en el registro.	
Flujos Alternos		
Acción del Actor	Respuesta del Negocio	
1. El estudiante no se presenta en el aula.	5. El profesor pregunta motivo de ausencia.	
6. El estudiante alega motivo.	7. El profesor registra la ausencia y el motivo en el registro.	
Poscondiciones	-	

Tabla 2.6. Caso de uso “Controlar Evaluación”

Caso de Uso:	Controlar Evaluación
Actores:	Estudiante (Inicia)
Trabajadores:	Profesor
Resumen:	El estudiante realiza una evaluación y culmina cuando el profesor califica y pasa la nota al registro de evaluación.
Precondiciones:	-

Flujo Normal de Eventos	
Acción del Actor	Respuesta del Negocio
1. El estudiante realiza una evaluación.	2. El profesor estima resultado dependiendo de la respuesta del estudiante. 3. El profesor busca en el registro y registra la actividad docente. 4. El profesor localiza al estudiante en el registro. 5. El profesor escribe la nota en el registro. 6. El profesor comunica la nota al estudiante.
Poscondiciones	-

Tabla 2.7. Caso de uso “Acreditar Premio”

Caso de Uso:	Acreditar Premio	
Actores:	Estudiante, Vicerrectora de Formación (Inicia)	
Trabajadores:	Secretaria Docente, Profesor	
Resumen:	Los exámenes de premios son lanzados mediante una convocatoria por la Vicerrectora de Formación. El estudiante solicita al profesor que quiere presentarse a examen de premio. La secretaria hace las verificaciones pertinentes y envía un listado con los aprobados y los estudiantes se presentan al examen. El caso de uso culmina cuando se le pasa la nota al estudiante, la cual queda archivada en el Acta de Examen de Premio que se adjunta al Acta de Examen Final de la asignatura.	
Precondiciones:	El estudiante debe haber culminado las asignaturas del semestre en ordinario y con cinco puntos la asignatura a la cual quiere presentarse a examen de premio.	
Flujo Normal de Eventos		
Vicerrectora de Formación	Estudiante	Respuesta del Negocio
1. La Vicerrectora de Formación emite una convocatoria para examen de premio.	2. El estudiante confirma su participación con el profesor de la asignatura.	3. El profesor confecciona un listado con los estudiantes que desean presentarse a examen de premio. 4. El profesor entrega el listado a

		<p>la secretaria docente.</p> <p>5. La secretaria verifica que los estudiantes cumplan con las condiciones necesarias para presentarse al examen de premio.</p> <p>6. La secretaria docente envía el listado de los estudiantes.</p>
7. La Vicerrectora de Formación indica aplicar el examen.	8. El estudiante realiza el examen de premio.	
9. La Vicerrectora de Formación indica revisar el examen de premio.		10. El profesor revisa el examen y envía resultados de los tres primeros lugares.
		<p>11. La secretaria docente consulta la resolución que indica qué puntuación recibe el estudiante de acuerdo al lugar.</p> <p>12. La secretaria docente llena el Acta de Examen de Premio.</p>
	13. El estudiante firma el Acta de Examen de Premio.	14. La secretaria docente adjunta el Acta de Examen de Premio al Acta de Examen Final de la asignatura.
Flujos Alternos		
Acción del Actor		Respuesta del Negocio
		5. El estudiante no cumple con las condiciones para realizar el examen de premio.
		10. No hay aprobados en el examen de premio.
Poscondiciones	-	

Tabla 2.8. Caso de uso “Acreditar Bonificación”

Caso de Uso:	Acreditar Bonificación	
Actores:	Estudiante (Inicia)	
Trabajadores:	Secretaria Docente	
Resumen:	El estudiante se presenta en secretaría con un documento que avale su participación en un evento. La secretaria verifica la autenticidad del documento y aprueba si es válido o no. El caso de uso finaliza cuando se le acredita la nota al estudiante.	
Precondiciones:	El estudiante tiene que presentar el documento que prueba la bonificación.	
Flujo Normal de Eventos		
Acción del Actor		Respuesta del Negocio
1. El estudiante presenta documento que avale su participación en un evento.		2. La secretaria verifica la autenticidad del documento. 3. La secretaria docente consulta la resolución ministerial. 4. La secretaria docente acredita la bonificación.
Flujos Alternos		
Acción del Actor		Respuesta del Negocio
		3. El documento no es válido.
Poscondiciones	-	

Tabla 2.9. Caso de uso “Crear Grupo Docente”

Caso de Uso:	Crear Grupo Docente
Actores:	Secretaria General (Inicia)
Trabajadores:	Secretaria Docente
Resumen:	La Secretaria General le indica a la secretaria docente que se va a realizar el cambio de período. La secretaria verifica que todos los estudiantes tengan sus notas pasadas, matriculan o eliminan estudiantes de acuerdo a las necesidades y culmina cuando se crean los grupos docentes al inicio del semestre.
Precondiciones:	Todos los estudiantes pertenecientes a una facultad tienen que tener sus notas pasadas y estar en estado de matriculados.

Flujo Normal de Eventos	
Acción del Actor	Respuesta del Negocio
1. La Secretaria General indica que se va a llevar a cabo el cambio de período.	2. La secretaria docente pasa las notas de los estudiantes que tengan notas pendientes. 3. La secretaria docente matricula o elimina de los grupos docentes a los estudiantes que estimen pertinente. 4. La secretaria docente desubica de los grupos docentes a los estudiantes que suspendieron alguna asignatura. 5. La secretaria docente ubica en los grupos docentes a los estudiantes de licencia especial.
Poscondiciones	-

2.5 Modelado del sistema

Con el conocimiento adquirido hasta el momento sobre los conceptos que rodean al campo de acción, se pueden analizar las características que debe tener el sistema para que se cumplan los objetivos planteados al inicio. Para ello se identifican los requisitos funcionales y no funcionales.

2.5.1 Requerimientos funcionales

“Los requisitos funcionales son capacidades o condiciones que el sistema debe cumplir” [9]. Especifican acciones que el sistema debe ser capaz de realizar, sin tomar en consideración ningún tipo de restricción física, de manera que especifican el comportamiento de entrada y salida del sistema y surgen de la razón fundamental de la existencia del producto. A continuación se listan las funcionalidades que a partir de los procesos de negocio estudiados y las actividades a automatizar fueron identificadas. Para una mayor comprensión del lector ver especificación de los requisitos en el Anexo 3.

RF 1. Crear Grupo Docente

RF 2. Modificar Grupo Docente

RF 3. Eliminar Grupo Docente

- RF 4. Matricular Grupo Docente
- RF 5. Desmatricular Grupo Docente
- RF 6. Consultar Grupo Docente
- RF 7. Solicitar Asignatura de Adelanto
- RF 8. Aprobar Solicitud de Adelanto de Asignatura
- RF 9. Rechazar Solicitud de Adelanto de Asignatura
- RF 10. Controlar Asistencia
- RF 11. Modificar Control Asistencia
- RF 12. Eliminar Control Asistencia
- RF 13. Mostrar Registro de Asistencia
- RF 14. Controlar Evaluación
- RF 15. Modificar Control Evaluación
- RF 16. Mostrar Registro de Evaluación
- RF 17. Acreditar Crédito Premio
- RF 18. Modificar Crédito Premio
- RF 19. Eliminar Crédito Premio
- RF 20. Acreditar Crédito Bonificación
- RF 21. Modificar Crédito Bonificación
- RF 22. Eliminar Crédito Bonificación
- RF 23. Asignar Profesor a Grupo Docente

- RF 24. Mostrar Expediente del Estudiante
- RF 25. Generar Grupo Docente Auto
- RF 26. Mostrar Resumen Histórico
- RF 27. Acreditar Crédito por Actividades
- RF 28. Modificar Crédito por Actividades
- RF 29. Eliminar Crédito por Actividades
- RF 30. Crear Modelo de Corte
- RF 31. Crear Registro Asistencia
- RF 32. Crear Acta Examen Final
- RF 33. Mostrar Notas Históricas Básicas
- RF 34. Mostrar Notas Históricas Optativas
- RF 35. Mostrar Notas Históricas Finales
- RF 36. Mostrar Hoja de Resultados Académicos
- RF 37. Mostrar Certificación de Notas
- RF 38. Mostrar Promoción por Asignatura
- RF 39. Mostrar Estudiantes Pendientes
- RF 40. Mostrar Listado de Aprobados para Asignatura de Adelanto
- RF 41. Mostrar Promoción por Año por Asignatura
- RF 42. Desubicar Profesor de Grupo Docente
- RF 43. Mostrar Aprobados por Convocatorias Prueba Nivel

2.5.2 Requerimientos no funcionales

“Los requerimientos no funcionales son propiedades o cualidades que el producto debe tener. Debe pensarse en estas propiedades como las características que hacen al producto atractivo, usable, rápido o confiable” [9]. Entre los requerimientos no funcionales del sistema propuesto se encuentran:

Usabilidad:

- **Facilidad de uso por parte de los usuarios:** el sistema debe presentar una interfaz amigable que permita la fácil interacción con el mismo y llegar de manera rápida y efectiva a la información buscada. Debe, además, ser una interfaz de manejo cómodo que posibilite a los usuarios sin experiencia una rápida adaptación.
- **Especificación de la terminología utilizada:** el sistema debe adaptarse al lenguaje y términos utilizados por los clientes en la rama abordada con vista a una mayor comprensión por parte del cliente de la herramienta de trabajo.
- **Emplear perfiles de usuario:** diferenciar las interfaces y opciones para los usuarios que accedan al sistema con diferentes roles (secretaria general, secretaria docente, administrador, estudiantes, etc.).
- **Menús:** el sistema debe presentar una serie de menús tanto laterales como desplegables que permitan el acceso rápido a la información por parte de los usuarios, aprovechando así las potencialidades de estas estructuras.

Seguridad:

- **Seguridad de la base de datos:** la base de datos deberá estar fraccionada en esquemas que permitan un mejor uso de la información y la división de forma lógica de las funcionalidades del sistema, trayendo consigo además la protección de la información al ocurrir un incidente sobre una parte de la base de datos. El SGBD escogido debe presentar facilidades de administración de roles y usuarios restringiendo el acceso a los datos.
- **Servicios Web restringidos:** los servicios Web que brinde el sistema deben estar restringidos a grupos de usuarios definidos y aprobados previamente.

- Políticas de seguridad por usuarios y roles: el sistema debe contar con un grupo de políticas de accesibilidad a las diferentes funcionalidades del mismo en dependencia del nivel de autorización que presente un usuario determinado.
- Autenticación basada en dos niveles: el sistema debe tener la posibilidad de, en caso que no exista un Directorio Activo, gestionar usuarios y roles, así como las prioridades y niveles de acceso en los mismos.
- Registro sistemáticos de incidencias: el sistema debe ser capaz de registrar el accionar del usuario, así como permitir auditorías y exámenes de las trazas tanto en tiempo real como en históricos. Se precisa un monitor de incidencia para la visualización y tratamiento de las mismas.
- Alta protección de los datos: al estar trabajando con información sensible, se hace necesario una alta protección de los datos a nivel de aplicación y de tráfico por la red, para tal fin se ha definido además la seguridad en varios niveles dentro de la aplicación: nivel de interfaz, nivel de acceso a datos y nivel de base de datos.

Rendimiento:

- Escalabilidad: el sistema debe ser capaz de mantener un rendimiento apropiado y una estabilidad adecuada al gestionar amplios volúmenes de datos, así como de permitir la incorporación de nuevas funcionalidades adaptándose de manera natural a los procesos de gestión que se requieran.

Hardware:

- Para el desarrollo: PC Intel Pentium 4 o superior, CPU 3GHZ o superior, 512 MB RAM o superior, 160 GB HDD o superior.
- Para explotación del cliente: PC Pentium 3 o superior, CPU 133 MHZ o superior, 128 RAM mínimo 512 RAM recomendada o superior.
- Para explotación del servidor: CPU Dual Core 2.0 GHZ o superior, memoria RAM de 4 GB (recomendado 6 GB), 250 GB HDD.

Software:

- Para el cliente: sistema operativo con interfaz gráfica y conexión a red. Navegador Web: (Mozilla FireFox _ recomendado).
- Para el servidor: sistema operativo Linux; Ubuntu Server 7.10 o superior.
- Servidor Web: Apache 2.2.9.
- Gestor de base de datos: PostgreSQL 8.3.5.
- Software controlador de versiones: Subversion v_1.4.4 (r25188).

Restricciones de diseño:

- Lenguaje de programación: PHP 5.2.8.
- El marco de trabajo base de desarrollo que se utilizará es: Zend Framework 1.7.6.
- Como IDE se empleará NetBeans 6.5.
- Como servidor Web se explotará Apache 2.2.9.
- El SGDB deberá ser PostgreSQL 8.3.5.
- El diseño de la base de datos se realizará con DBDesigner 4.0.
- El modelado UML se hará con Visual Paradigm 3.0.
- El sistema operativo a utilizar en el entorno de desarrollo deberá ser: Windows XP SP 2 o Ubuntu 7.10 o superior.
- El repositorio principal, el entorno de prueba y el servidor de base de datos estarán montados sobre Ubuntu Server 7.10 o superior.

Apariencia o interfaz externa e interna:

- Interfaz Web: la interfaz deberá ser sencilla con colores suaves a la vista y sin cúmulo de imágenes u objetos que distraigan al cliente del objetivo de su empleo.
- Interfaz interna: la interfaz interna estará determinada por los desarrolladores, construyendo así una vista escalable de las clases o agrupaciones de clases que permitirán un mejor encapsulamiento de las funcionalidades y una mayor abstracción modular del sistema.

Soporte:

- Manual de usuario: el sistema deberá presentar un manual de usuario, permitiendo con ello un correcto uso de sus funcionalidades y brindarle al usuario una mayor experiencia del trabajo con el mismo.
- Documentación actualizada del grupo de desarrollo: se precisa que la documentación del sistema esté actualizada en todos los aspectos, fases de trabajo y ciclos de desarrollo del mismo, permitiendo con ello un respaldo tanto ingenieril como legal del desarrollo de dicho sistema.
- Grupo de soporte y asesoría: el sistema contará con un grupo de soporte y asesoría al cliente del producto destinado a brindar asesoría y soporte técnico al mismo.

Cuestiones legales:

- El sistema debe ser sometido a un análisis legal por parte de los abogados y personal autorizado con vistas a declarar su autenticidad y evitar restricciones legales para su uso y comercialización; así mismo se debe proceder a una evaluación y certificación por parte del cliente del producto.

Portabilidad:

- El sistema está basado en un diseño comercial, entendiéndose en un producto propiamente dicho y no en una solución abierta. Al ser conceptualizado como producto deberá poderse instalar, así como actualizar y desinstalar, sobre Linux o Windows respectivamente.

La actualización a la que se hace referencia vela por un conjunto de funcionalidades que se le pueden incorporar a los módulos ya instalados del sistema, así como otros que se le quieran agregar. Todo este diseño modular del sistema determina un bajo acoplamiento entre las distintas entidades del mismo.

2.5.3 Actores del sistema

“Cada trabajador del negocio (inclusive si fuera un sistema ya existente) que tiene actividades a automatizar es un candidato a actor del sistema. Si algún actor del negocio va a interactuar con el sistema, entonces también será un actor del sistema. Los actores del sistema no son parte de él, pueden intercambiar información con él, ser un recipiente pasivo de información y representar el rol que juega una o varias personas, un equipo o un sistema automatizado“[9].

Tabla 2.10. Actores del sistema

Actores del sistema	Justificación
Estudiante	Es la persona encargada de solicitar asignaturas de adelanto y el principal beneficiado en la realización de las funcionalidades: mostrar registro de evaluaciones, mostrar registro de asistencias y mostrar expediente académico.
Profesor	Es la persona encargada de generar y controlar el registro de asistencias y evaluaciones de los grupos docentes a los que imparte clases. Generar el Acta de Examen Final, el modelo de corte y consultar cualquier grupo docente.
Secretaria Docente	Es la persona encargada de la gestión de los grupos docentes, acreditar premios, bonificaciones, créditos por actividades, la asignación y desubicación de los profesores de un grupo y aprobar o rechazar las solicitudes de asignaturas de adelanto hechas por los estudiantes. Puede ver la hoja de los resultados académicos de un estudiante, el resumen histórico, la certificación de notas, entre otros reportes.

2.5.4 Diagrama de los casos de uso del sistema

“Los casos de uso son artefactos narrativos que describen, bajo la forma de acciones y reacciones, el comportamiento del sistema desde el punto de vista del usuario. Por lo tanto, establece un acuerdo entre clientes y desarrolladores sobre las condiciones y posibilidades (requisitos) que debe cumplir el sistema” [9]. Para una mejor comprensión del sistema la relación entre los actores del sistema se representa a través de un catálogo de actores (Ver figura 6) y los casos de uso se representan agrupados por paquetes (Ver figura 7).

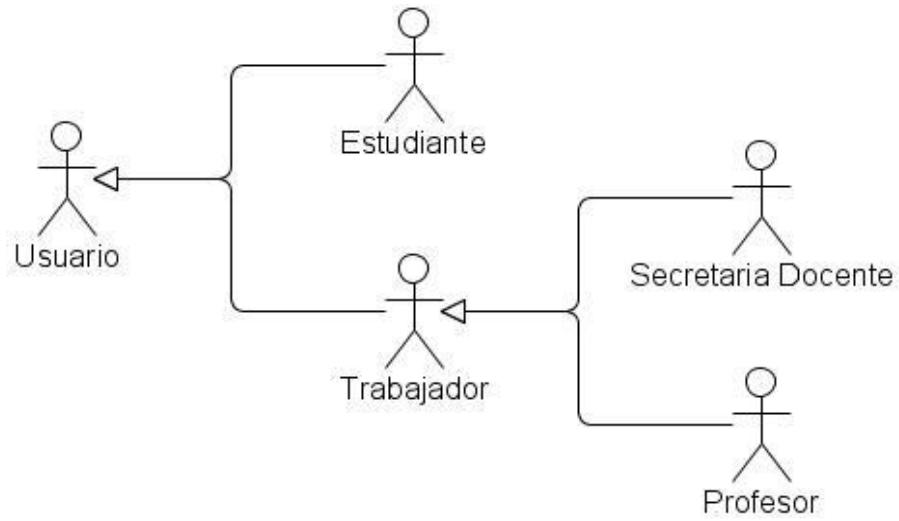


Figura 6. Catálogo de actores del sistema

En la siguiente figura se muestra el diagrama de paquetes de los casos de uso del sistema:

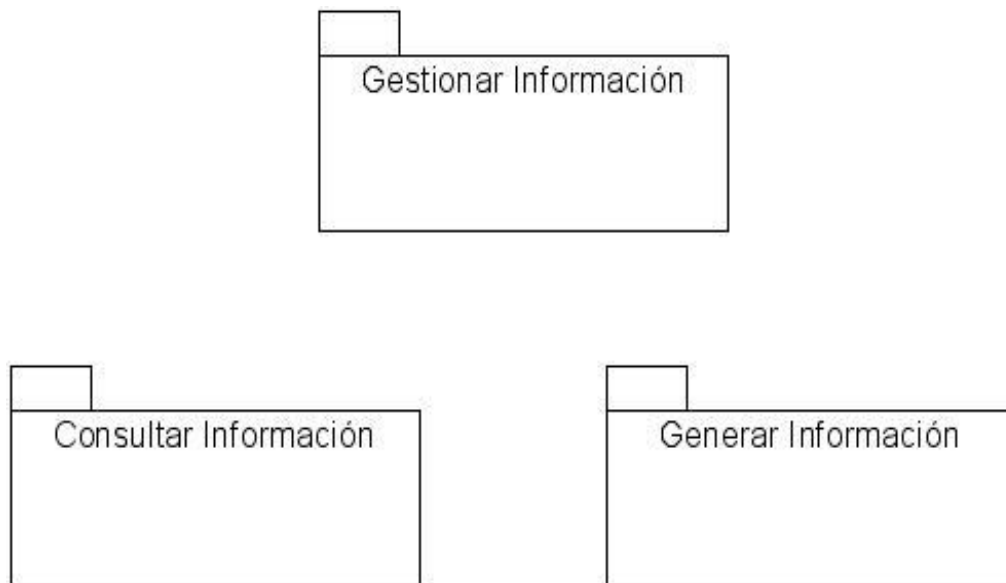


Figura 7. Diagrama de paquetes de los casos de uso del sistema

El paquete “Gestionar Información” contiene los siguientes casos de uso:

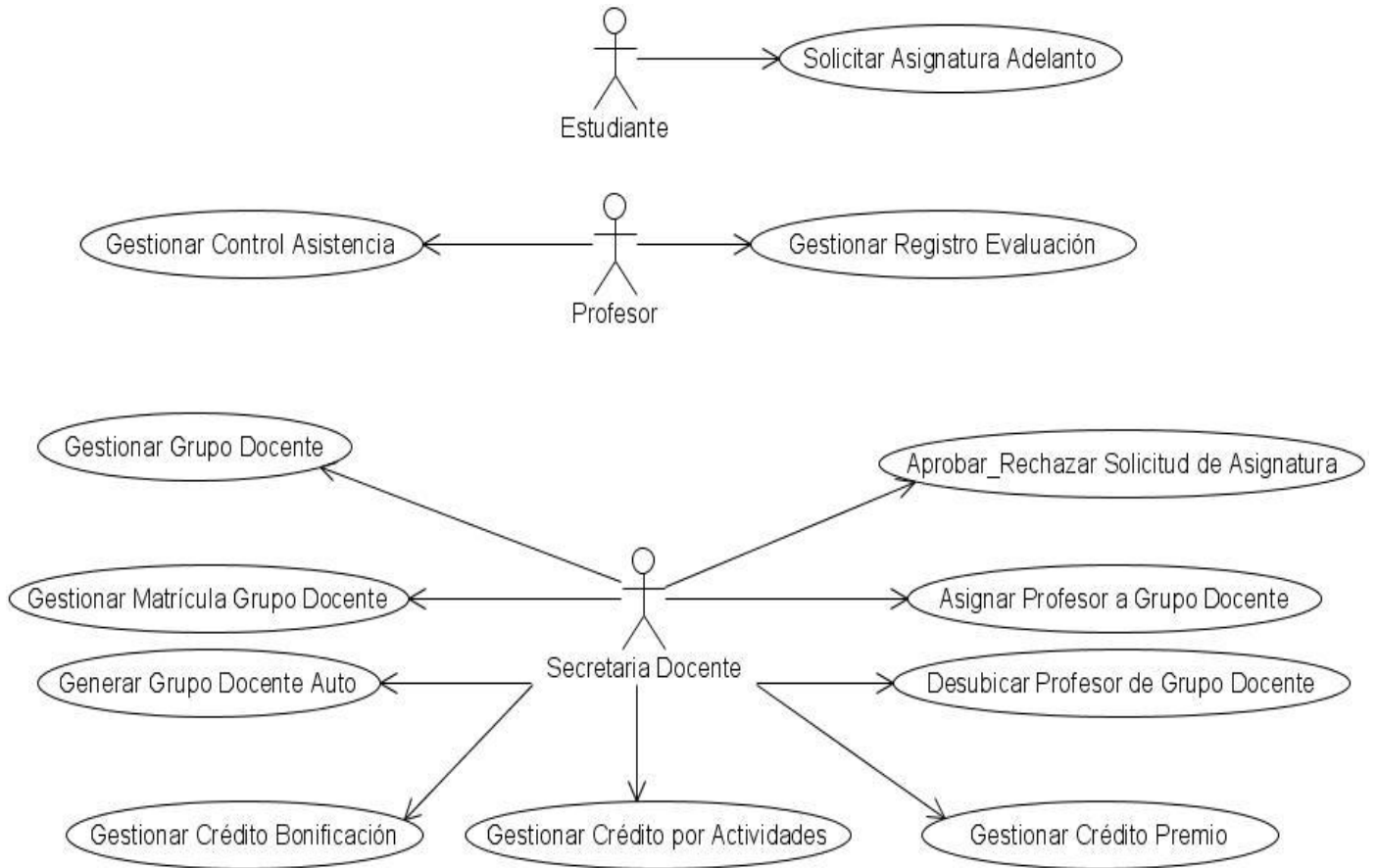


Figura 8. Diagrama de casos de uso. Paquete “Gestionar Información”

El paquete “Generar Información” contiene los siguientes casos de uso:

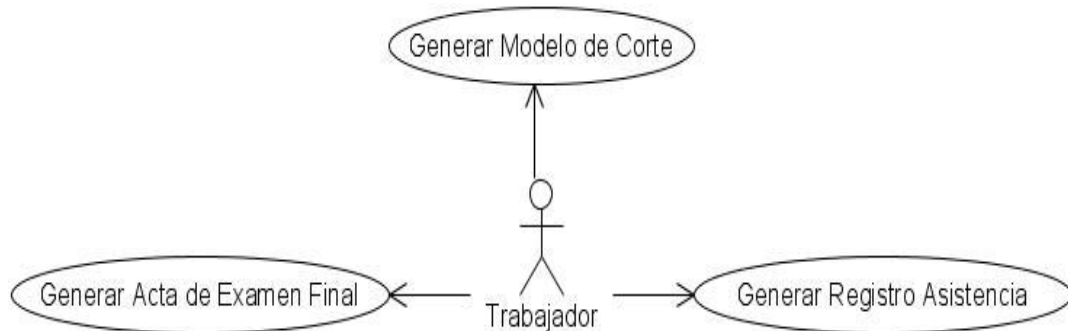


Figura 9. Diagrama de casos de uso. Paquete “Generar Información”

El paquete “Consultar Información” contiene los siguientes casos de uso:

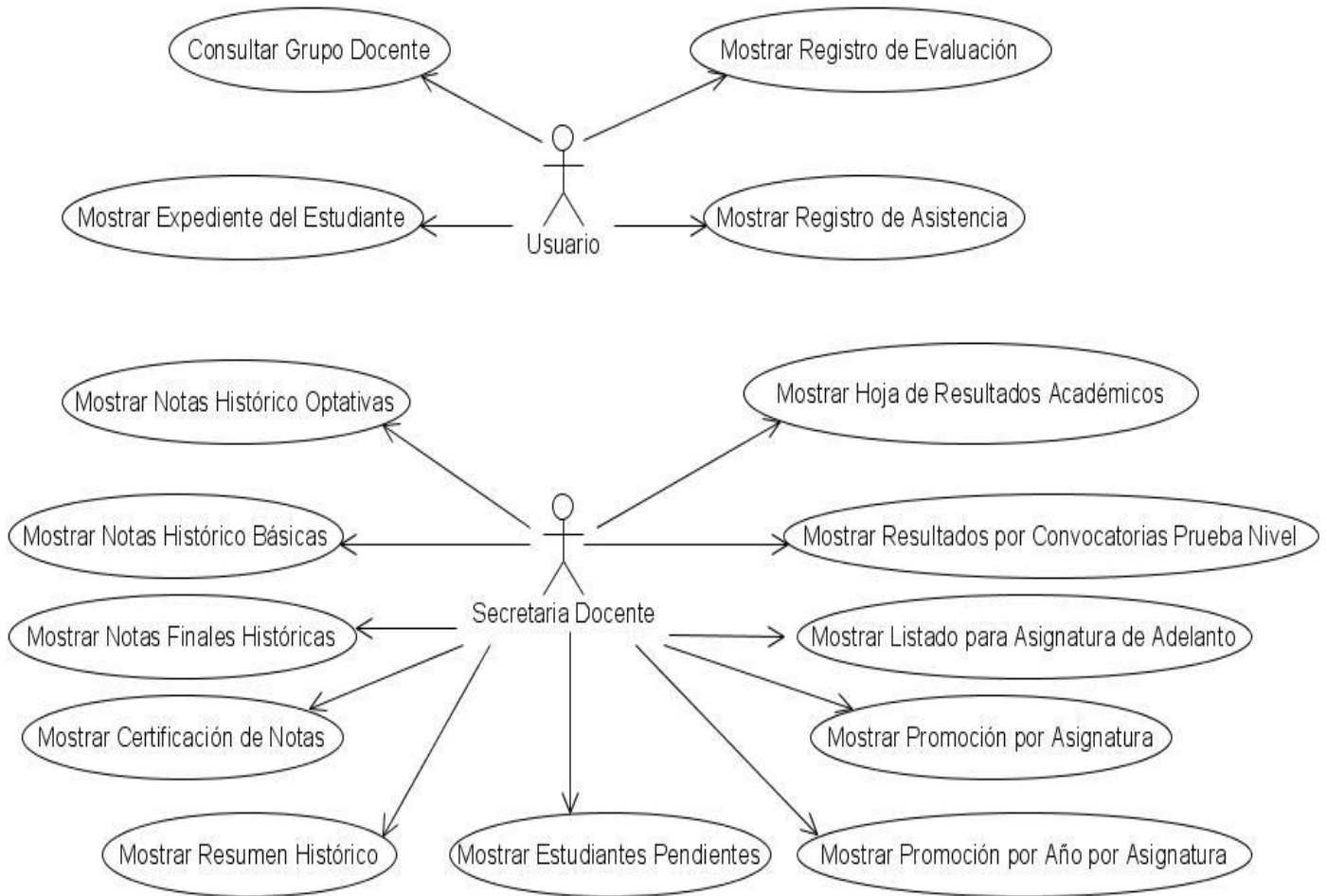


Figura 10. Diagrama de casos de uso. Paquete “Consultar Información”

“Para priorizar los casos de uso, es necesario determinar cuáles casos de uso son necesarios para el desarrollo en las primeras iteraciones y cuáles pueden dejarse para más tarde” [9]. Debido a la gran cantidad de procesos involucrados en el sistema y a su complejidad operacional, se determinaron para desarrollar en una primera versión los siguientes casos de uso críticos, que constituyen la vista de casos de uso arquitectónicamente significativos. (Ver figura 11)

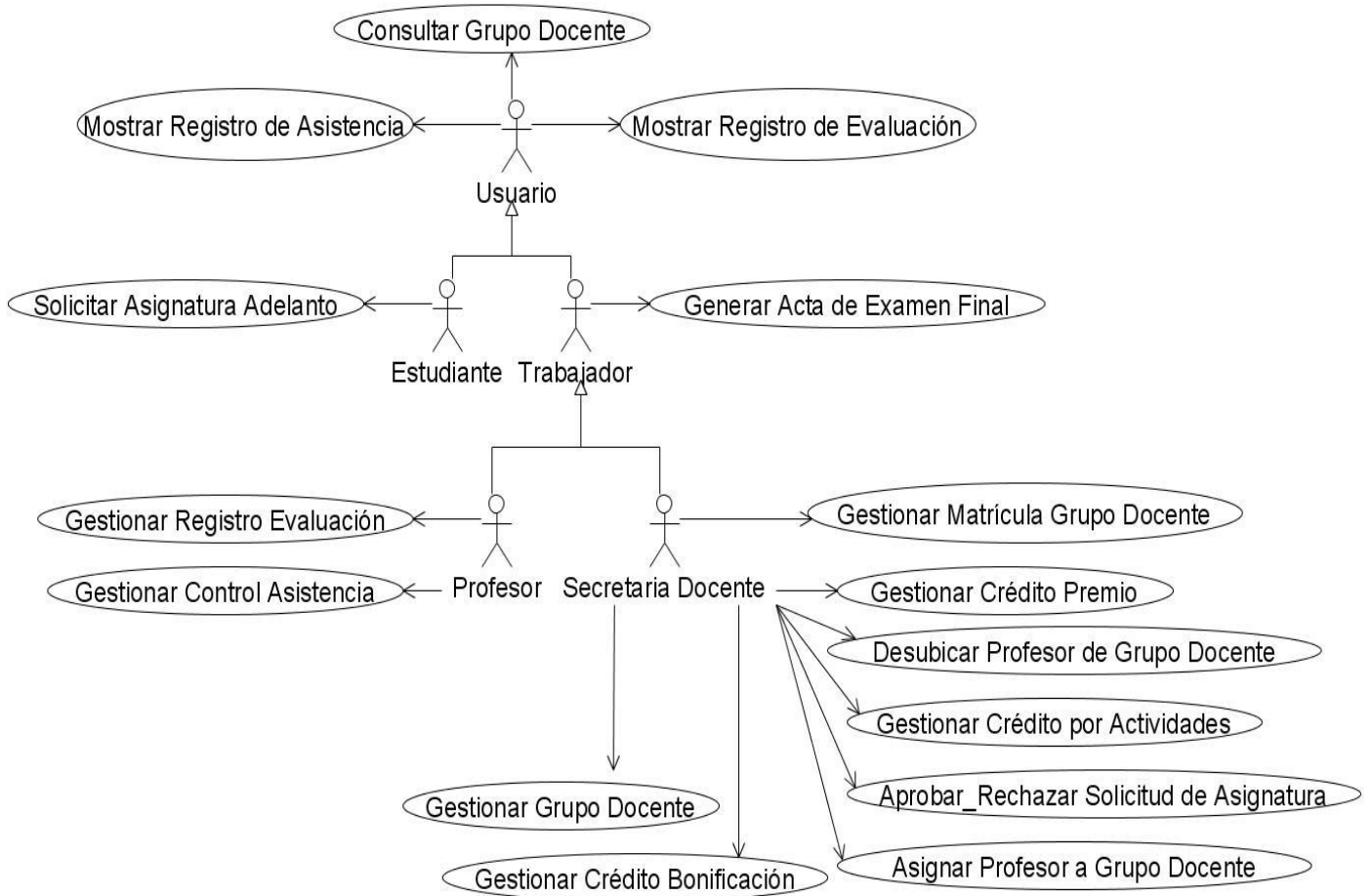


Figura 11. Vista de casos de uso arquitectónicamente significativos

2.5.5 Descripciones abreviadas de los casos de uso del sistema

La descripción de los casos de uso del sistema permite comprender el comportamiento del sistema desde el punto de vista del usuario. A continuación se muestran las descripciones abreviadas de los casos de uso del sistema, las descripciones detalladas pueden ser encontradas en el Anexo 4.

Tabla 2.11. Resumen del caso de uso “Gestionar Grupo Docente”

CUS-1	Gestionar Grupo Docente
Actor	Secretaria Docente
Descripción	El caso de uso se inicia cuando la secretaria docente necesita realizar una operación sobre determinado grupo docente. El sistema permite seleccionar las opciones crear, modificar o

	eliminar un grupo docente. El caso de uso finaliza al concluir las acciones correspondientes a la opción seleccionada.
Referencia	RF1, RF2, RF3

Tabla 2.12. Resumen del caso de uso “Gestionar Matrícula en Grupo Docente”

CUS-2	Gestionar Matrícula en Grupo Docente
Actor	Secretaria Docente
Descripción	El caso de uso se inicia cuando la secretaria docente decide matricular uno o varios estudiantes en un grupo docente, cambiar estudiantes de grupo docente o desmatricular estudiantes de un grupo docente. El sistema permite realizar la opción seleccionada finalizando el caso de uso.
Referencia	RF4, RF5

Tabla 2.13. Resumen del caso de uso “Gestionar Control Asistencia”

CUS-3	Gestionar Control Asistencia
Actor	Profesor
Descripción	El caso de uso se inicia cuando el profesor decide pasar la asistencia del grupo docente al que imparte clases. Para ello se dirige al registro de asistencia, donde puede realizar las acciones de pasar asistencia, modificar asistencia y eliminar asistencia. El caso de uso finaliza cuando realice las acciones que desea y salga del sistema.
Referencia	RF10, RF11, RF12

Tabla 2.14. Resumen del caso de uso “Gestionar Registro Evaluación”

CUS-4	Gestionar Registro Evaluación
Actor	Profesor
Descripción	El caso de uso se inicia cuando el profesor decide gestionar una evaluación, el profesor puede agregar evaluaciones al registro y modificarlas. El sistema muestra el registro de evaluaciones y el caso de uso finaliza cuando realice las acciones que desea y salga del sistema.
Referencia	RF14, RF15

Tabla 2.15. Resumen del caso de uso “Gestionar Crédito Premio”

CUS-5	Gestionar Crédito Premio
Actor	Secretaria Docente
Descripción	El caso de uso se inicia cuando la secretaria docente accede a realizar una operación de acreditar, modificar o eliminar los premios de un estudiante en una asignatura. El sistema ejecuta la acción indicada por la secretaria docente finalizando el caso de uso.
Referencia	RF17, RF18, RF19

Tabla 2.16. Resumen del caso de uso “Gestionar Crédito Bonificación”

CUS-6	Gestionar Crédito Bonificación
Actor	Secretaria Docente
Descripción	El caso de uso se inicia cuando la secretaria docente accede a realizar una operación de acreditar, modificar o eliminar las bonificaciones de un estudiante. El sistema ejecuta la acción indicada por la secretaria docente finalizando el caso de uso.
Referencia	RF20, RF21, RF22

Tabla 2.17. Resumen del caso de uso “Asignar Profesor a Grupo Docente”

CUS-7	Asignar Profesor a Grupo Docente
Actor	Secretaria Docente
Descripción	El caso de uso se inicia cuando la secretaria docente accede a realizar la operación de asignar un profesor a un grupo docente dada una asignatura y un departamento. Si el grupo docente ya le fue asignado al profesor, se muestra un mensaje de notificación y no se le asigna. En caso contrario, el sistema ejecuta la acción indicada terminando el caso de uso.
Referencia	RF23

Tabla 2.18. Resumen del caso de uso “Desubicar Profesor de Grupo Docente”

CUS-8	Desubicar Profesor de Grupo Docente
Actor	Secretaria Docente
Descripción	El caso de uso se inicia cuando la secretaria docente accede a realizar la operación de eliminar a un profesor de un grupo docente. El sistema ejecuta la acción indicada por la secretaria docente terminando el caso de uso.
Referencia	RF42

Tabla 2.19. Resumen del caso de uso “Gestionar Crédito por Actividades”

CUS-9	Gestionar Crédito por Actividades
Actor	Secretaria Docente
Descripción	El caso de uso se inicia cuando la secretaria docente accede a realizar una operación de acreditar, modificar o eliminar los créditos por actividades de un estudiante. El sistema ejecuta la acción indicada por la secretaria docente finalizando el caso de uso.
Referencia	RF27, RF28, RF29

Tabla 2.20. Resumen del caso de uso “Consultar Grupo Docente”

CUS-10	Consultar Grupo Docente
Actor	Usuario
Descripción	El caso de uso se inicia cuando la secretaria docente, el profesor o el estudiante acceden a realizar la operación de buscar los grupos docentes dada una asignatura y ver su matrícula. El sistema muestra los grupos docentes finalizando el caso de uso.
Referencia	RF6

Tabla 2.21. Resumen del caso de uso “Mostrar Registro de Asistencia”

CUS-11	Mostrar Registro de Asistencia
Actor	Usuario
Descripción	El caso de uso se inicia cuando la secretaria docente, el profesor o el estudiante acceden a realizar la operación de ver el registro de asistencia de un grupo docente. El sistema muestra el registro de asistencia y finaliza el caso de uso.
Referencia	RF13

Tabla 2.22. Resumen del caso de uso “Mostrar Registro de Evaluación”

CUS-12	Mostrar Registro de Evaluación
Actor	Usuario
Descripción	El caso de uso se inicia cuando la secretaria docente, el profesor o el estudiante acceden a realizar la operación de ver el registro de evaluación de un grupo docente. El sistema muestra el registro de evaluaciones y finaliza el caso de uso.
Referencia	RF16

Tabla 2.23. Resumen del caso de uso “Generar Acta de Examen Final”

CUS-13	Generar Acta de Examen Final
Actor	Trabajador
Descripción	El caso de uso se inicia cuando la secretaria docente o el profesor acceden a realizar la operación de crear el Acta de Examen Final de una asignatura. El sistema genera el Acta de Examen Final con los integrantes del grupo docente finalizando el caso de uso.
Referencia	RF32

Tabla 2.24. Resumen del caso de uso “Solicitar Asignatura Adelanto”

CUS-14	Solicitar Asignatura Adelanto
Actor	Estudiante
Descripción	El caso de uso se inicia cuando el estudiante accede a realizar la operación de solicitar adelantar una o varias asignaturas básicas. El sistema muestra la lista de las asignaturas que de acuerdo a la precedencia y el curso académico el estudiante puede adelantar. El estudiante selecciona una o varias asignaturas terminando así el caso de uso.
Referencia	RF7

Tabla 2.25. Resumen del caso de uso “Aprobar_Rechazar Solicitud de Asignatura”

CUS-15	Aprobar_Rechazar Solicitud de Asignatura
Actor	Secretaria Docente
Descripción	El caso de uso se inicia cuando la secretaria docente accede a realizar la operación de aprobar o rechazar la(s) solicitud(es) de las asignaturas de adelanto. El sistema muestra el listado de las solicitudes y la secretaria docente ejecuta la acción indicada finalizando el caso de uso.
Referencia	RF8, RF9

2.6 Conclusiones

Con la realización de este capítulo se inició el desarrollo de la propuesta de solución, se analizaron los procesos involucrados en el negocio, así como la obtención de un listado de funcionalidades que debe tener el sistema (expresados en los requisitos funcionales) donde se obtuvieron finalmente los casos de uso y los artefactos principales correspondientes a este flujo de trabajo tales como la especificación de

requisitos, diagramas de casos de uso y descripción de los casos de uso. Con la determinación de los casos de uso, se está en condiciones de comenzar la elaboración del mismo, específicamente, de realizar el análisis y diseño del sistema, partiendo siempre, por supuesto, de las funcionalidades consideradas en este capítulo.

Capítulo III: Elaboración de la solución propuesta

3.1 Introducción

En este capítulo se realiza una estimación del esfuerzo mediante el método de estimación por puntos de casos de uso a partir de las características de los requisitos, expresados en los casos de uso. Se modela el análisis y el diseño del Módulo Registro y Control Docente, los cuales son muy importantes en el desarrollo del software ya que constituye la vista lógica de la arquitectura. Por lo que se ajusta el resultado de estos modelos a las tecnologías y lenguajes que serán utilizados. Se muestran las realizaciones de los casos de uso críticos, definidos en el capítulo anterior y que van a ser incluidos en la primera iteración del sistema, mediante los diagramas de clases del análisis y los diagramas de clases del diseño, expresados en diagramas de clases Web.

3.2 Estimación del esfuerzo

“La estimación mediante el análisis de puntos de casos de uso es un método de estimación del tiempo de desarrollo de un proyecto mediante la asignación de "pesos" a un cierto número de factores que lo afectan, para finalmente, contabilizar el tiempo total estimado para el proyecto a partir de esos factores” [10]. Mediante esta técnica existe una probabilidad de estimar el esfuerzo que el equipo de desarrollo empleará para elaborar el sistema, a partir de las características de sus requisitos, expresados en los casos de uso, por lo que es decisión de la universidad por las características de los proyectos la utilización de esta técnica. A continuación, se realiza la estimación para la primera iteración del desarrollo donde se detallan los pasos a seguir para la aplicación de este método.

El primer paso para la estimación consiste en el cálculo de los puntos de casos de uso sin ajustar.

Se calcula a partir de la siguiente ecuación:

$UUCP = UAW + UUCW$ donde,

UUCP: puntos de casos de uso sin ajustar

UAW: factor de peso de los actores sin ajustar

UUCW: factor de peso de los casos de uso sin ajustar

Factor de peso de los actores sin ajustar

“Este valor se calcula mediante un análisis de la cantidad de actores presentes en el sistema y la complejidad de cada uno de ellos. La complejidad de los actores se establece teniendo en cuenta en primer lugar si se trata de una persona o de otro sistema y en segundo lugar la forma en la que el actor interactúa con el sistema” [10].

Tabla 3.1. Factor de peso de los actores sin ajustar

Actores	Descripción	Complejidad	Factor de Peso
Usuario	Una persona que interactúa con el sistema mediante una interfaz gráfica.	Complejo	3
Estudiante			
Trabajador			
Profesor			
Secretaria Docente			

UAW = Sumatoria de la multiplicación de la cantidad de actores de un tipo por su factor de peso.

Cantidad de actores de tipo complejo: 5

$$UAW = 5 * 3$$

$$UAW = 15$$

Factor de peso de los casos de uso sin ajustar

“Este valor se calcula mediante un análisis de la cantidad de casos de uso presentes en el sistema y la complejidad de cada uno de ellos. La complejidad de los casos de uso se establece teniendo en cuenta la cantidad de transacciones efectuadas en el mismo, donde una transacción se entiende como una secuencia de actividades atómica, es decir, se efectúa la secuencia de actividades completa, o no se efectúa ninguna de las actividades de la secuencia y está representada por uno o más pasos del flujo de eventos principal del caso de uso, pudiendo existir más de una transacción dentro del mismo caso de uso” [10].

Tabla 3.2. Factor de peso de los casos de uso sin ajustar

Caso de Uso	Descripción	Complejidad	Factor de Peso
CUS Gestionar Grupo Docente	Contiene 7 transacciones.	Medio	10
CUS Gestionar Matrícula en Grupo Docente	Contiene 5 transacciones.	Medio	10
CUS Gestionar Control Asistencia	Contiene 7 transacciones.	Medio	10
CUS Gestionar Registro Evaluación	Contiene 6 transacciones.	Medio	10
CUS Gestionar Crédito Premio	Contiene 7 transacciones.	Medio	10
CUS Gestionar Crédito Bonificación	Contiene 7 transacciones.	Medio	10
CUS Asignar Profesor a Grupo Docente	Contiene 2 transacciones.	Simple	5
CUS Desubicar Profesor de Grupo Docente	Contiene 2 transacciones.	Simple	5
CUS Gestionar Crédito por Actividades	Contiene 7 transacciones.	Medio	10
CUS Consultar Grupo Docente	Contiene 4 transacciones.	Medio	10
CUS Mostrar Registro de Asistencia	Contiene 2 transacciones.	Simple	5
CUS Mostrar Registro de Evaluación	Contiene 2 transacciones.	Simple	5
CUS Generar Acta de Examen Final	Contiene 2 transacciones.	Simple	5
CUS Solicitar Asignatura Adelanto	Contiene 2 transacciones.	Simple	5
CUS Aprobar_Rechazar Solicitud de Asignatura	Contiene 5 transacciones.	Medio	10

UUCW = Sumatoria de los casos de uso por su complejidad.

$$UUCW = 9 * 10 + 6 * 5$$

$$UUCW = 120$$

Finalmente, los puntos de casos de uso sin ajustar resultan

$$UUCP = UAW + UUCW$$

$$UUCP = 15 + 120$$

$$UUCP = 135$$

El segundo paso es calcular los puntos de casos de uso ajustados.

$$UCP = UUCP * TCF * EF \text{ donde,}$$

UCP: puntos de casos de uso ajustados

UUCP: puntos de casos de uso sin ajustar

TCF: factor de complejidad técnica

EF: factor de ambiente

Factor de complejidad técnica

Para calcular este coeficiente se analizan una serie de factores que pueden determinar la complejidad técnica del sistema, a estos factores se les asocia un valor de 0 a 5 que determina el vínculo del mismo con las características deseadas para el sistema.

Tabla 3.3. Factor de complejidad técnica

Factor	Descripción	Peso	Valor Asignado	Peso * Valor
T1	Sistema distribuido	2	0	0
T2	Objetivos de performance o tiempo de respuesta	1	4	4
T3	Eficiencia del usuario final	1	2	2
T4	Procesamiento interno complejo	1	3	3
T5	El código debe ser reutilizable	1	5	5
T6	Facilidad de instalación	0.5	4	2
T7	Facilidad de uso	0.5	3	1.5
T8	Portabilidad	2	4	8
T9	Facilidad de cambio	1	3	3
T10	Concurrencia	1	5	5
T11	Incluye objetivos especiales de seguridad	1	5	5
T12	Provee acceso directo a terceras partes	1	0	0

T13	Se requieren facilidades especiales de entrenamiento a usuarios	1	3	3
-----	---	---	---	---

El Factor de complejidad técnica resulta:

$$TCF = 0.6 + 0.01 * \Sigma (\text{Peso} * \text{Valor asignado})$$

$$TCF = 0.6 + 0.01 * (0+4+2+3+5+2+1.5+8+3+5+5+0+3)$$

$$TCF = 0.6 + 0.01 * 41.5$$

$$TCF = 1.015$$

Factor de ambiente

Para calcular este coeficiente se analizan una serie de factores que pueden determinar el tiempo requerido para el desarrollo del sistema, teniendo en cuenta aspectos como habilidades, conocimientos, etc. de los involucrados en la realización del sistema. A estos factores se les asocia un valor de 0 a 5 que determina el vínculo del mismo con las características deseadas para el sistema.

Tabla 3.4. Factor ambiente

Factor	Descripción	Peso	Valor Asignado	Valor * Peso
E1	Familiaridad con el modelo de proyecto utilizado	1.5	0	0
E2	Experiencia en la aplicación	0.5	3	1.5
E3	Experiencia en orientación a objetos	1	4	4
E4	Capacidad del analista líder	0.5	5	2.5
E5	Motivación	1	4	4
E6	Estabilidad de los requerimientos	2	2	4
E7	Personal part-time	-1	0	0
E8	Dificultad del lenguaje de programación	-1	2	-2

El Factor de ambiente resulta:

$$EF = 1.4 - 0.03 * \Sigma (\text{Peso} * \text{Valor asignado})$$

$$EF = 1.4 - 0.03 * \Sigma (0+1.5+4+2.5+4+4+0-2)$$

$$EF = 1.4 - 0.03 * 14$$

$$EF = 0.98$$

Finalmente, los puntos de casos de uso ajustados resultan:

$$UCP = UUCP * TCF * EF$$

$$UCP = 135 * 1.015 * 0.98$$

$$UCP = 134.2845$$

El esfuerzo en horas-hombre viene dado por:

$$E = UCP * CF \text{ donde,}$$

E: esfuerzo

UCP: puntos de casos de uso ajustados

CF: factor de conversión

Para calcular CF

$$\text{Total}_{EF} = \text{Cantidad EF} < 3 \text{ (entre E1 – E6)} + \text{Cantidad EF} > 3 \text{ (entre E7, E8)}$$

$$\text{Total}_{EF} = 2 + 0$$

$$\text{Total}_{EF} = 2$$

$$CF = 20 \text{ horas-hombre (si Total}_{EF} \leq 2)$$

$$CF = 28 \text{ horas-hombre (si Total}_{EF} = 3 \text{ ó Total}_{EF} = 4)$$

$$CF = \text{abandonar o cambiar proyecto (si Total}_{EF} \geq 5)$$

Para este tipo de proyecto 20 horas-hombre/punto de casos de uso, es decir, un punto de caso de uso toma 20 horas-hombre.

$$E = UCP * CF$$

$$E = 134.2845 * 20$$

$$E = 2865.69 \text{ horas-hombres}$$

Se considera que este esfuerzo representa un porcentaje del esfuerzo del flujo de trabajo de implementación. Para una estimación más completa de la duración total del proyecto hay que agregar a la estimación del esfuerzo obtenida por los puntos de casos de uso, las estimaciones de esfuerzo de las demás actividades relacionadas con el desarrollo de software. Teniendo en cuenta los siguientes valores

porcentuales para la distribución del esfuerzo entre las diferentes actividades de un proyecto, que estadísticamente se considera aceptable, se obtiene:

Tabla 3.5. Distribución del esfuerzo

Actividad	Porcentaje	Horas-Hombre
Análisis	10 %	716.4225
Diseño	20 %	1432.845
Implementación	40 %	2865.69
Pruebas	15 %	1074.63375
Sobrecarga (otras actividades)	15 %	1074.63375
Total	100 %	7164.225

Esfuerzo Total (horas-hombres) 7164.225.

Para la etapa de análisis y diseño se requiere de un esfuerzo de 2149.2675 horas-hombres, si se considera que trabajan dos personas, 48 horas como promedio en la semana, esta etapa debe terminarse en aproximadamente 22 semanas.

3.3 Análisis

“Durante el análisis, se analizan los requisitos que se describen en la captura de requerimientos, refinándolos y estructurándolos. El objetivo de hacerlo es conseguir una comprensión más precisa de los requisitos y una descripción de los mismos que sea fácil de mantener y que ayude a estructurar el sistema entero, incluyendo su arquitectura” [11].

El análisis abarca las abstracciones primarias de clases y objetos, así como los mecanismos presentes en el dominio del problema, o sea del mundo real, no se consideran clases que definen detalles y soluciones en el sistema de software. Las clases que se modelan refinan los requisitos funcionales obtenidos anteriormente, son identificadas con sus relaciones y descritas en un diagrama de clases utilizando los siguientes estereotipos:

- Clase interfaz: modelan la interacción entre el sistema y sus actores.
- Clase controladoras: coordinan la realización de uno o unos pocos casos de uso, relacionando las actividades de los objetos que implementan sus funcionalidades.

- Clases entidad: modelan información que posee larga vida y que es a menudo persistente.

3.3.1 Diagramas de clases del análisis

En los diagramas de clases del análisis se representan los conceptos en un dominio del problema, representa las cosas del mundo real, no de la implementación automatizada. En el análisis se usan los diagramas de colaboración, ya que el objetivo principal es identificar las funcionalidades de cada objeto y las responsabilidades sobre ellos. En el Anexo 5 se pueden consultar los diagramas de colaboración para cada uno de los escenarios de los casos de usos. A continuación se representan los diagramas de clases del análisis, por cada realización de caso de uso del sistema, para su mayor comprensión.

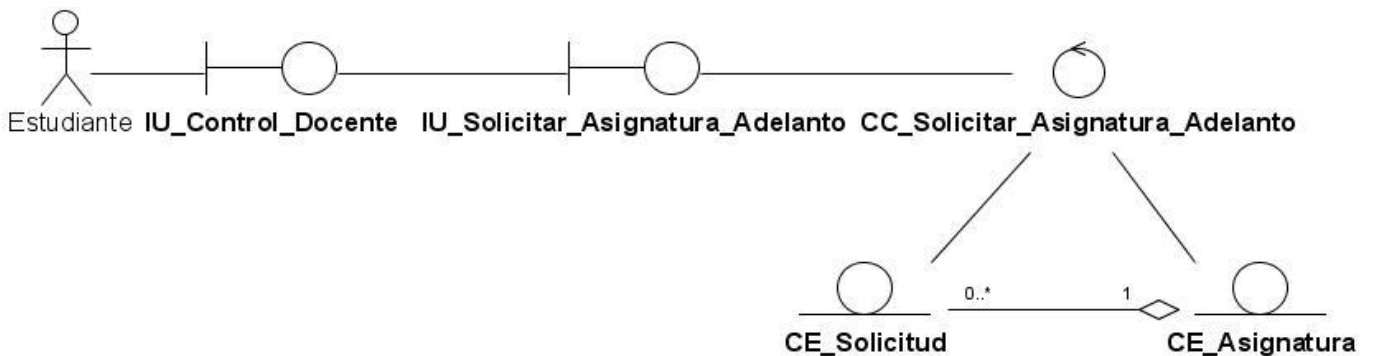


Figura 12. Diagrama de clases del análisis: CUS “Solicitar Asignatura Adelanto”

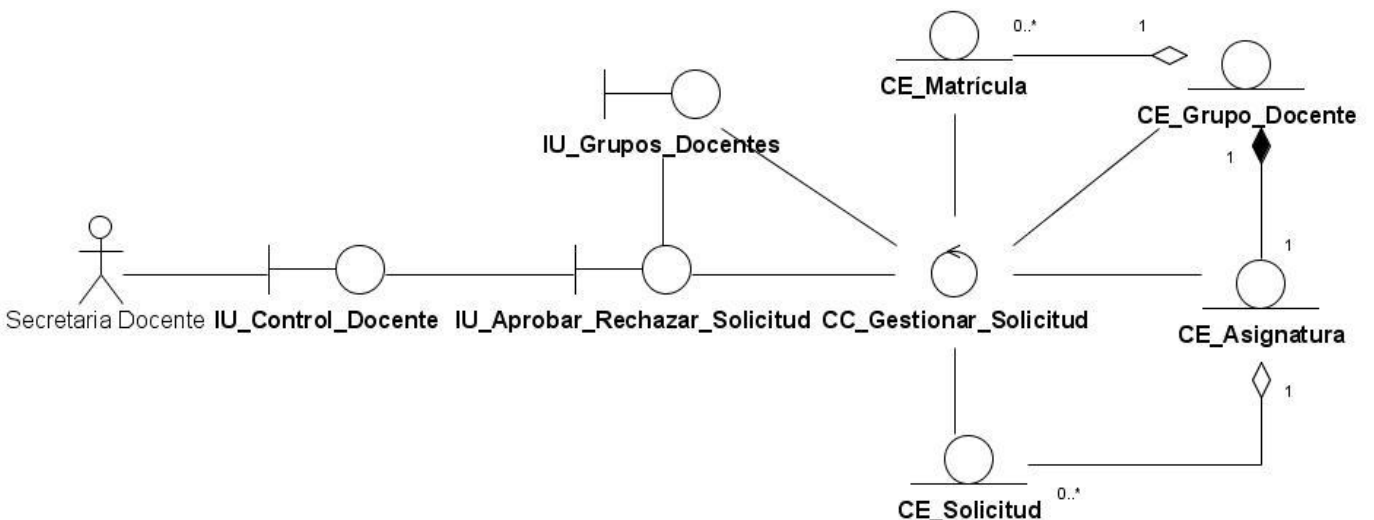


Figura 13. Diagrama de clases del análisis: CUS “Aprobar_Rechazar Solicitud de Asignatura”

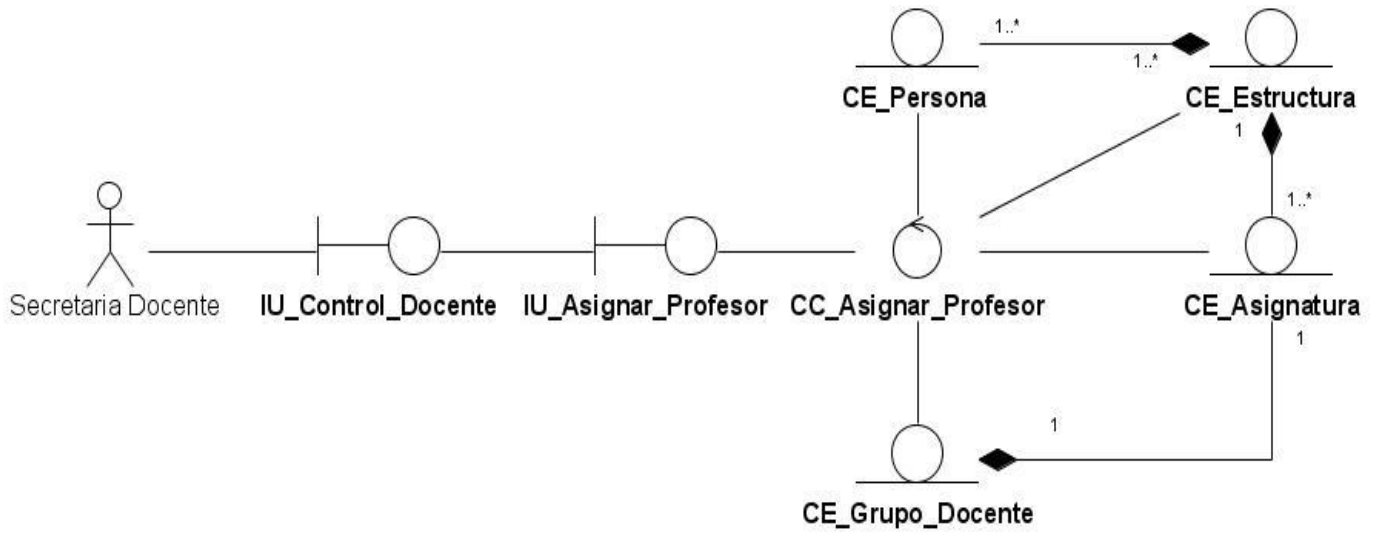


Figura 14. Diagrama de clases del análisis: CUS “Asignar Profesor a Grupo Docente”

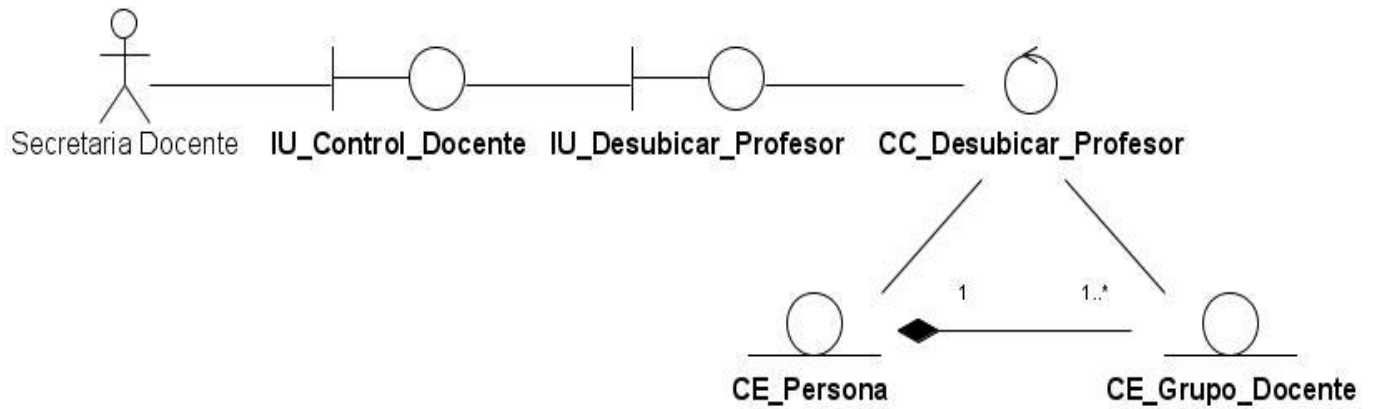


Figura 15. Diagrama de clases del análisis: CUS “Desubicar Profesor de Grupo Docente”

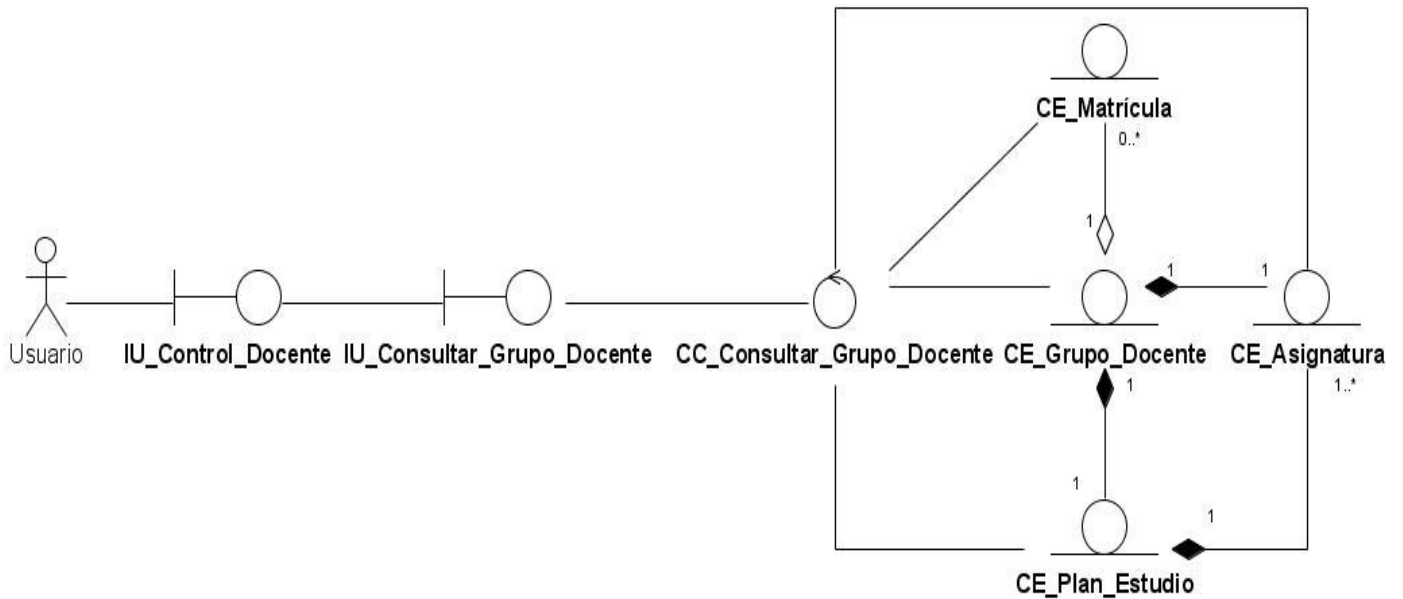


Figura 16. Diagrama de clases del análisis: CUS “Consultar Grupo Docente”

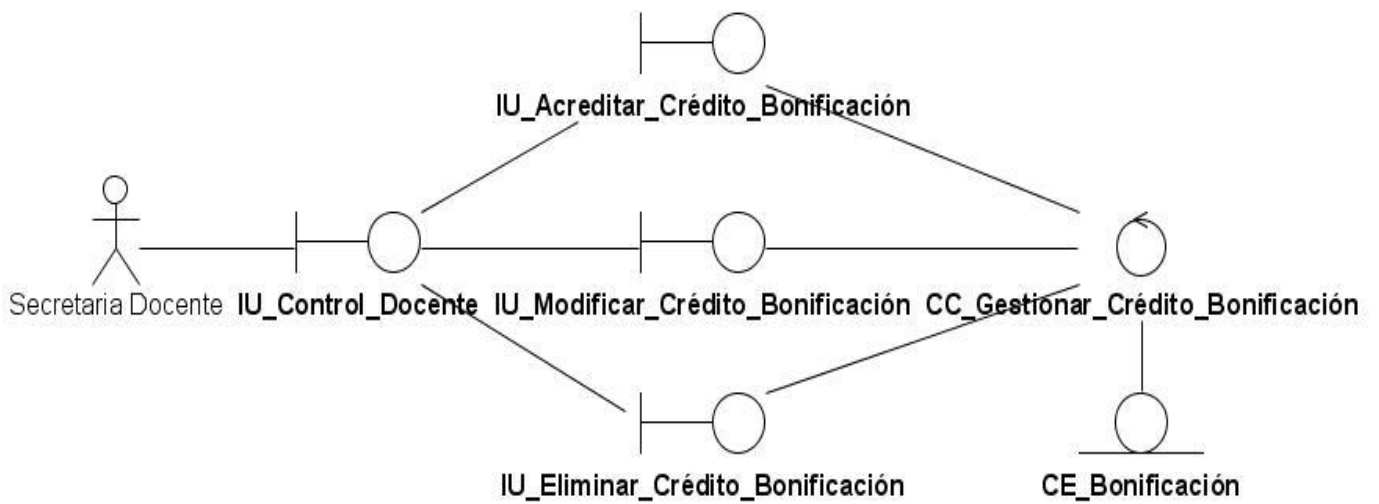


Figura 17. Diagrama de clases del análisis: CUS “Gestionar Crédito Bonificación”

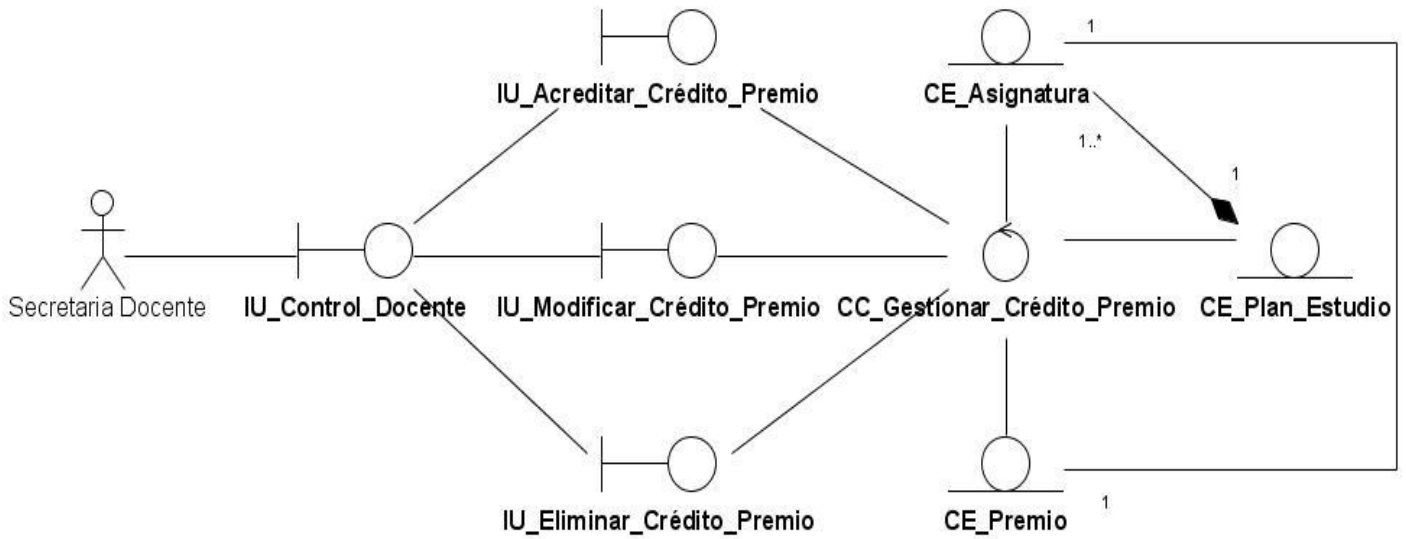


Figura 18. Diagrama de clases del análisis: CUS “Gestionar Crédito Premio”

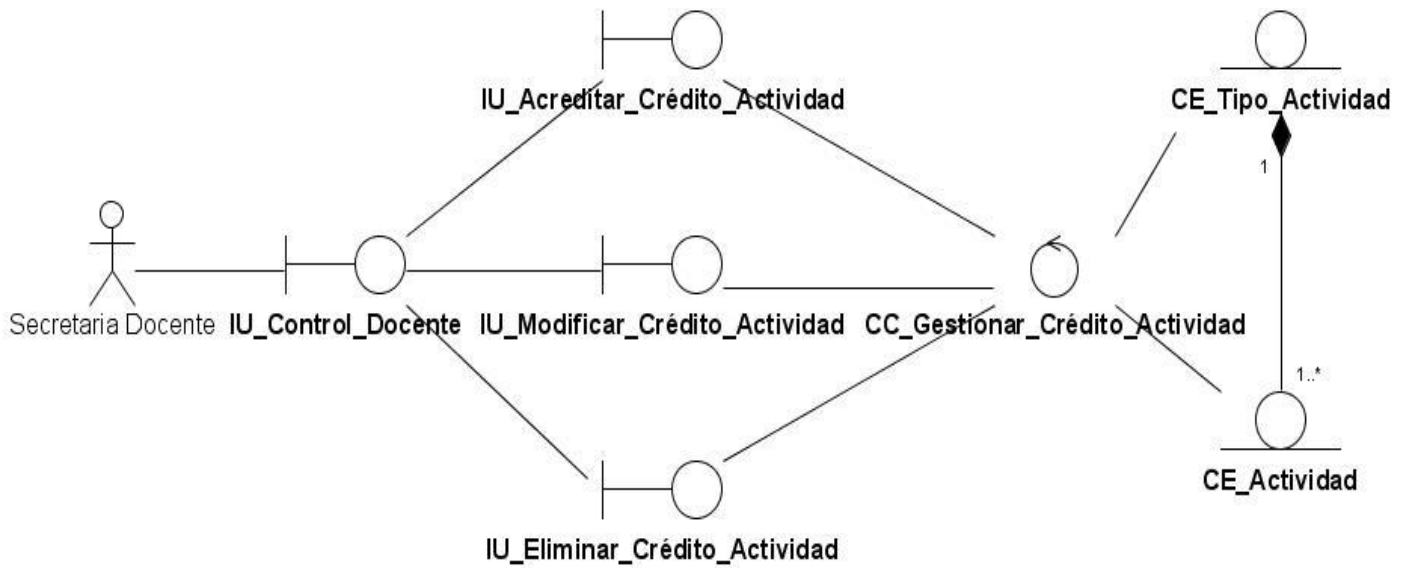


Figura 19. Diagrama de clases del análisis: CUS “Gestionar Crédito por Actividades”

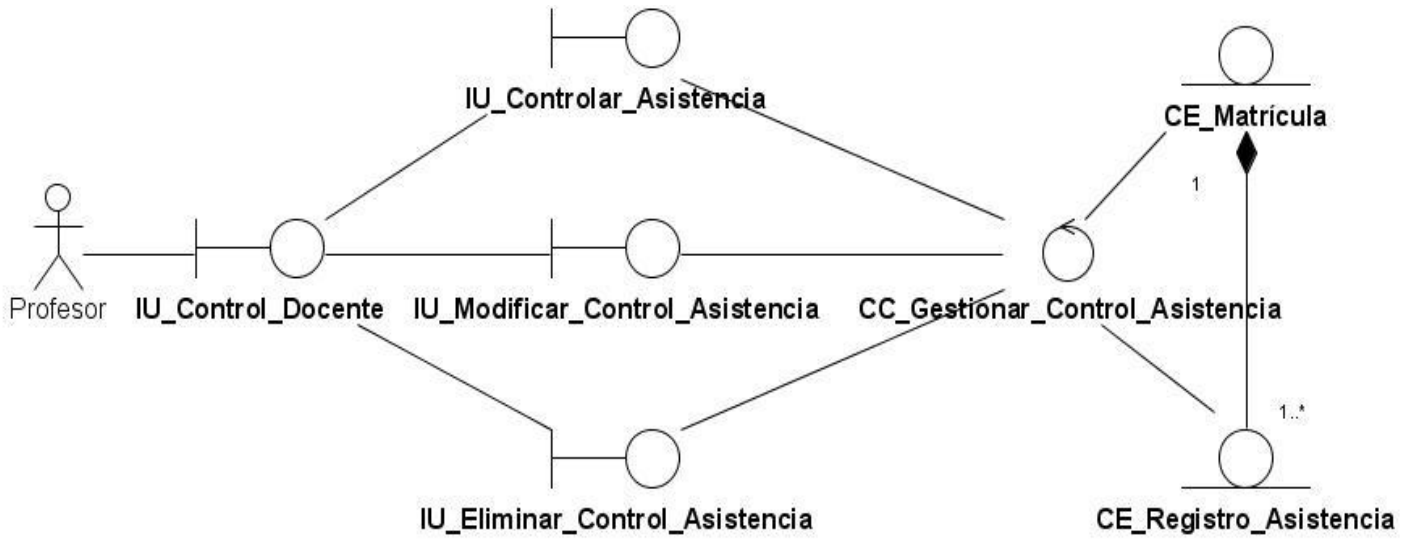


Figura 20. Diagrama de clases del análisis: CUS “Gestionar Control Asistencia”

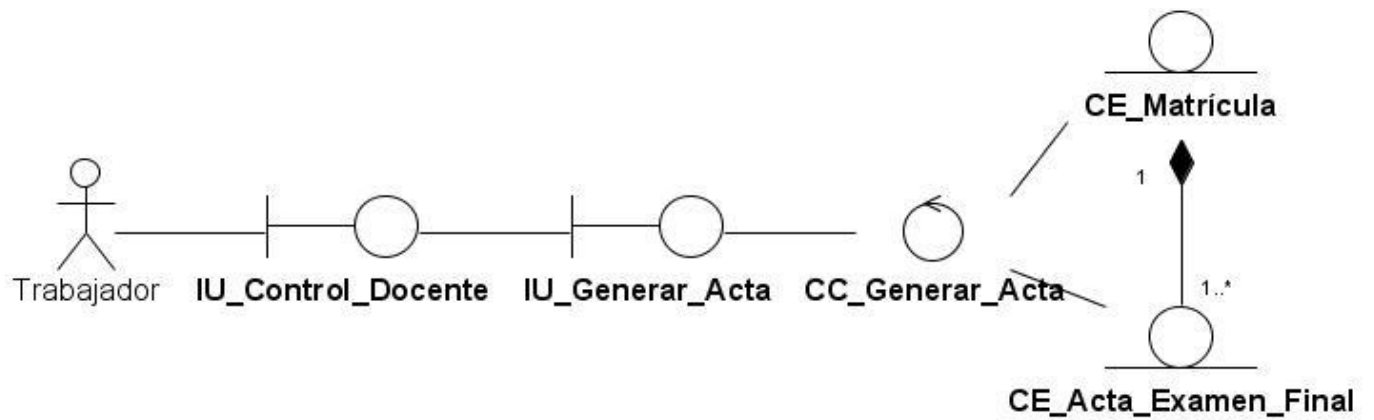


Figura 21. Diagrama de clases del análisis: CUS “Generar Acta de Examen Final”

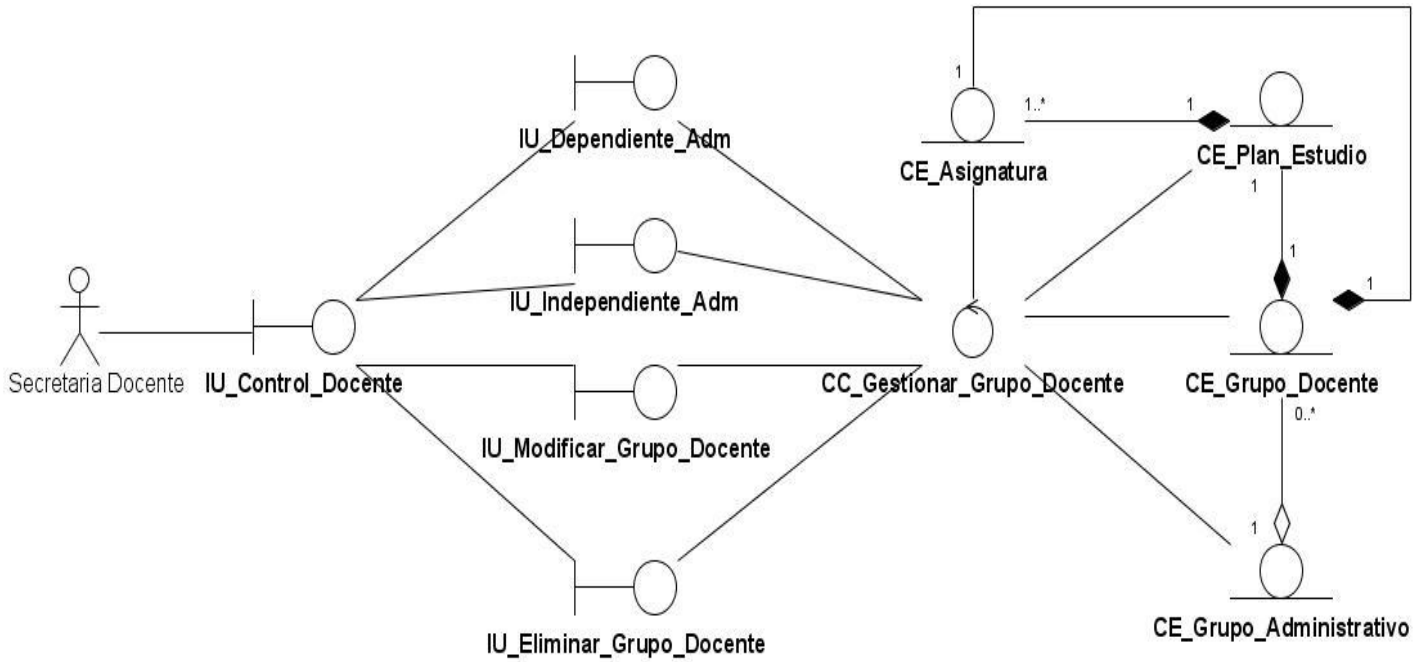


Figura 22. Diagrama de clases del análisis: CUS “Gestionar Grupo Docente”

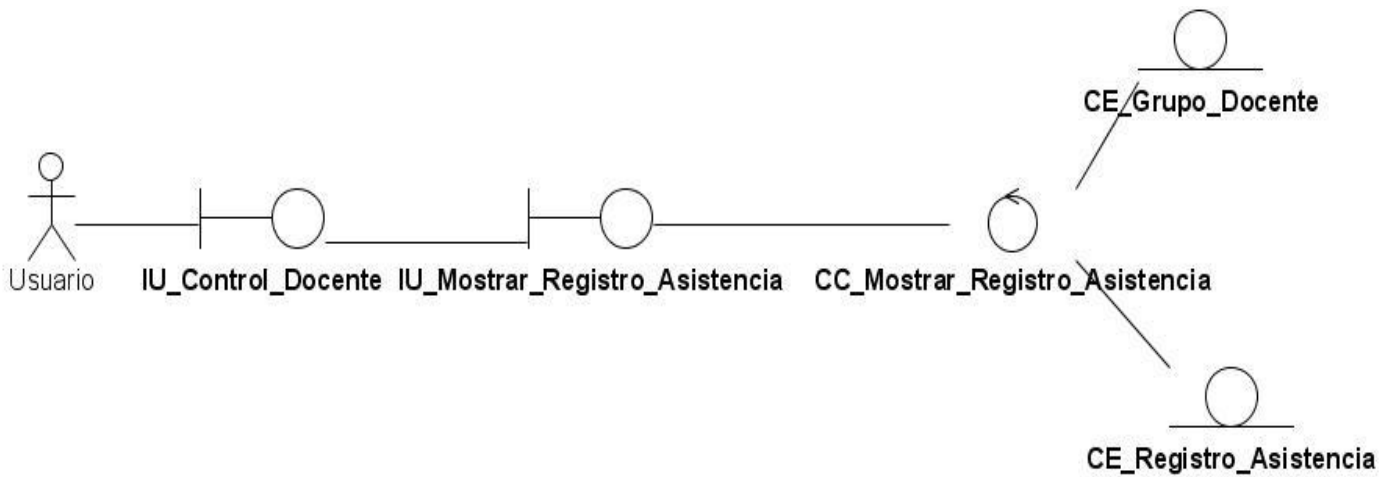


Figura 23. Diagrama de clases del análisis: CUS “Mostrar Registro de Asistencia”

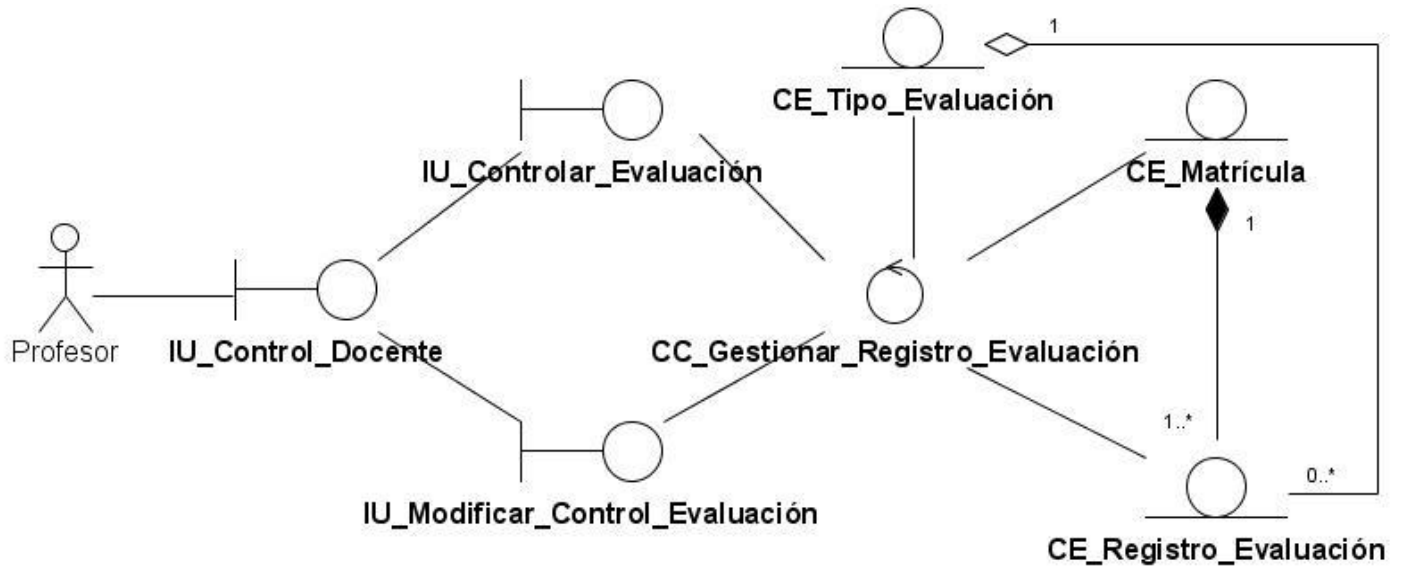


Figura 24. Diagrama de clases del análisis: CUS “Gestionar Registro Evaluación”

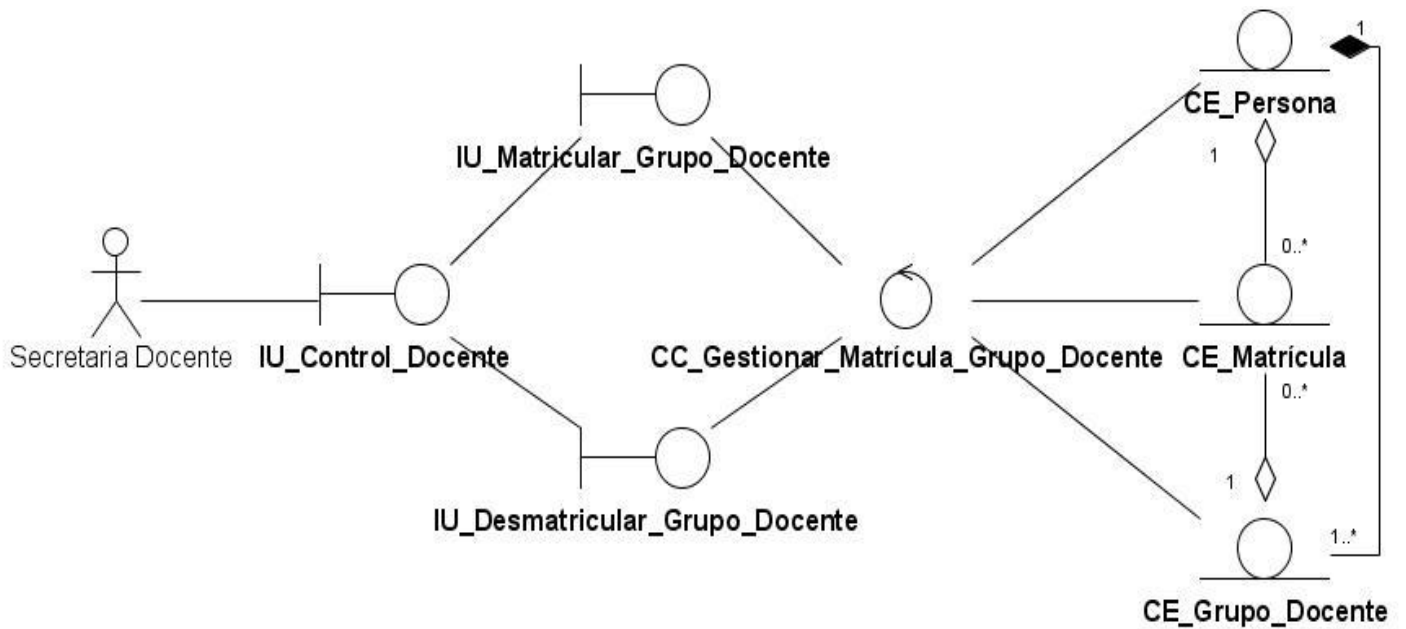


Figura 25. Diagrama de clases del análisis: CUS “Gestionar Matrícula en Grupo Docente”

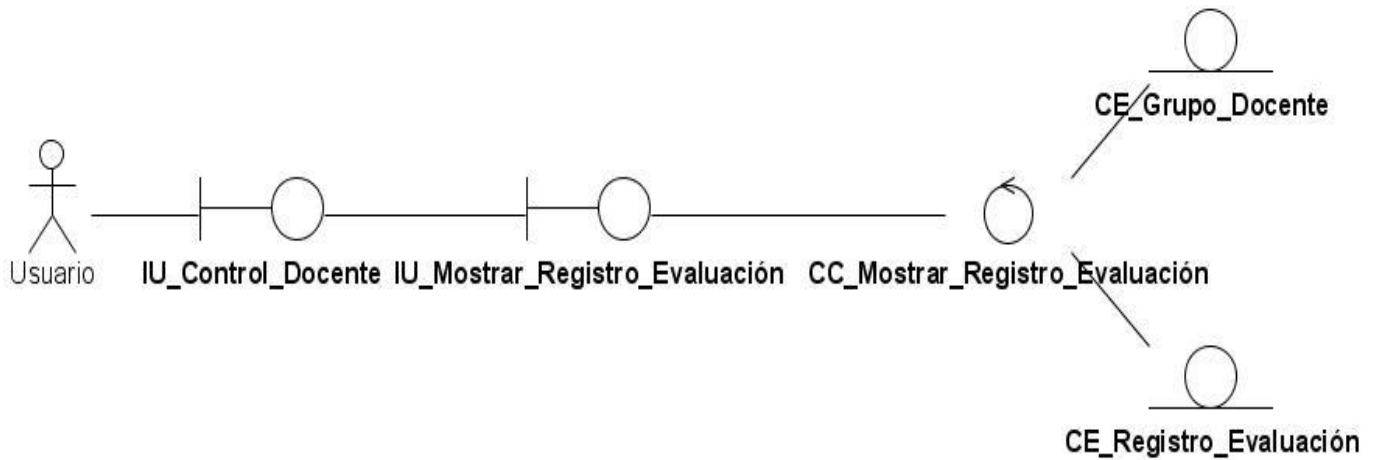


Figura 26. Diagrama de clases del análisis: CUS “Mostrar Registro de Evaluación”

3.4 Diseño

En el diseño el resultado del análisis es expandido a una solución técnica ya que se agregan nuevas clases que proveen de la infraestructura técnica: interfaces de usuario, manejo de bases de datos para almacenar objetos, comunicaciones con otros sistemas, etc. El diseño resulta en especificaciones detalladas para la fase de programación. Se modela el sistema de manera que soporte todos los requerimientos, incluyendo a diferencia del análisis, los requerimientos no funcionales.

3.4.1 Patrones de diseño

Los patrones son soluciones a problemas recurrentes en el entorno de desarrollo. Se pueden considerar como recetas para solucionar varias veces un problema del mismo tipo. Los desarrolladores lo usan como una forma de reutilizar la experiencia, clasificando las soluciones con términos de común denominación.

En el diseño de la propuesta de solución se tiene en cuenta un patrón clásico del diseño conocido como arquitectura MVC que implementa Zend Framework. MVC está formado por tres niveles:

- El modelo representa la información con la que trabaja la aplicación, es decir, su lógica de negocio. Se encarga de la abstracción de la lógica relacionada con los datos, haciendo que la vista y las acciones sean independientes de, por ejemplo, el tipo de gestor de bases de datos utilizado por la aplicación.

- La vista transforma el modelo en una página Web que permite al usuario interactuar con ella.
- El controlador se encarga de procesar las interacciones del usuario y realiza los cambios apropiados en el modelo o en la vista. Mantiene aislado al modelo y a la vista de los detalles del protocolo utilizado para las peticiones.

En la realización del diseño se utilizaron además patrones GRASP, los cuales describen los principios fundamentales de la asignación de responsabilidades a objetos, patrones GOF y patrones asociados al modelo.

Patrones GRASP utilizados

Experto: este patrón se tiene en cuenta para la asignación de responsabilidades a las clases de forma tal que las mismas contengan la información necesaria para poder ejecutar una acción específica, o sea, el controlador frontal le asigna responsabilidades a las clases controladoras para realizar acciones de acuerdo a las peticiones que recibe. El uso de este patrón permitirá a los objetos valerse de su propia información para hacer lo que se les pide, favorece la existencia de mínimas relaciones entre las clases, lo que permite contar con un sistema robusto y fácil de mantener.

Creador: este patrón se tiene en cuenta para la asignación de responsabilidades a las clases relacionadas con la creación de objetos, de forma tal que una instancia de un objeto sólo pueda ser creada por el objeto que contiene la información necesaria para ello. Existe un único *script* PHP que crea una única instancia del controlador frontal, dicho controlador frontal sólo crea instancias de las clases controladoras y estas, a su vez, sólo crean objetos de la clase *Zend_View*. El uso de este patrón permite crear las dependencias mínimas necesarias entre las clases, lo que favorece al mantenimiento del sistema y ofrece mejores oportunidades de reutilización.

Alta cohesión: este patrón se tiene en cuenta para realizar un diseño que evite contener clases con un alto grado de abstracción, que asuman responsabilidades que podían haber delegado a otros objetos o que tengan responsabilidades muy complejas. Se diseñaron las clases de forma tal que contengan las mínimas responsabilidades necesarias y colaboren con otras para llevar a cabo una tarea. Se tienen las clases controladoras que se encargan de ejecutar acciones de acuerdo a las peticiones que le llegan y las clases de acceso a datos que interactúan con el modelo, de forma tal que se elimina la sobrecarga de

funcionalidades en las clases controladoras. Este patrón permitirá tener clases fáciles de mantener, entender y reutilizar.

Controlador: este patrón se tiene en cuenta para realizar las asignaciones en cuanto al manejo de los eventos del sistema y definir sus operaciones. Zend Framework contribuye a la utilización de este patrón ya que define un controlador frontal que implica que todas las solicitudes son dirigidas a un único *script* PHP que se encarga de instanciarlo y redirigir las llamadas.

Bajo Acoplamiento: el acoplamiento mide la fuerza con que una clase está conectada a otra, de esta forma una clase con bajo acoplamiento debe tener un número mínimo de dependencia con otras clases. Las diferentes clases controladoras sólo dependen de un único controlador frontal para realizar sus funcionalidades. Además, una clase de acceso a datos sólo depende de la clase controladora que la usa, así se pueden realizar cambios en cada clase de forma independiente. El uso de los patrones Experto y Creador contribuyen al bajo acoplamiento entre las clases del sistema. Este patrón se tuvo en cuenta por la importancia que significa realizar un diseño de clases independientes que puedan soportar los cambios de una manera fácil y permitan la reutilización.

Patrones GOF utilizados

Decorator: este patrón permite añadir funcionalidad a una clase dinámicamente. Zend Framework implementa este patrón en la clase `Zend_View`, encargada de asignarle responsabilidades a objetos de manera dinámica y configurarlos con nuevos atributos.

Singleton: garantiza la existencia de una única instancia para una clase y la creación de un mecanismo de acceso global a dicha instancia. Zend Framework posee una instancia única del controlador frontal disponible mediante este patrón para lograr una vía de entrada única a las solicitudes.

Facade: este patrón permite utilizar una interfaz común para un conjunto de interfaces del módulo, haciendo que el mismo sea más fácil de usar.

Factory: proporciona una interfaz para la creación de objetos interdependientes o interrelacionados, sin especificar sus clases concretas. Zend Framework utiliza este patrón en la implementación de un conjunto de clases para el acceso a datos.

Patrones del modelo utilizados

Zend Framework provee un API para el acceso a datos conformado por un conjunto de clases que implementan los patrones *Table Data Gateway* y *Row Data Gateway*. Estos patrones convierten el modelo generado en una estructura altamente flexible que presenta, entre otras ventajas, una mayor escalabilidad y portabilidad.

Row Data Gateway (RDG) asume el comportamiento de un objeto que actúa como puerta de enlace a una fila de una tabla en la base de datos; brinda objetos que representan registros de una fila en alguna tabla de la base de datos, pero tiene la particularidad de facilitar el acceso a dichos objetos con los mecanismos regulares de programación del lenguaje utilizado. Cada columna en la base de datos, al emplear este patrón, se convierte en un campo que actúa como interfaz por cada registro de datos. Por otra parte, es posible acudir a métodos estáticos de acceso a datos, así como a clases polimórficas.

Table Data Gateway define la estructura de acceso por registro en las entidades de la base de datos, especifica su acceso a nivel de tabla, proponiendo un objeto que se comporte como una puerta de enlace a cada tabla de la base de datos.

Doctrine utiliza el patrón *Active Record* para acceder a los datos de la base de datos. Una fila en la tabla se envuelve en una clase que implementa métodos de acceso para cada columna de la tabla, de manera que se asocian filas únicas de la base de datos con objetos del lenguaje de programación usado. Cuando se crea uno de estos objetos, se añade una fila a la tabla de la base de datos y cuando se modifican los atributos del objeto, se actualiza la fila de la base de datos.

3.4.2 Diagrama de clases del diseño

La forma tradicional de modelar las clases del diseño, no es factible a la hora de diseñar una aplicación Web. Por ese motivo, se utiliza una extensión de UML para Web, que se adapta a la arquitectura de este tipo de sistemas.

Para obtener un nivel correcto de abstracción y detalle se modelan los artefactos del sistema como las páginas, los enlaces entre las mismas, todo el código que irán creando, así como el contenido dinámico de estas una vez que estén en el navegador del cliente; estos son los artefactos que se necesitan modelar para que el desarrollador los implemente luego y obtener así el producto final.

Los diagramas de secuencia forman parte del modelado dinámico del sistema y proporcionan una vista detallada de los casos de uso. Son diagramas que muestran la interacción organizada de objetos, mediante mensajes que se envían entre sí, en una secuencia de tiempo. Son útiles para observar la vida de los objetos en un sistema, identificar llamadas a realizar o posibles errores del modelado estático que imposibiliten el flujo de información. En el Anexo 6 se puede consultar una representación de los diagramas de secuencia. A continuación se representan los diagramas de clases del diseño, por cada realización de caso de uso del sistema, para su mayor comprensión.

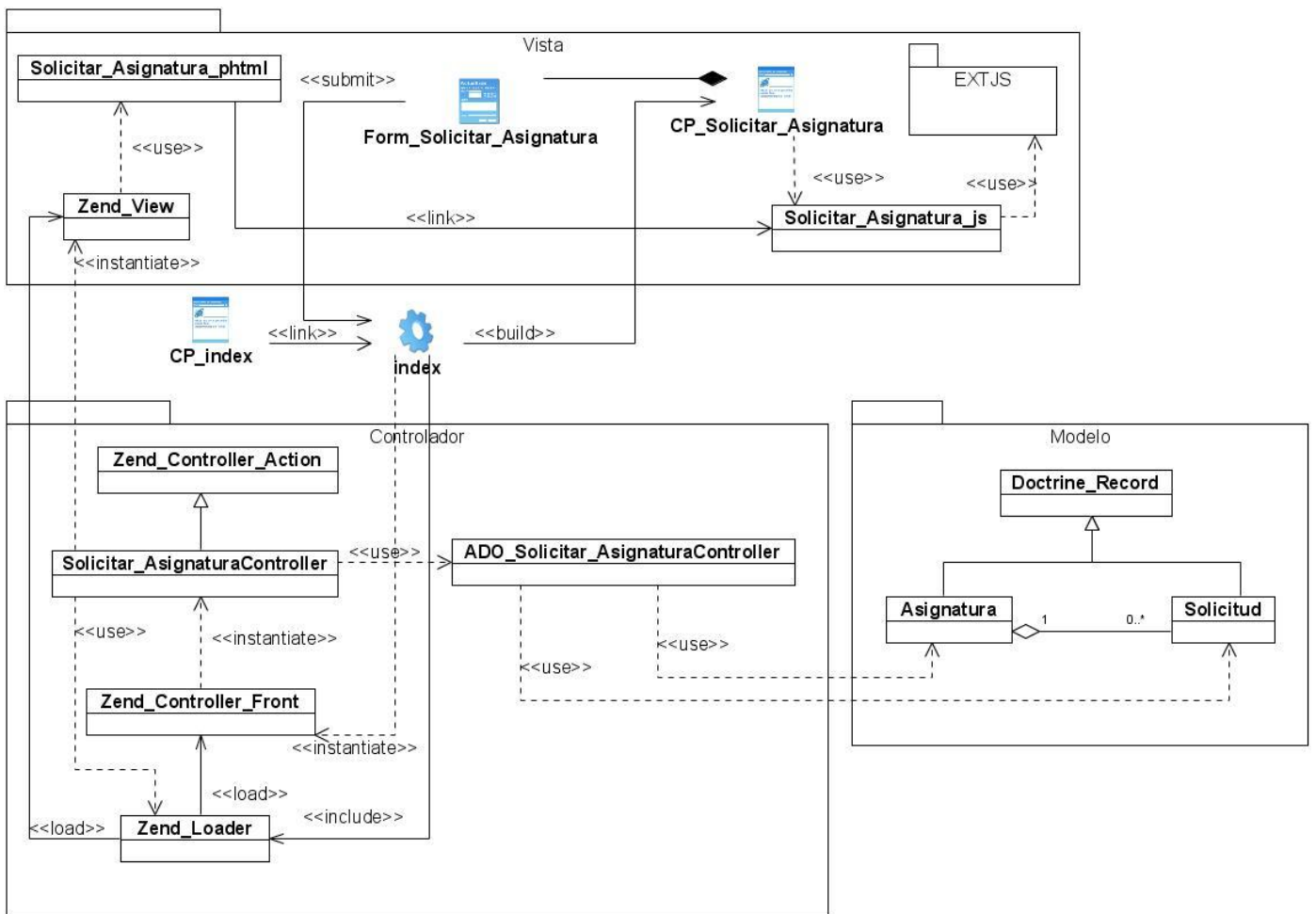


Figura 27. Diagrama de clases Web: CUS “Solicitar Asignatura Adelanto”

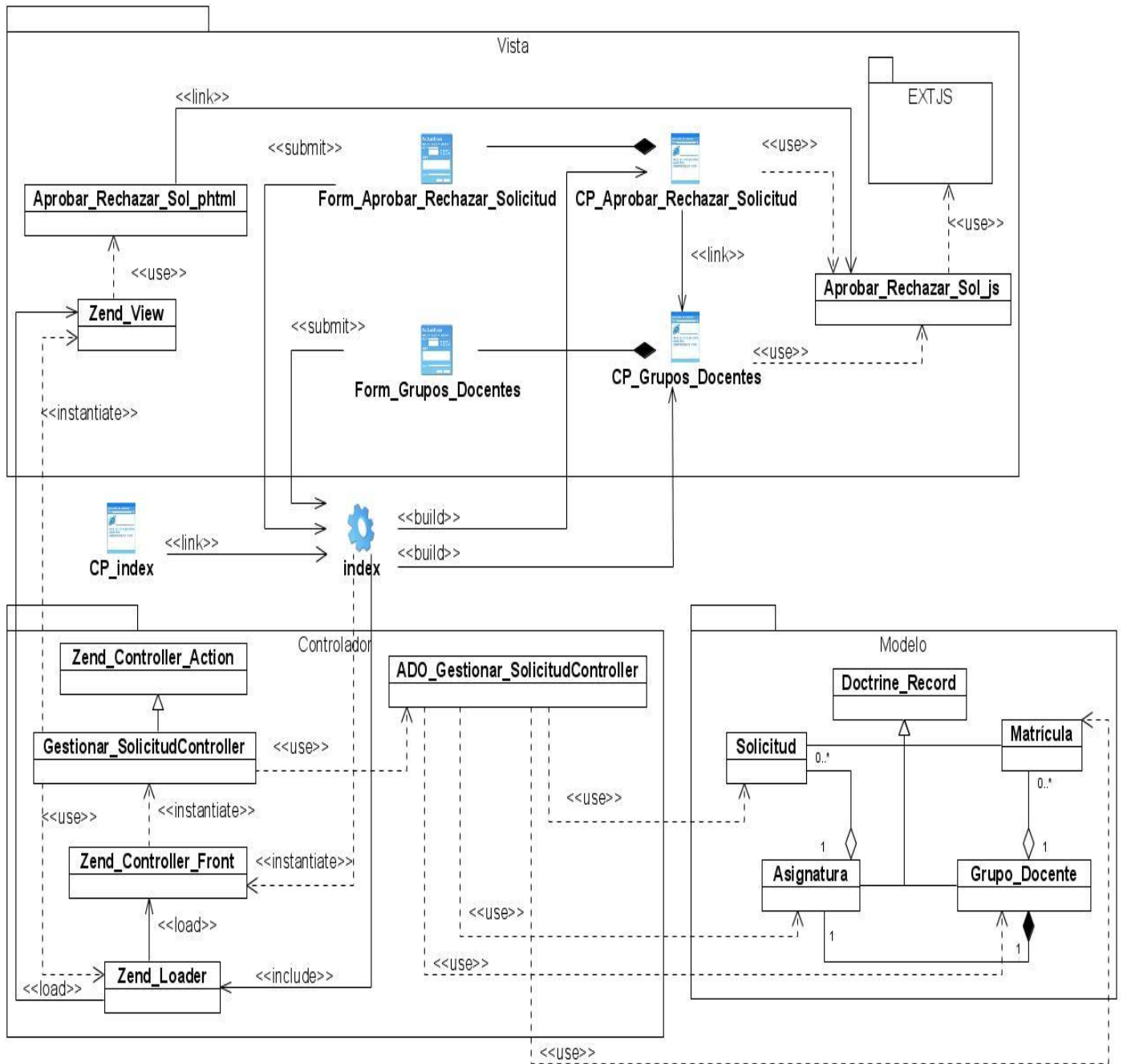


Figura 28. Diagrama de clases Web: CUS “Aprobar_Rechazar Solicitud de Asignatura”

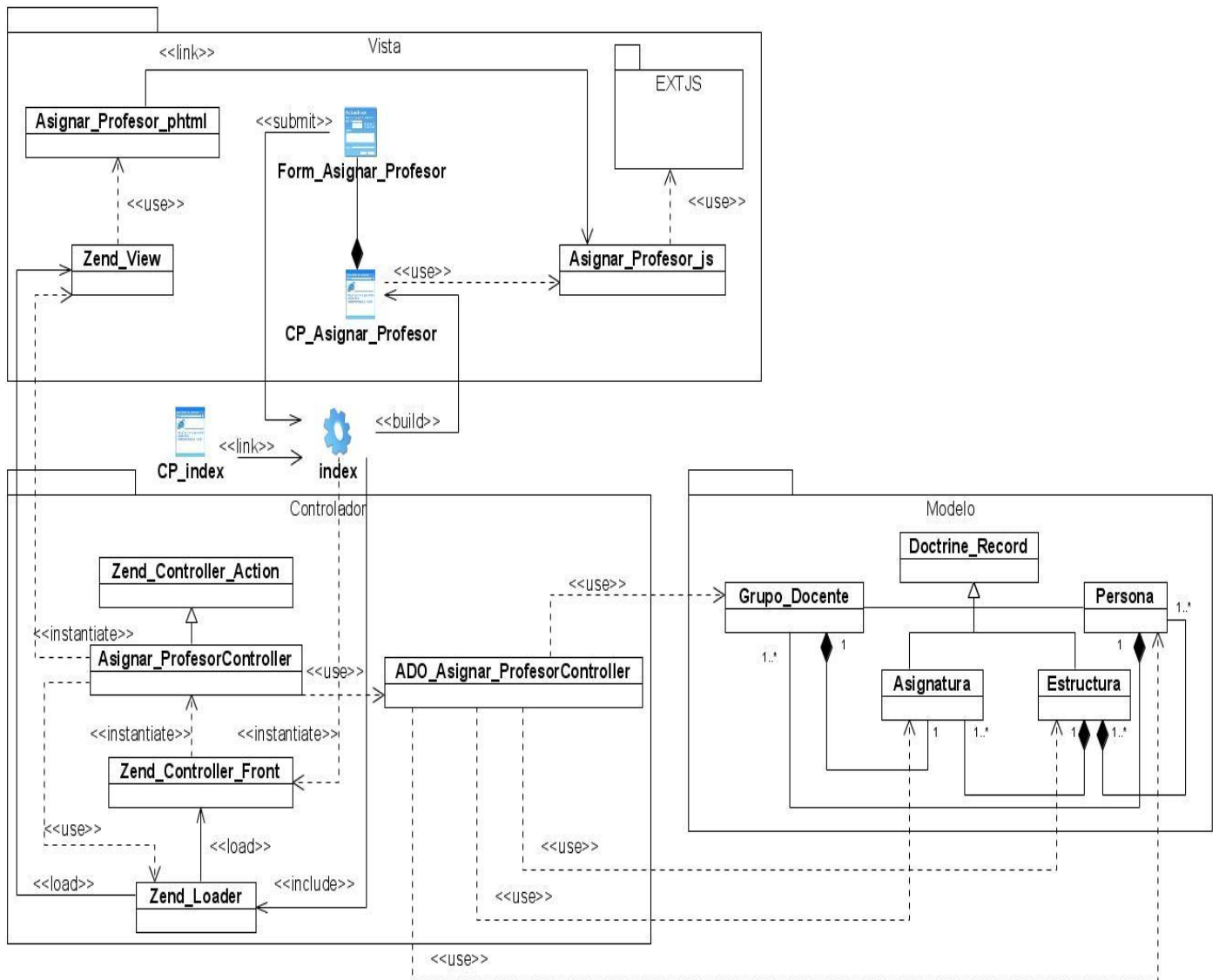


Figura 29. Diagrama de clases Web: CUS “Asignar Profesor a Grupo Docente”

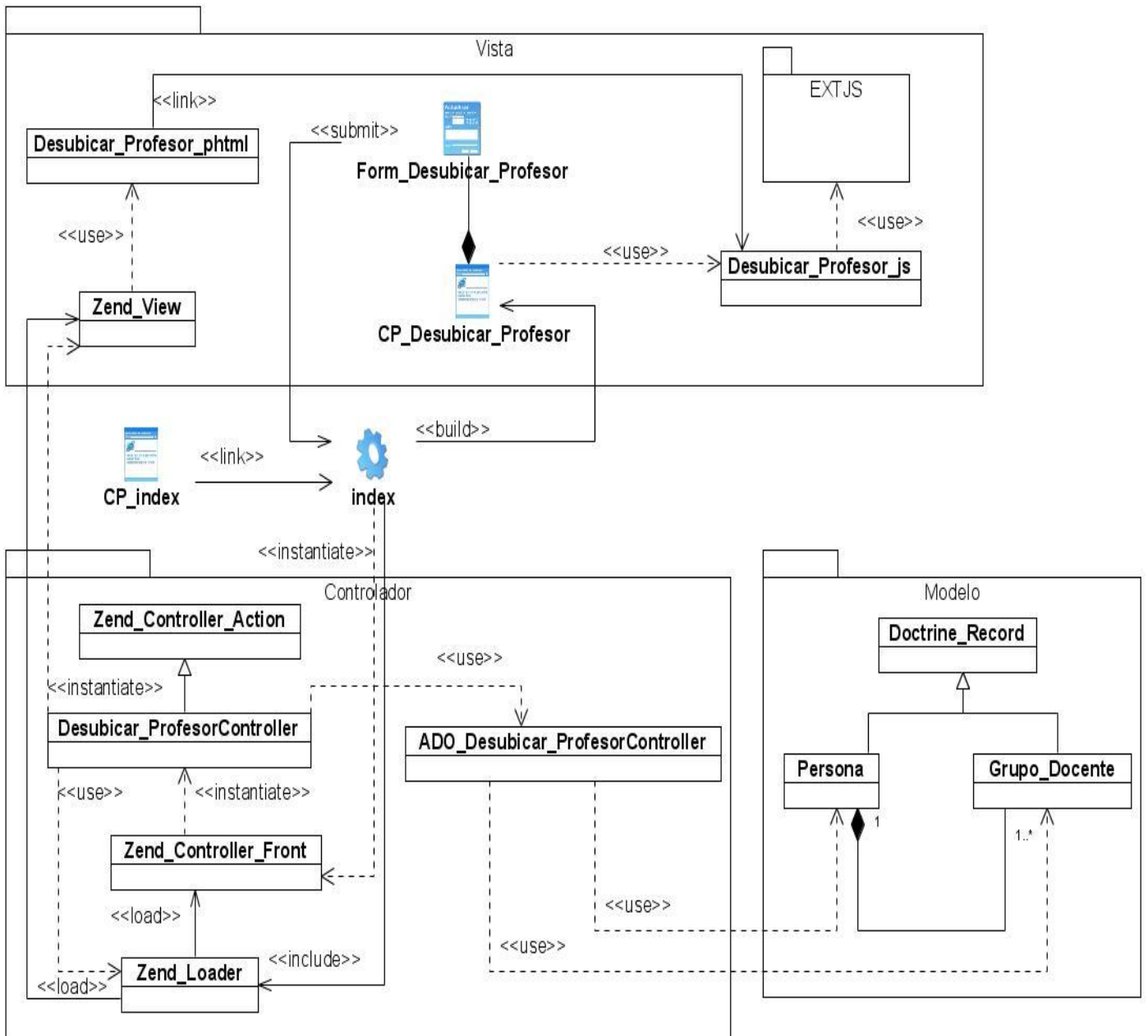


Figura 30. Diagrama de clases Web: CUS “Desubicar Profesor de Grupo Docente”

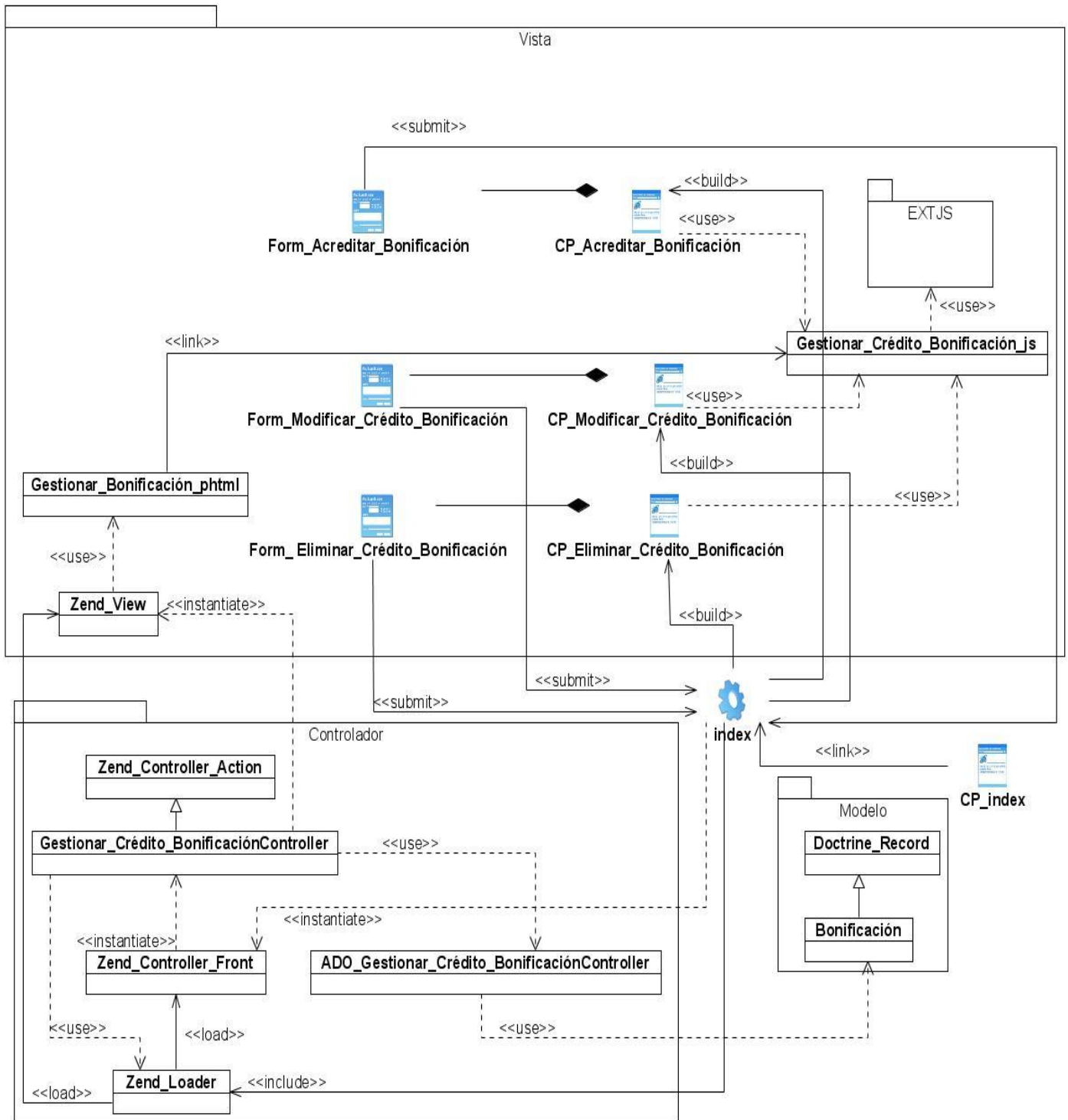


Figura 32. Diagrama de clases Web: CUS “Gestionar Crédito Bonificación”

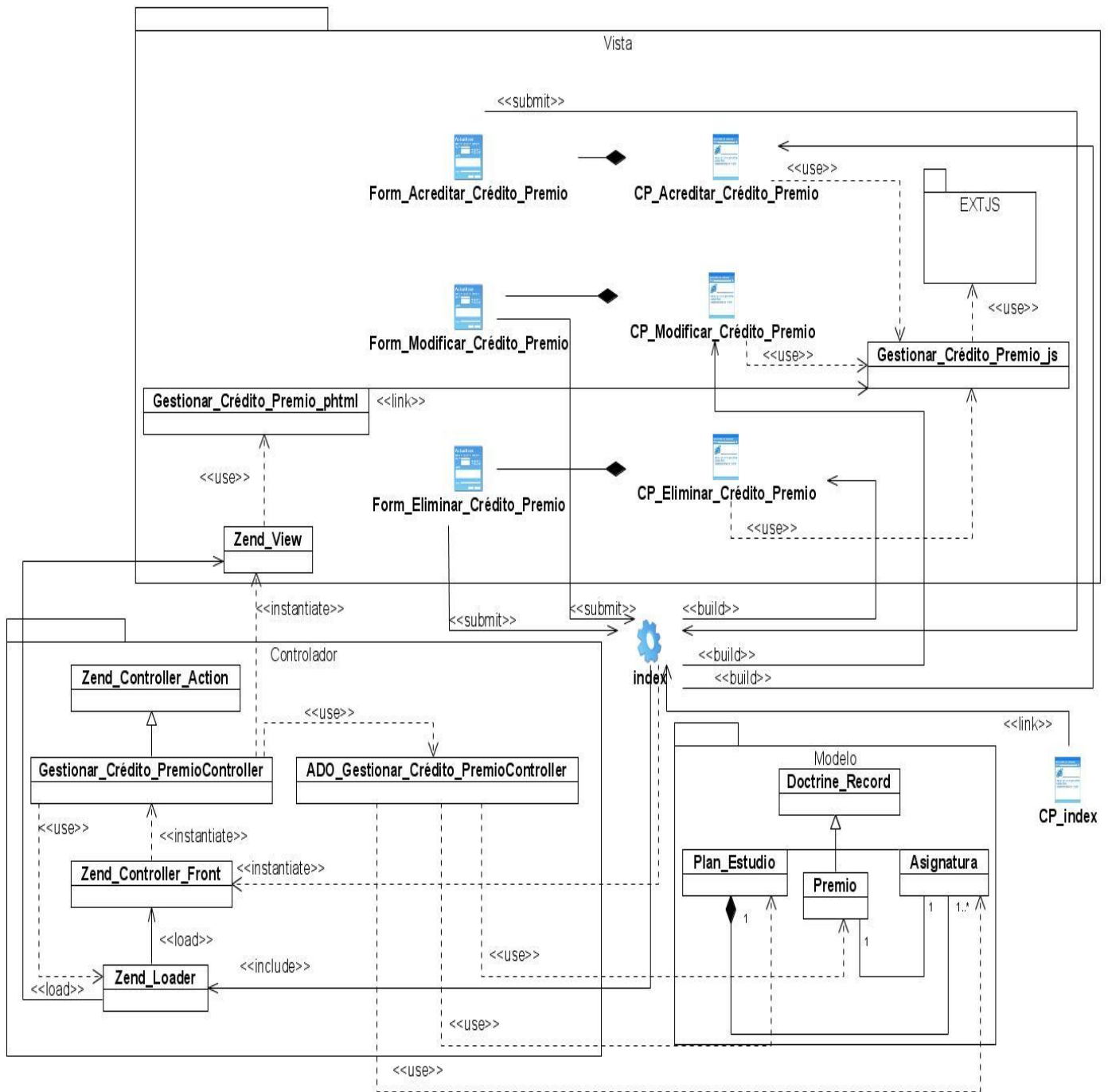


Figura 33. Diagrama de clases Web: CUS “Gestionar Crédito Premio”

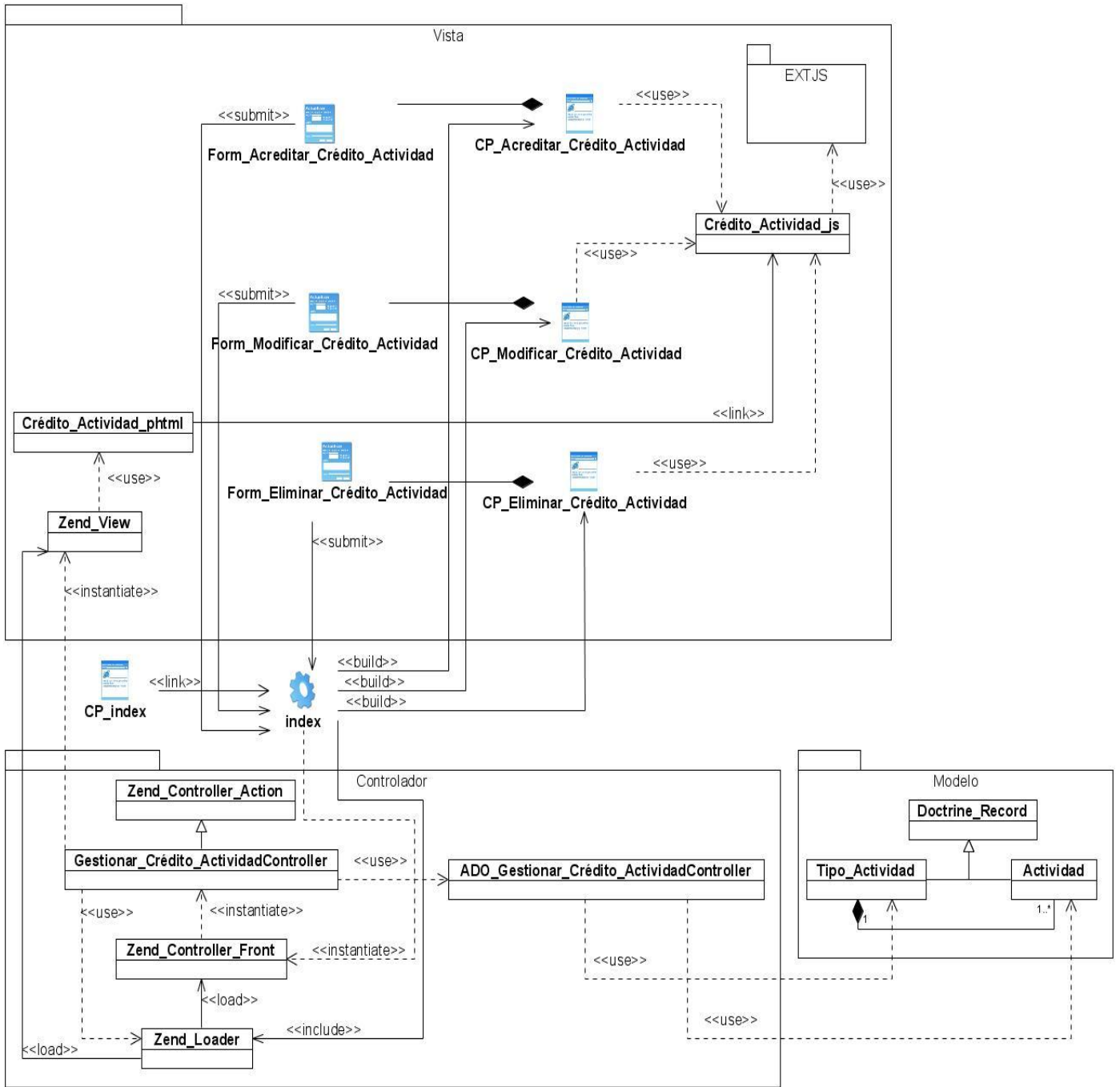


Figura 34. Diagrama de clases Web: CUS “Gestionar Crédito por Actividades”

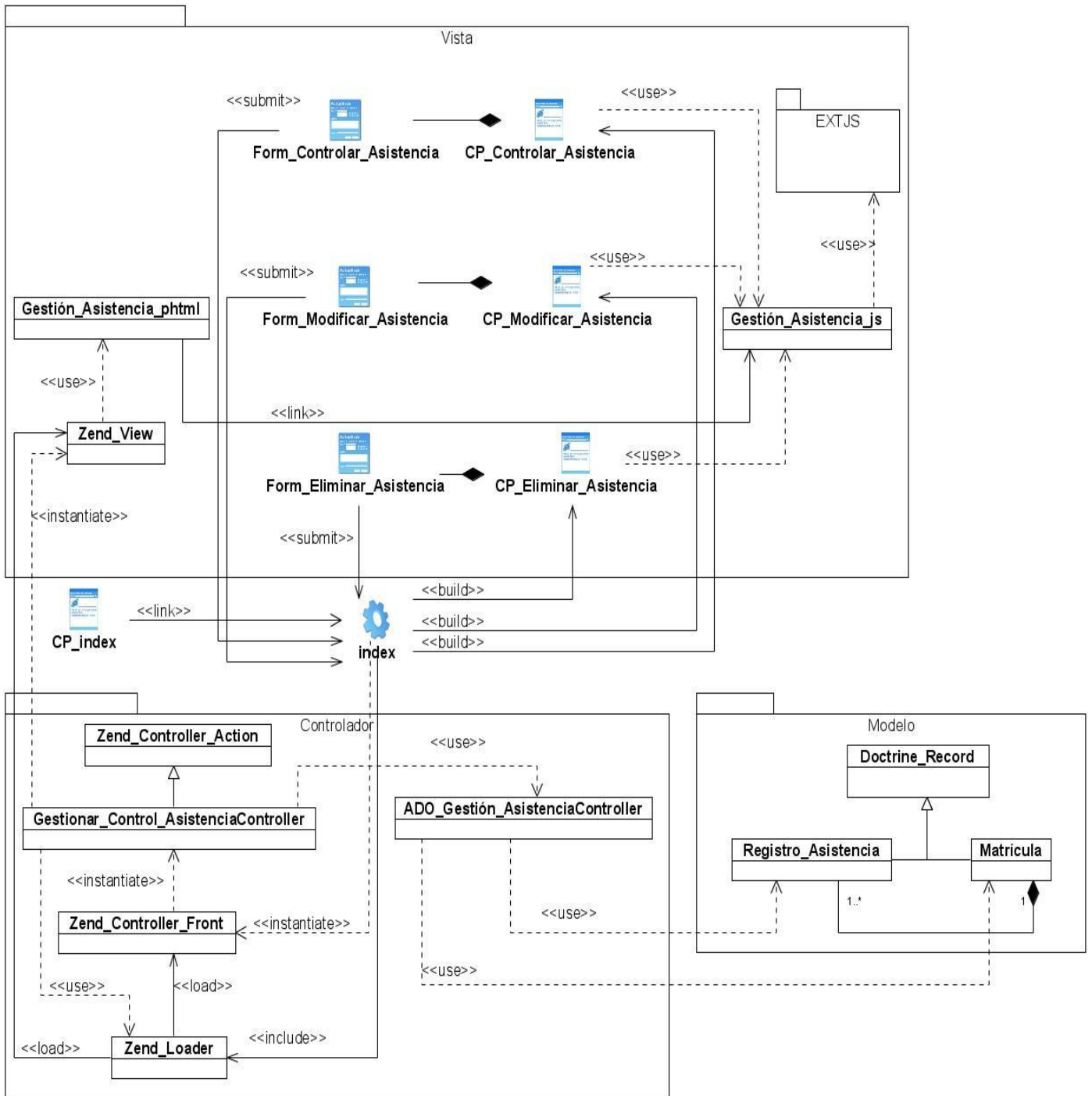


Figura 35. Diagrama de clases Web: CUS “Gestionar Control Asistencia”

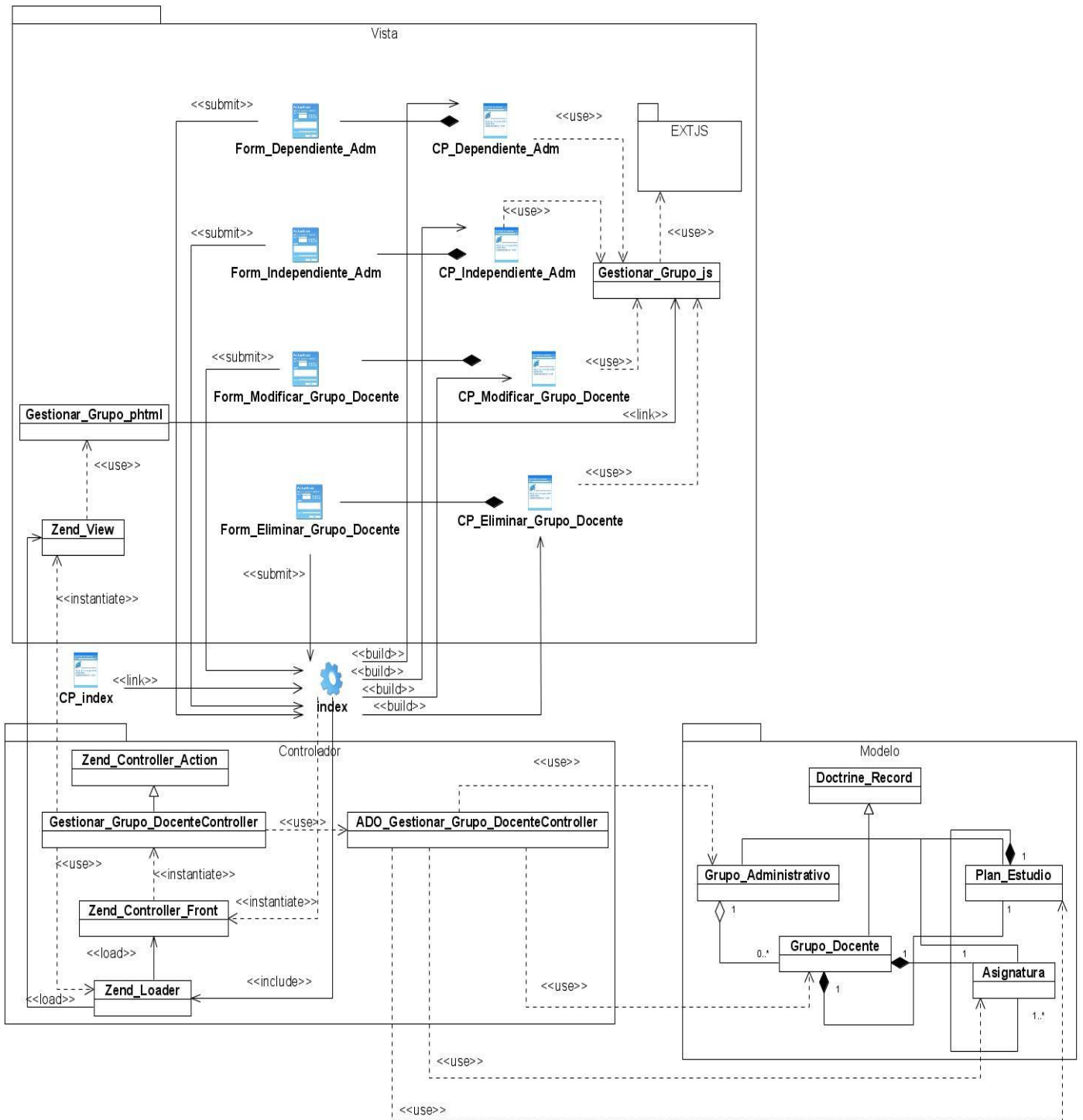


Figura 37. Diagrama de clases Web: CUS “Gestionar Grupo Docente”

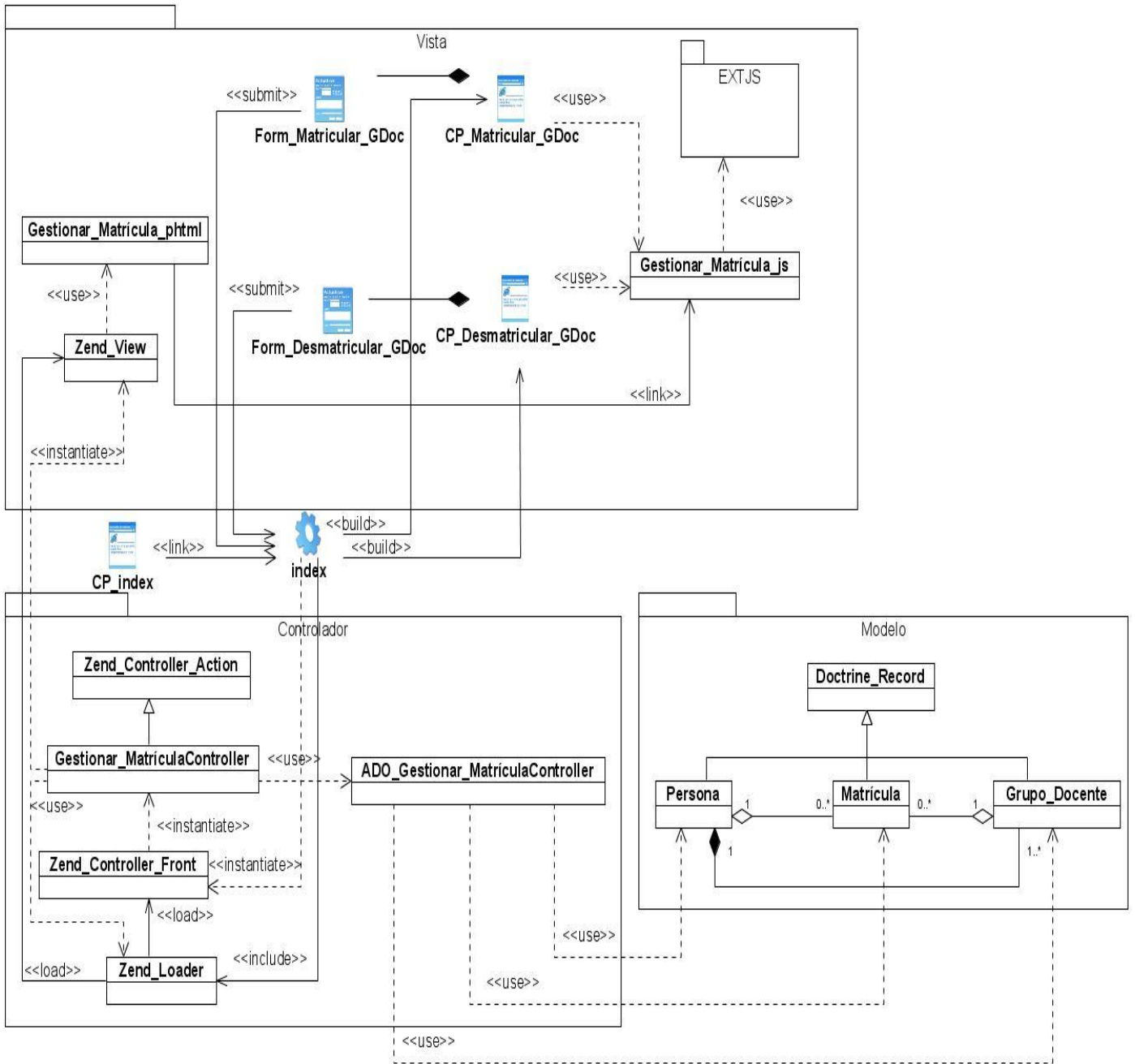


Figura 38. Diagrama de clases Web: CUS “Gestionar Matrícula en Grupo Docente”

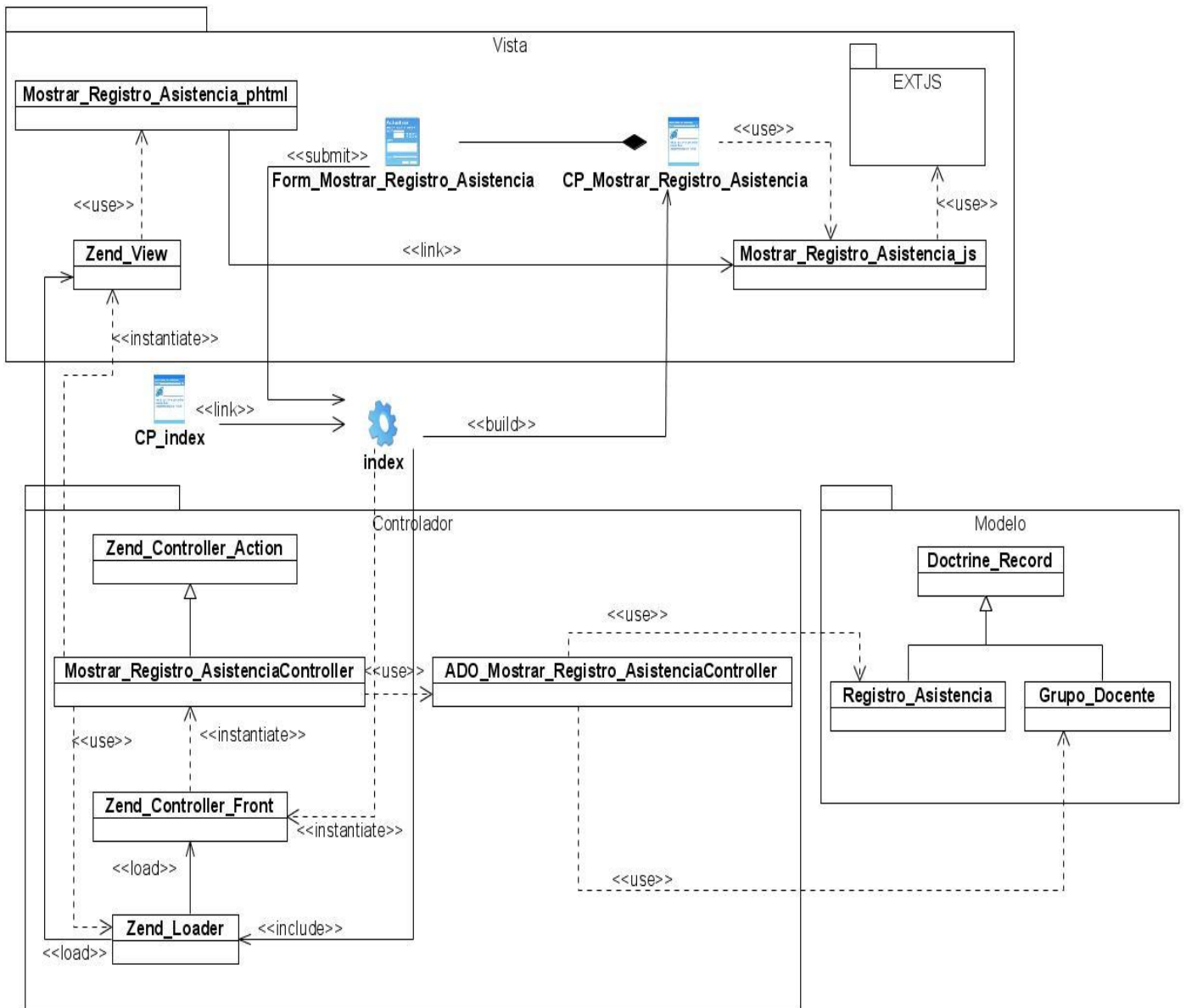


Figura 40. Diagrama de clases Web: CUS “Mostrar Registro de Asistencia”

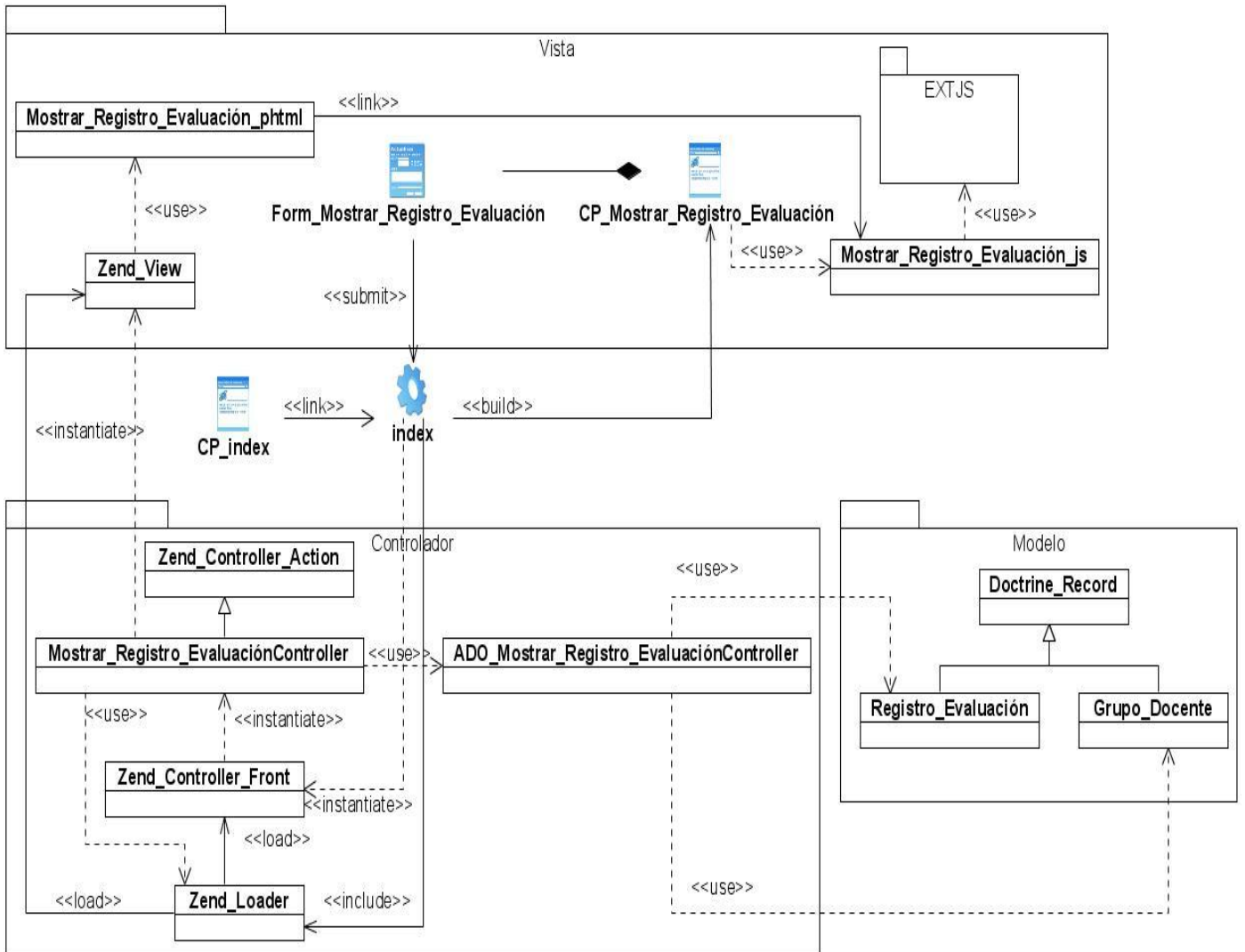


Figura 41. Diagrama de clases Web: CUS “Mostrar Registro de Evaluación”

3.5 Diseño de la base de datos

Para el diseño de la base de datos se utilizaron los modelos físicos y lógicos de datos, a través de los cuales se diseñaron las clases persistentes con sus relaciones y se modeló la distribución de las tablas en la base de datos. El diseño propuesto satisface las necesidades de persistencia de los datos que el módulo requiere en cumplimiento de sus requerimientos funcionales y permite la integración con el resto del sistema.

3.5.1 Modelo lógico

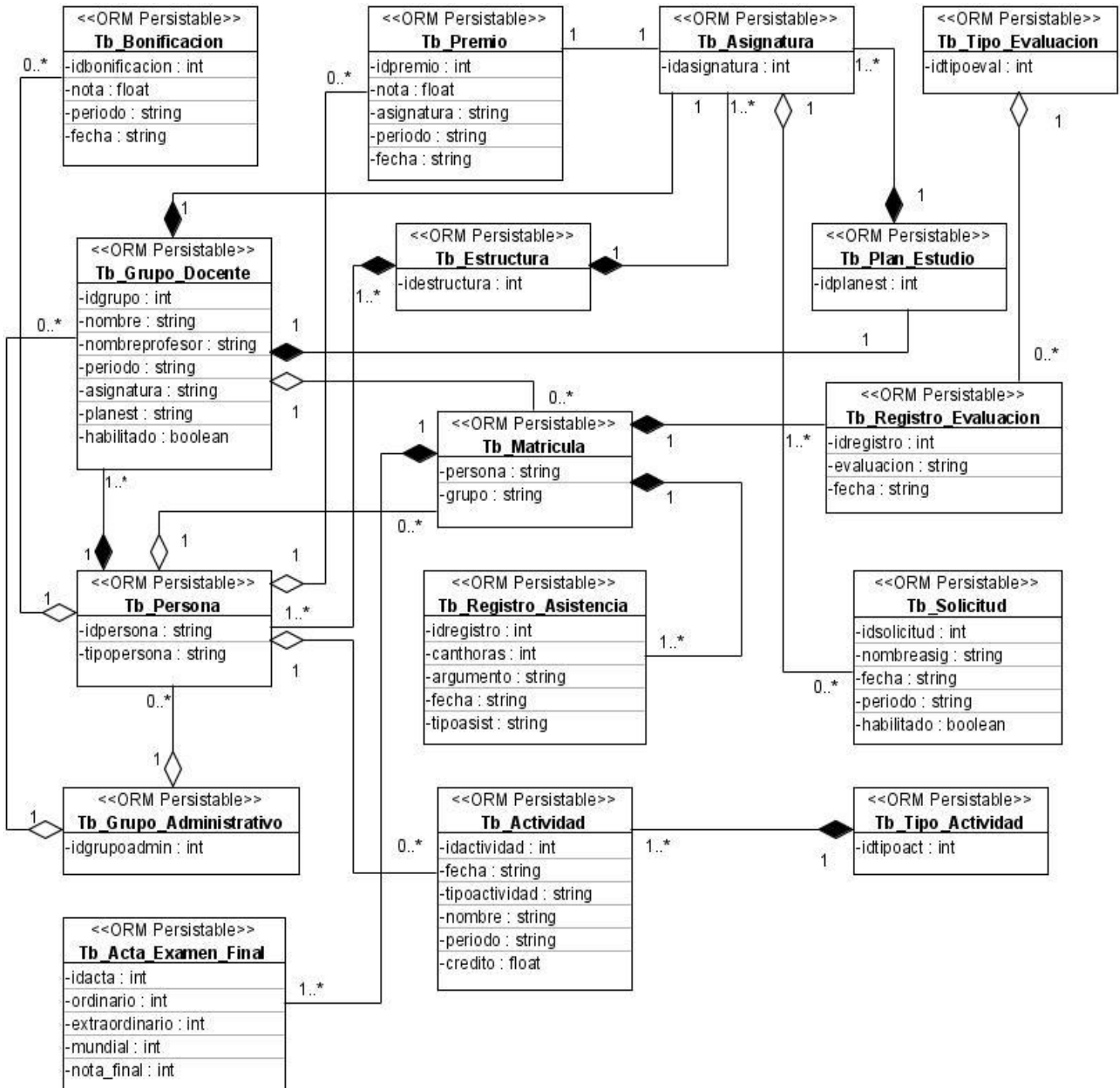


Figura 42. Modelo lógico

3.5.2 Modelo físico

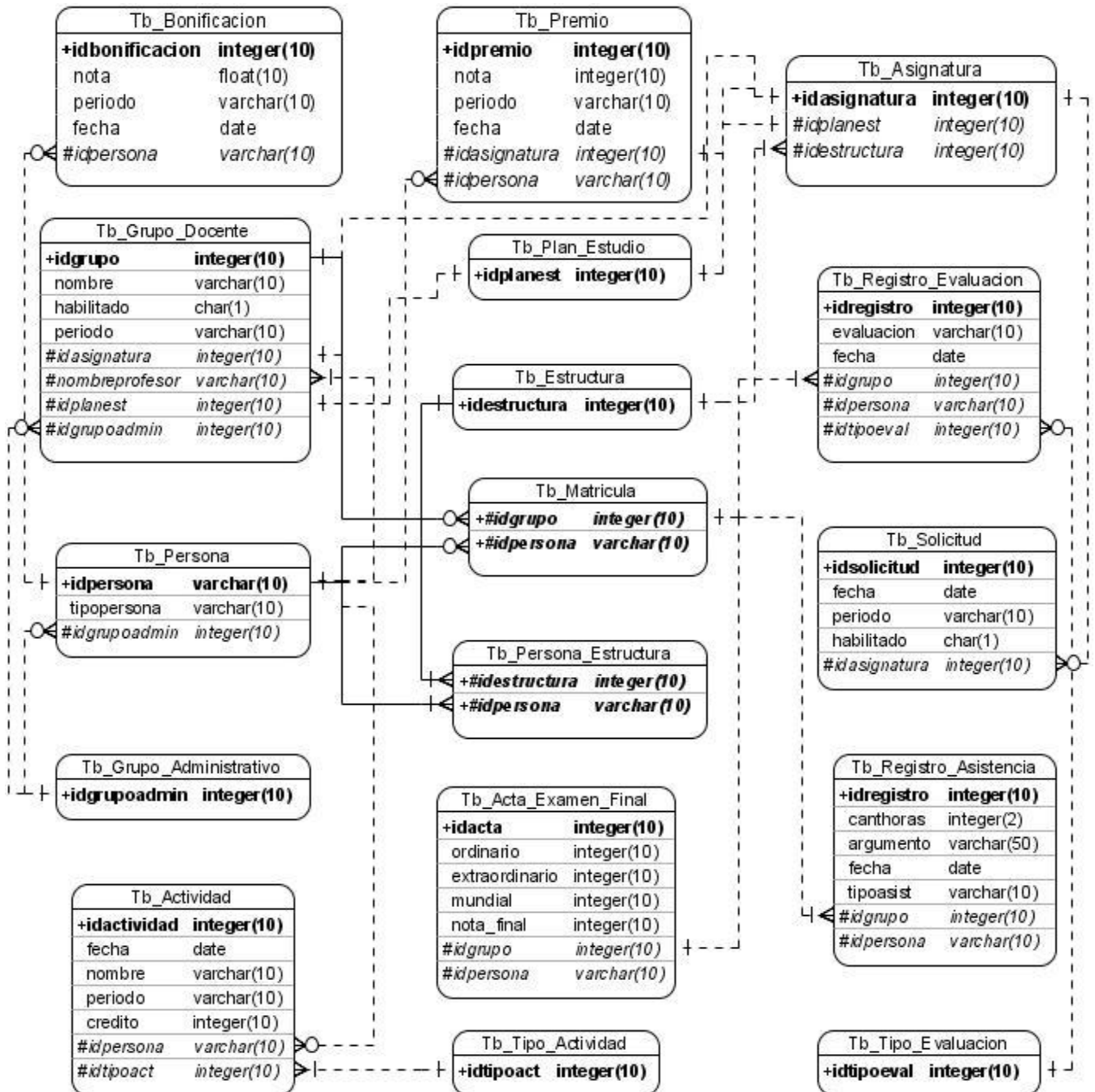


Figura 43. Modelo físico

3.6 Diagrama de despliegue

El modelo de despliegue representa los nodos que conforman la topología de hardware sobre la que se ejecuta el sistema y sus relaciones. Típicamente, los nodos son conectados por asociaciones de comunicación tales como enlaces de red, conexiones TCP, IP, microondas, etc. A continuación se muestra la distribución física del sistema teniendo en cuenta la arquitectura del software y del hardware.

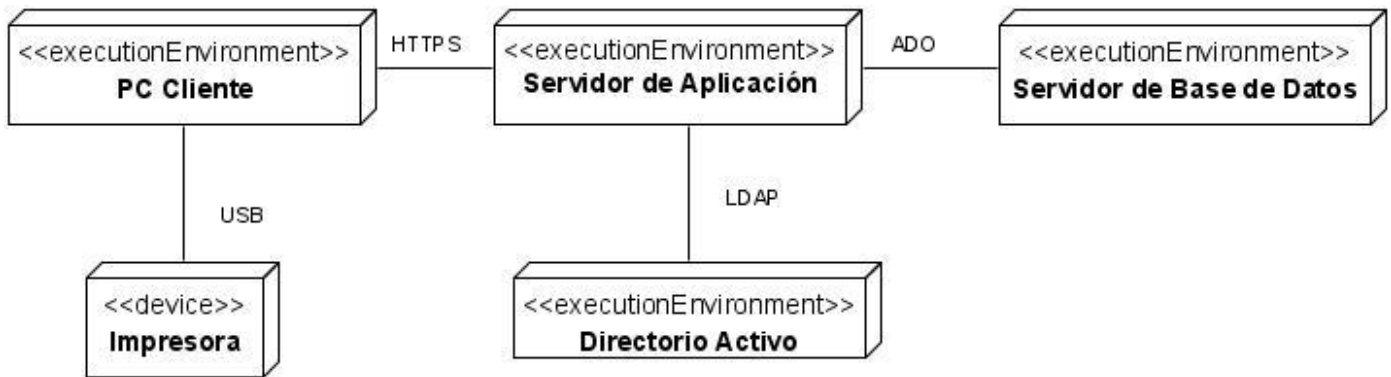


Figura 44. Diagrama de despliegue

3.7 Concepción general de la ayuda

El acceso al sistema será realizado por trabajadores que tienen poco o escaso conocimiento informático, que se verán a menudo trabajando en un ambiente bastante dinámico, por lo cual deben estar asistidos en cada momento con la información necesaria.

La existencia de un manual de usuario donde se explique el funcionamiento, los requerimientos y una lista de preguntas frecuentes con sus respuestas, será un elemento de mucha utilidad, acompañado por una ayuda local de la página donde se encuentre el usuario o accesible en el propio menú de los servicios del sistema.

3.8 Tratamiento de errores

Como fundamento de la aplicación para la prevención de errores provocados por entradas incorrectas por parte de los usuarios se propone que estos sólo realicen las entradas mínimas necesarias dentro del dominio del sistema, una de las técnicas empleadas es conducir a los usuarios solamente por los caminos que concuerden con las funcionalidades del sistema permitidas, inhabilitando el resto de los controles de la interfaz que no necesiten ser utilizados en cada una de las operaciones.

Se debe realizar en todo momento un correcto uso de las buenas prácticas de validación de entradas tanto del lado del cliente y como del servidor. Lo que garantizará la integridad del flujo y almacenamiento de la información enviada por el usuario.

Se propone además, que exista un mecanismo de réplicas de mensajes que permitan conocer los detalles de las excepciones lanzadas por el sistema en cada momento, realizando el tratamiento de las mismas e impidiendo que errores conocidos dejen inhabilitadas temporal o radicalmente las funcionalidades del sistema.

3.9 Conclusiones

Teniendo en cuenta la estimación realizada y los beneficios tangibles que se podrán obtener, entre los que se destacan el uso más eficiente del tiempo de trabajo de las secretarías y la accesibilidad a la información almacenada de los estudiantes y considerando un grupo de factores que ofrece la universidad cuya utilización no genera costo alguno, se puede concluir que la realización del Módulo Registro y Control Docente resulta factible.

Se finaliza la etapa de análisis y diseño del sistema obteniendo un modelo más detallado de la solución propuesta. Se realizaron los diagramas de clases del análisis y del diseño de los casos de uso críticos del sistema, así como los diagramas de colaboración y secuencia para cada una de las funcionalidades descritas, lo que permite una idea más específica del sistema que se propone. Se describe la arquitectura y los patrones que se tuvieron en cuenta para el diseño de los casos de uso. Se muestran además los modelos físico y lógico de datos que satisfacen las necesidades del módulo y que son de gran importancia para la implementación del mismo, así como el diagrama de despliegue.

Conclusiones

Para el desarrollo de esta investigación se realizó un estudio bibliográfico que permitió tener un conocimiento profundo de la situación actual y las tendencias de los sistemas de gestión académica. A través del mismo, se demostró la necesidad de desarrollar un módulo capaz de automatizar los procesos de control docente para la nueva versión del Sistema Automatizado para la Gestión Académica – “Akademos”. La automatización de estos procesos debe permitir que el sistema pueda ser aplicado en diferentes centros educacionales.

La propuesta que se hace en este trabajo del Módulo Registro y Control Docente para la nueva versión del proyecto Akademos V2.0, permitirá gestionar la información referente a los estudiantes de un centro de estudios y acceder a la misma a través de un grupo de reportes. La solución que se presenta brinda la posibilidad de eliminar la información duplicada y permitirá el almacenamiento de todos los datos necesarios para el seguimiento de los estudiantes. Las funcionalidades que se tuvieron en cuenta para el desarrollo de esta propuesta permitieron obtener el diseño de un módulo encaminado a eliminar las dificultades presentadas por el actual módulo del sistema.

El desarrollo del trabajo con la metodología RUP permitió documentar el mismo desde sus inicios, lo que servirá como base de estudio para futuros miembros del equipo de trabajo permitiendo de esta forma una comprensión más rápida y fácil de todas las etapas de concepción del módulo. La propuesta de análisis y diseño obtenida, constituye una base para el trabajo de los implementadores del módulo, así como para futuras iteraciones o ciclos de desarrollo.

Recomendaciones

A partir del estudio realizado en este trabajo y teniendo en cuenta un grupo de ideas que surgieron durante el desarrollo del análisis y diseño del módulo, se considera que se puede recomendar a la dirección del proyecto Akademos las siguientes acciones a tener en cuenta de forma tal que se logre una mayor eficiencia:

- Continuar la investigación comenzada en este trabajo manteniendo un seguimiento sobre los sistemas de gestión académica que son utilizados en el mundo, para garantizar mejoras en las próximas fases de desarrollo del módulo.
- Realizar el análisis y diseño de los casos de uso del sistema que no se tuvieron en cuenta para el desarrollo de esta iteración, considerando la importancia que tienen para la utilización del módulo en los centros educacionales.
- Implementar los casos de uso propuestos para esta iteración utilizando el análisis y diseño obtenido en este trabajo.

Bibliografía referenciada

1. AUTORES, C. D. Grupo GES Sistemas Avanzados .2007, Disponible en:
http://www.grupoges.com.mx/grupoges_escolarges.php
2. AUTORES, C. D. ¿Qué aporta a mi Centro? ¿Qué es Ágora? Disponible en:
http://www.kherian.com/agora_que.asp
3. MARRERO, D. P. C. J. L. A. O. Y. S. R. H. E. Manual de Usuario del SIGENU. Sistema de Gestión de la Nueva Universidad. Abril del 2007.
4. AUTORES, C. D. La definición de software libre. Disponible en: <http://www.gnu.org/philosophy/free-sw.es.html>
5. GARBEE, B. Conferencia Internacional de software libre. 21 de Enero 2006, Disponible en:
<http://www.opensourceworldconference.com/malaga06/es/modules/news/article.php?storyid=354>
6. FRANK BENAVIDEZ DALMENDRAY, D. C. R. Estudio de alternativas para la migración del Sistema Automatizado para la Gestión Académica “Akademos” a software libre. Mayo 2007, Disponible en:
<http://biblioteca.uci.cu/sbd/biuci/index.html>
7. RAMOS, E. I. Centro de Validación y Certificación de Competencias Profesionales UCI. Mayo 2007, Disponible en: <http://biblioteca.uci.cu/sbd/biuci/index.html>
8. AUTORES, C. D. Conferencia #2 Ingeniería de Software. Fase de Inicio. Modelo del Negocio. 2007_2008, Disponible en: <http://teleformacion.uci.cu/mod/resource/view.php?id=11553>
9. AUTORES, C. D. Conferencia #4 Ingeniería de Software. Flujo de Requerimiento. 2008_2009, Disponible en: <http://teleformacion.uci.cu/mod/resource/view.php?id=12103>
10. AUTORES, C. D. Clase Teórica-Práctica 6. Técnicas de Estimación. Ingeniería de Software 1. 2008_2009, Disponible en: <http://teleformacion.uci.cu/mod/resource/view.php?id=12606>
11. IVAR JACOBSON, G. B., JAMES RUMBAUGH. El Proceso Unificado de Desarrollo del software. Editorial Félix Varela, 2004. Vol. 1.

Bibliografía consultada

ARTAZA, D. D. L. D. I. G. D. AJAX: Asynchronous JavaScript and XML (AJAX). 24 Octubre 2005.

AUTORES, C. D. Conceptos básicos. ¿Qué es HTML? Disponible en:
http://gias720.dis.ulpgc.es/Gias/Cursos/Tutorial_html/concepto.htm#html

AUTORES, C. D. El modelo cliente/servidor. Disponible en:
<http://agamenon.uniandes.edu.co/~revista/articulos/cliser.html>

AUTORES, C. D. Información detallada módulos Siga. Disponible en: <http://www.dara.es/siga/sigainf.htm>

AUTORES, C. D. January 2009 Web Server Survey. 2009, Disponible en:
http://news.netcraft.com/archives/2009/01/16/january_2009_web_server_survey.html

AUTORES, C. D. Lenguaje JSP. 27 junio 2007, Disponible en:
<http://petra.euitio.uniovi.es/~i6950404/wiki/pmwiki.php?n=Tema8.LenguajeJSP>

AUTORES, C. D. Principales definiciones de los términos más usados en Internet. 2008, Disponible en:
<http://www.informaticamilenium.com.mx/Paginas/espanol/sitioWeb.htm#dinternet>

AUTORES, C. D. Script del lado del servidor. 1 febrero 2009, Disponible en:
http://es.wikipedia.org/wiki/Script_del_lado_del_servidor

AUTORES, C. D. Servicio Web. 22 abril 2009, Disponible en: http://es.wikipedia.org/wiki/Servicio_Web

AUTORES, C. D. Servidor HTTP Apache. 29 abril 2009, Disponible en:
http://es.wikipedia.org/wiki/Servidor_HTTP_Apache

AUTORES, C. D. Una introducción a APACHE. 2006, Disponible en:
http://linux.ciberaula.com/articulo/linux_apache_intro/

ELISABEL PÉREZ URBAY, N. M. H. Análisis y Diseño del módulo Estudiantes para Akademos. Junio 2007, Disponible en: <http://biblioteca.uci.cu/sbd/biuci/index.html>

HERNÁNDEZ, A. S. Arquitectura para Akademos v2.0. Junio 2008, Disponible en:

<http://biblioteca.uci.cu/sbd/biuci/index.html>

INFORMÁTICA, C. D. A. F. D. Patrones del “*Gang of Four*”. Unidad Docente de Ingeniería del Software.

LARMAR.C. UML y Patrones. Introducción al análisis y diseño orientado a objetos. Editorial Félix Varela, 2004. Vol. 1.

LIC. YANOSKI CALDERÍN DELGADO, D. J. A. T. G. GESTACAD. Sistema para la gestión académica.

LUIS GIRALDO, Y. Z. Herramientas de desarrollo de ingeniería de SW para Linux. 24 de Septiembre de 2005.

PEDRO HERNÁNDEZ VALVERDE, N. J. J., ANA MARIA MORENO SÁNCHEZ CAPUCHINO. El Proceso Unificado de Rational (RUP) y su relación con las técnicas y métodos de la ingeniería de usabilidad de software. 2004_2005, Disponible en: <http://is.ls.fi.upm.es/doctorado/Trabajos20042005/Hernandez.pdf>

PRESSMAN.R. Ingeniería de Software. Un enfoque práctico. Editorial Félix Varela, 2005. Parte 1 y Parte 2.

PROENZA, Y. E. A. Diseño Avanzado de Aplicaciones Web. EXT – Zend Framework y Doctrine, 2009, Disponible en: Revista de Software Libre ATIX # 2, <http://atix.opentelematics.org>

RODÓN, D. C. y FIGUEREDO, R. A. Análisis y Diseño del Sistema de Venta y Admisión de la plataforma de Servicios Postales Universidad de las Ciencias Informáticas, 2007.

SAMPIER, R. H. Metodología de la Investigación. 2008.

TORRE, A. D. L. Lenguajes del lado servidor o cliente. 2006, Disponible en:

http://www.adelat.org/media/docum/nuke_publico/lenguajes_del_lado_servidor_o_cliente.html

VELÁZQUEZ, C. M. Análisis y Diseño del Subsistema Matrícula. Sistema Automatizado para la Gestión Académica – “Akademos 2.0”. Junio 2008, Disponible en: <http://biblioteca.uci.cu/sbd/biuci/index.html>.

Glosario de términos

API: Interfaz de Programación de Aplicaciones de su significado en inglés *Application Programming Interface*, es un conjunto de funciones y procedimientos que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción.

CGI: Interfaz de Entrada Común de su significado en inglés *Common Gateway Interface*. Tecnología que permite a un cliente solicitar datos de un servidor Web.

Cookies: fragmento de información que se almacena en el disco duro de un visitante a una página Web a través de su navegador a petición del servidor de la página.

Desmatricular: término utilizado por las secretarías docentes para eliminar a un estudiante de un grupo docente.

Desubicar: término utilizado por las secretarías docentes para eliminar un profesor de un grupo docente.

DHTML: HTML Dinámico de su significado en inglés *Dynamic HTML*, es un conjunto de técnicas que permiten crear sitios Web interactivos utilizando una combinación de lenguaje HTML estático, un lenguaje interpretado en el lado del cliente (como JavaScript), el lenguaje de hojas de estilo en cascada (CSS) y la jerarquía de objetos DOM.

DOM: Modelo en Objetos para la Representación de Documentos de su significado en inglés *Document Object Model*, es una interfaz de programación de aplicaciones que proporciona un conjunto estándar de objetos para representar documentos HTML y XML, un modelo estándar sobre cómo pueden combinarse dichos objetos y una interfaz estándar para acceder a ellos y manipularlos.

Estructuras Administrativas: estructuras administrativas de la institución, Ejemplo: Facultad 1 y Facultad 2, que son estructuras de tipo facultad y 1501 y 1403, que son de tipo grupo.

Grupo Docente: relación estudiante-asignatura que representa a los estudiantes que están matriculados en una asignatura dada y la reciben en un grupo administrativo específico.

HTTP: Protocolo de Transferencia de Hipertexto de su significado en inglés *HyperText Transfer Protocol*. Es el protocolo usado para las transferencias de la Web.

HTTPS: versión segura del protocolo HTTP. El sistema HTTPS utiliza un cifrado basado en *Secure Socket Layers* (SSL) para crear un canal cifrado de transferencia entre el servidor remoto y el cliente.

Layouts: es la ordenación y colocación de todos los elementos que componen una página Web, es decir, textos, imágenes, tablas, gráficos, colores, tipo de letra, etc.

RSS: siglas de *Really Simple Syndication* es parte de la familia de los formatos XML desarrollado específicamente para todo tipo de sitios que se actualicen con frecuencia y por medio del cual se puede compartir la información y usarla en otros sitios Web o programas.

Scripts: conjunto de instrucciones que permiten la automatización de tareas creando pequeñas utilidades.

SOAP: siglas de *Simple Object Access Protocol*, es un protocolo estándar que está actualmente bajo el auspicio de la W3C que define cómo dos objetos en diferentes procesos pueden comunicarse por medio de intercambio de datos XML. SOAP es uno de los protocolos utilizados en los servicios Web.

UDDI: siglas del catálogo de negocios de Internet denominado *Universal Description and Discovery and Integration*. El registro en el catálogo se hace en XML. UDDI es una iniciativa industrial abierta (sufragada por la OASIS) entroncada en el contexto de los servicios Web.

Web: es el término más usado para referirse al *World Wide Web* que es la red mundial de páginas o documentos de texto entrelazados. Un documento entrelazado no es más que un documento que contiene enlaces a otros documentos o páginas de texto.

Widgets: pequeña aplicación o módulo que realiza una función concreta, generalmente de tipo visual, dentro de otras aplicaciones o sistemas operativos.

Wizards: programa que mediante pantallas interactivas facilita y sirve de guía para el uso de otros programas o aplicaciones.

WSDL: siglas de *Web Services Description Language* formato XML que se utiliza para describir servicios Web.

W3C: siglas de *World Wide Web Consortium*. Es un comité que promulga estándares con sedes en Estados Unidos, Europa y Japón. W3C se dedica, en parte, a estimular el desarrollo de estándares abiertos para la Web, como por ejemplo los lenguajes HTML y XML, promocionar la interoperabilidad y en conseguir que la Web logre todo su potencial.

XML: sigla de Lenguaje de Marcas Extensible de su significado en inglés de *eXtensible Markup Language*, es un metalenguaje extensible de etiquetas desarrollado por el W3C.

Anexos

Anexo 1: Diagrama de objetos del modelo de negocio

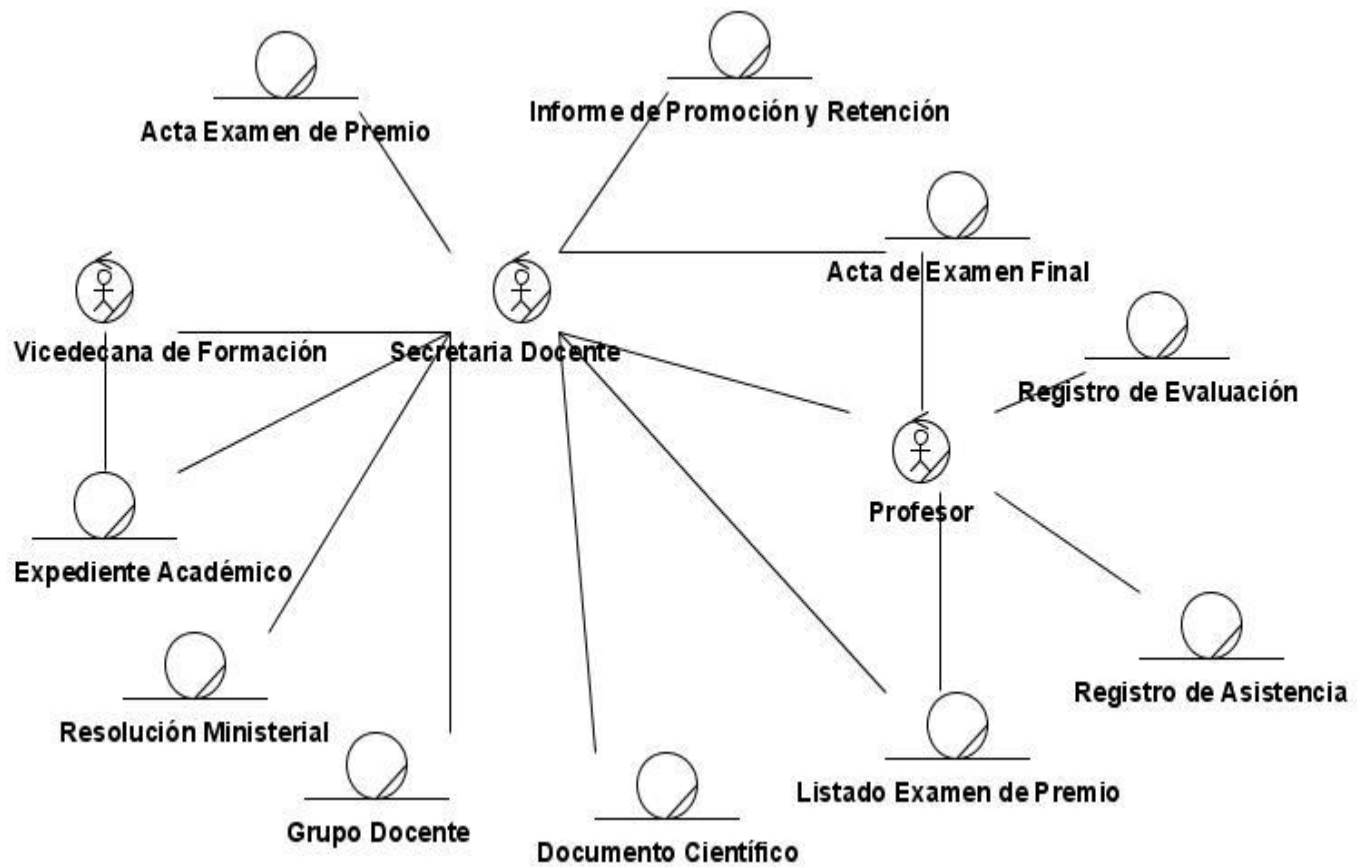


Figura 45. Diagrama de objetos del modelo de negocio

Anexo 2: Diagramas de actividades de los casos de uso del negocio

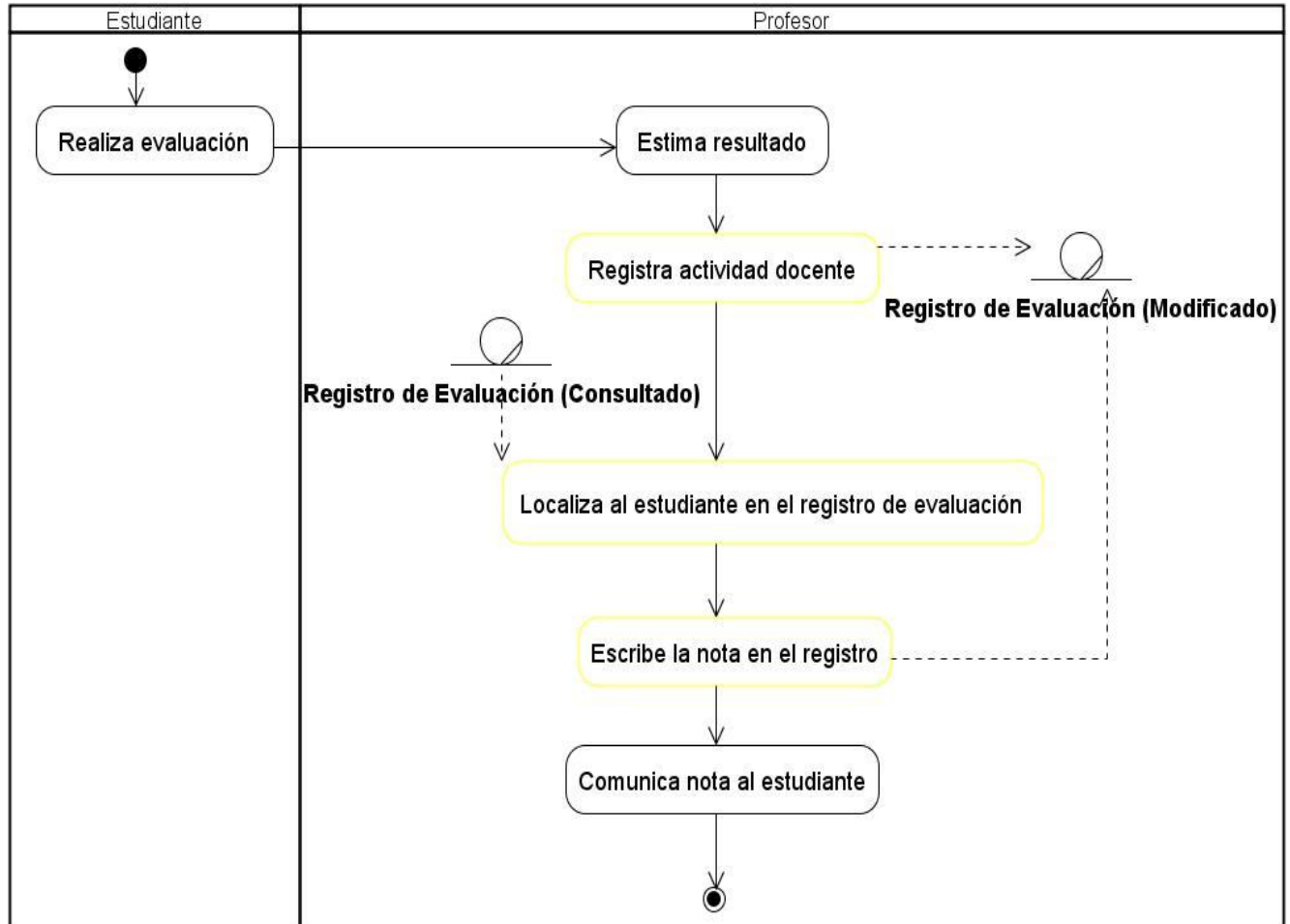


Figura 46. Diagrama de actividades del caso de uso del negocio “Controlar Evaluación”

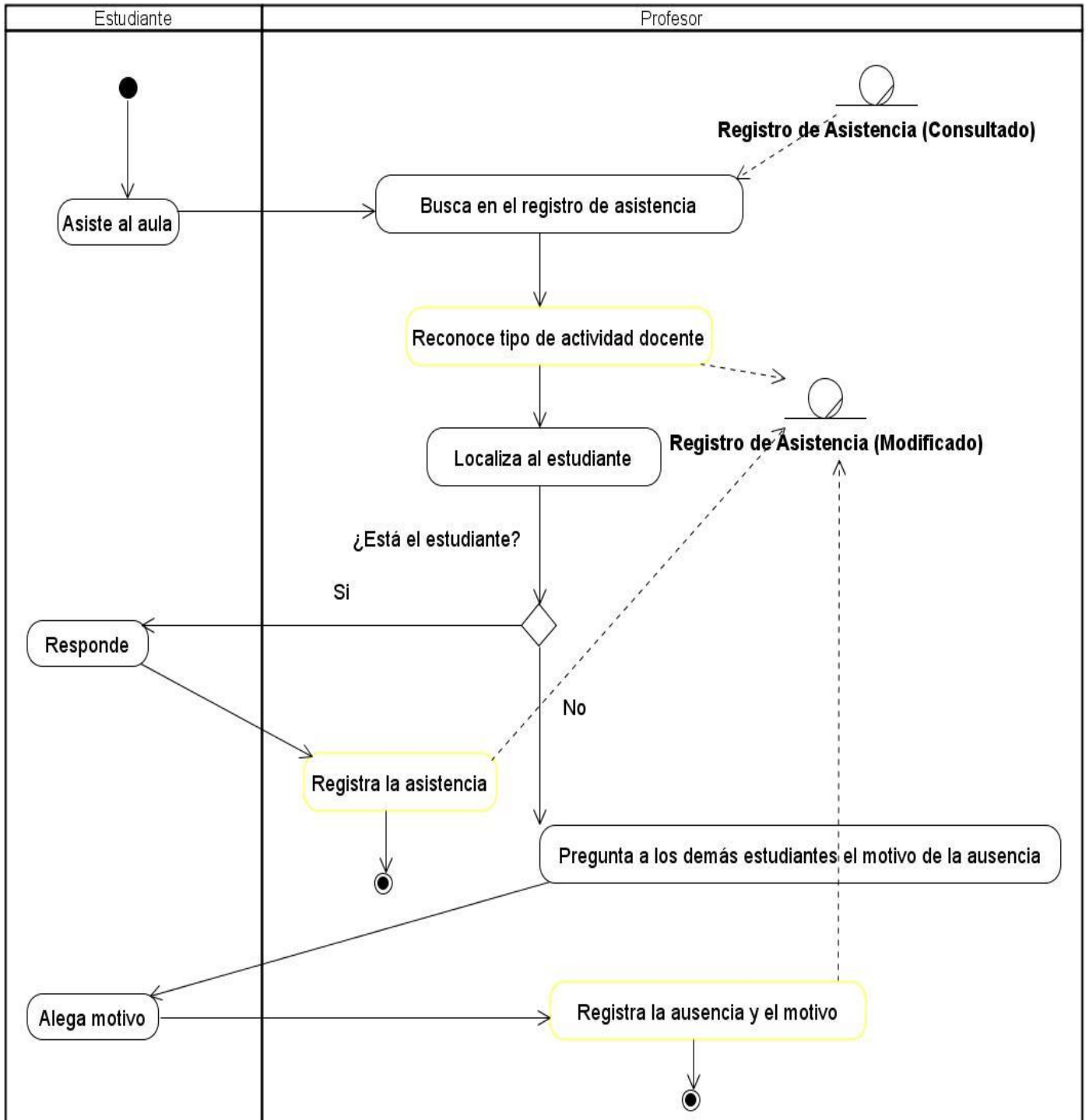


Figura 47. Diagrama de actividades del caso de uso del negocio "Controlar Asistencia"

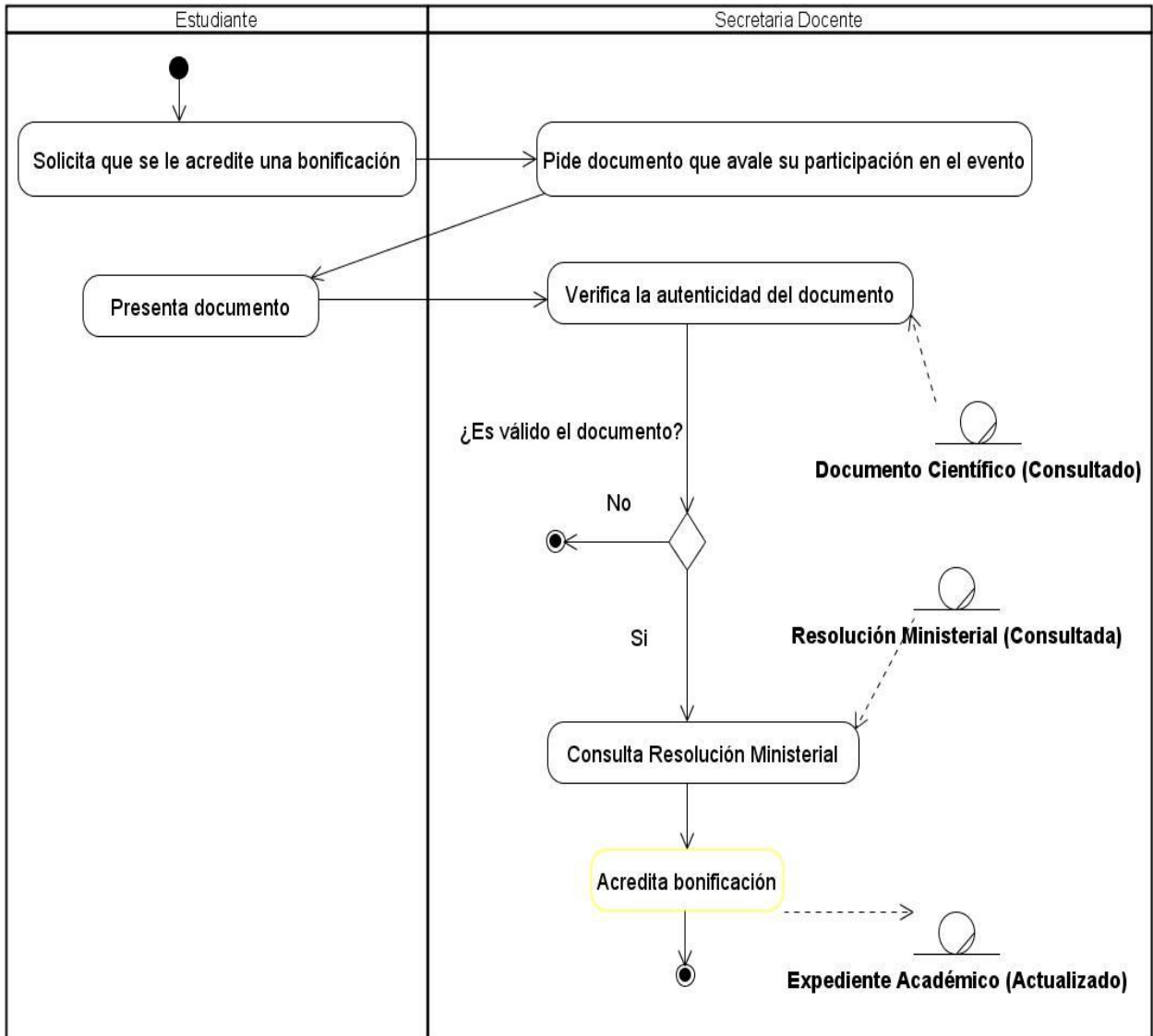


Figura 48. Diagrama de actividades del caso de uso del negocio “Acreditar Bonificación”

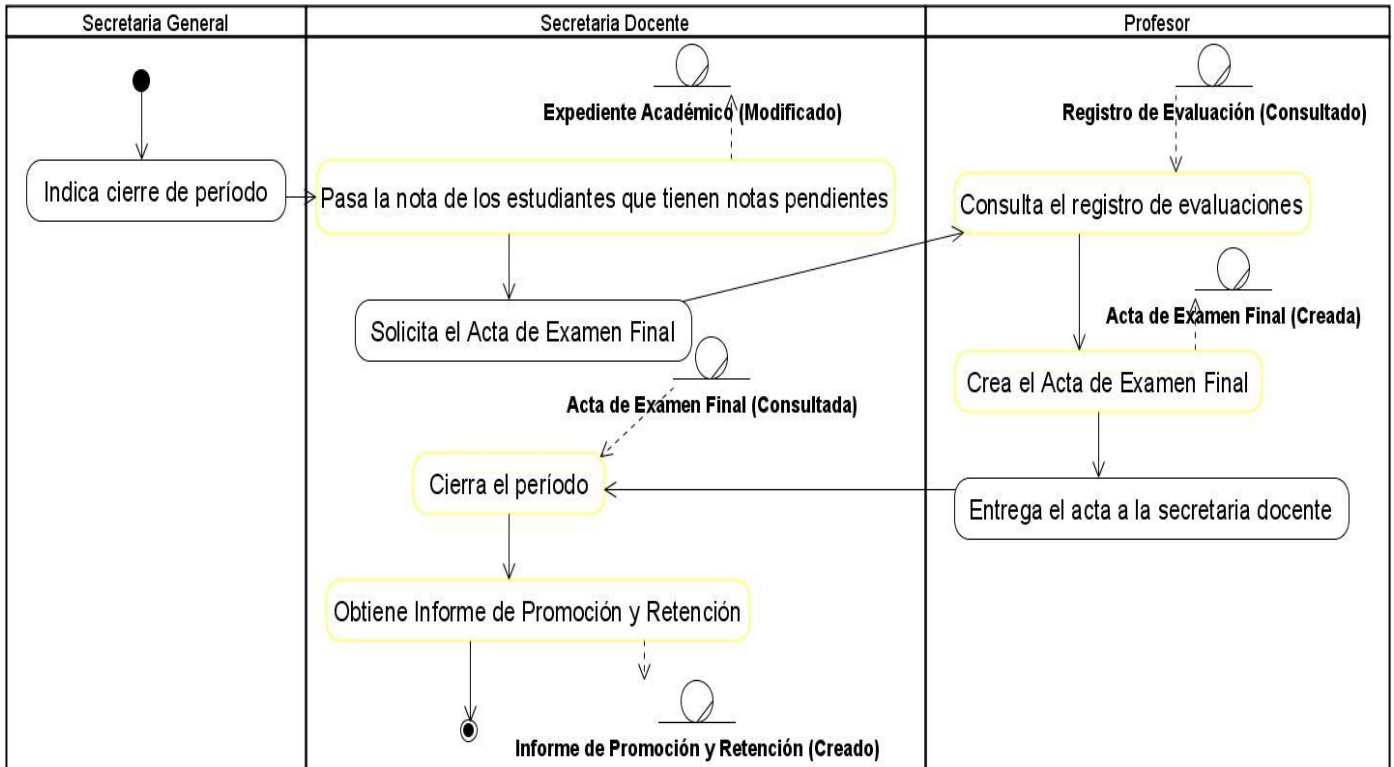


Figura 49. Diagrama de actividades del caso de uso del negocio “Cerrar Período”

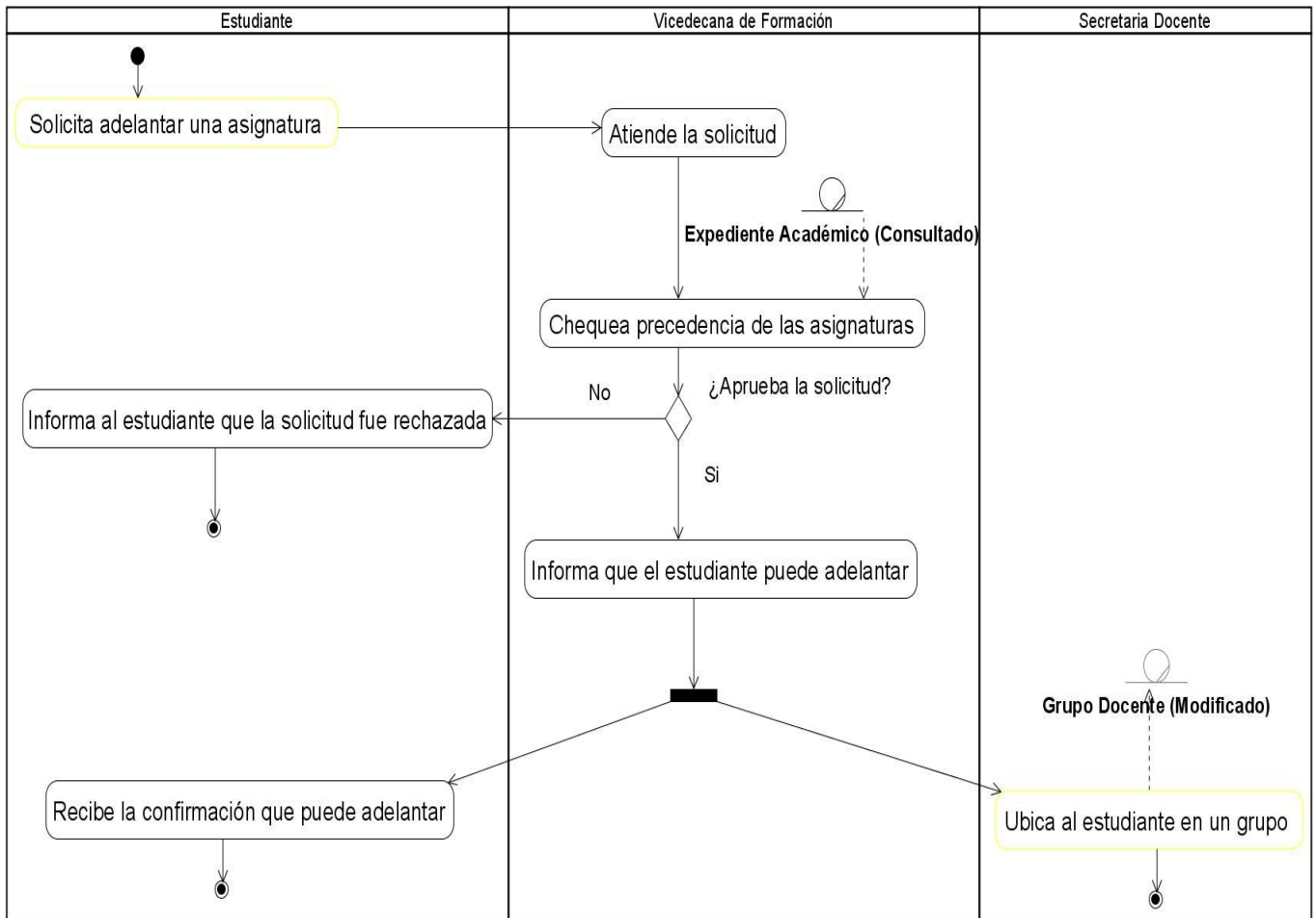


Figura 50. Diagrama de actividades del caso de uso del negocio “Solicitar Asignatura de Adelanto”

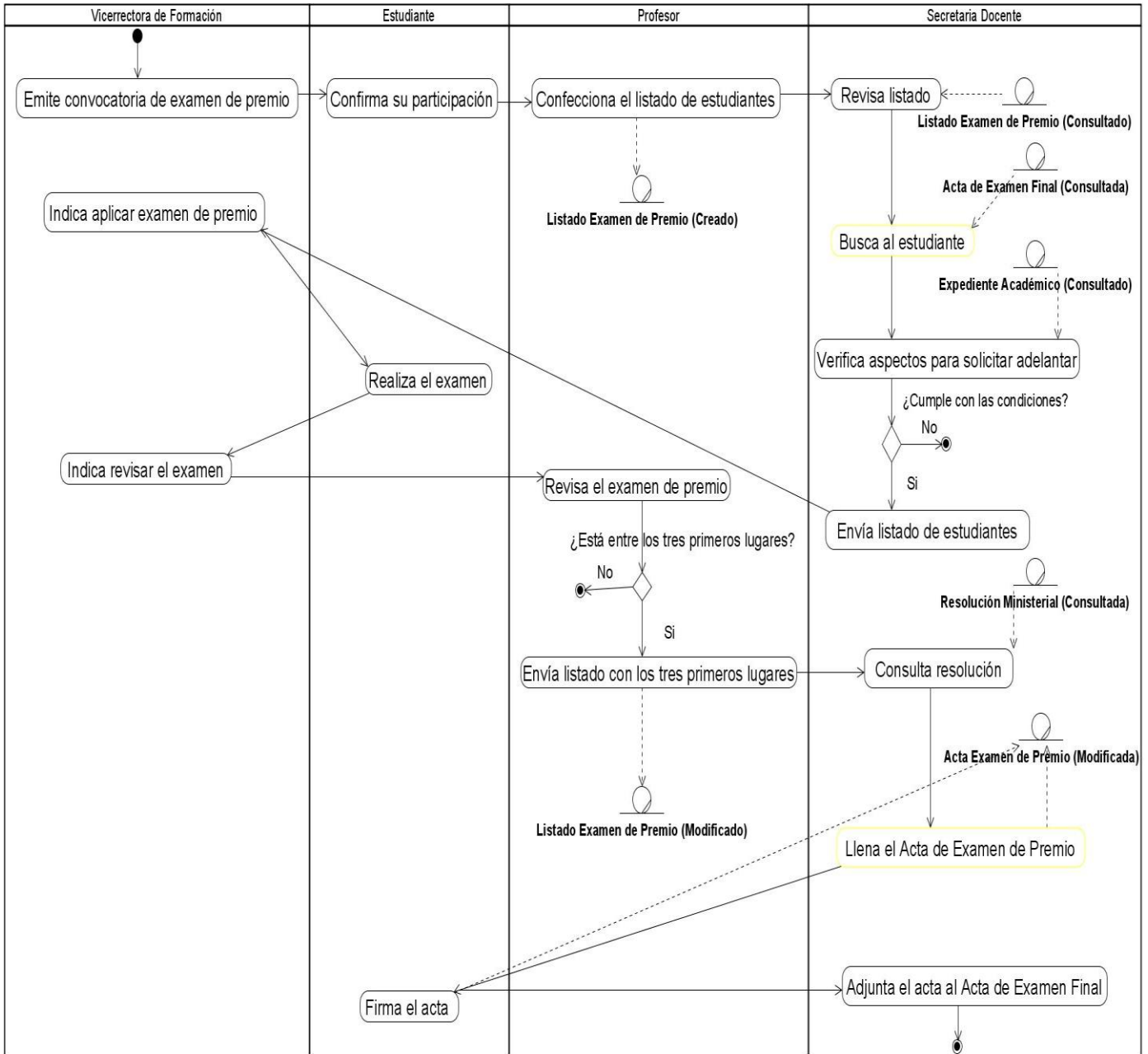


Figura 51. Diagrama de actividades del caso de uso del negocio “Acreditar Premio”

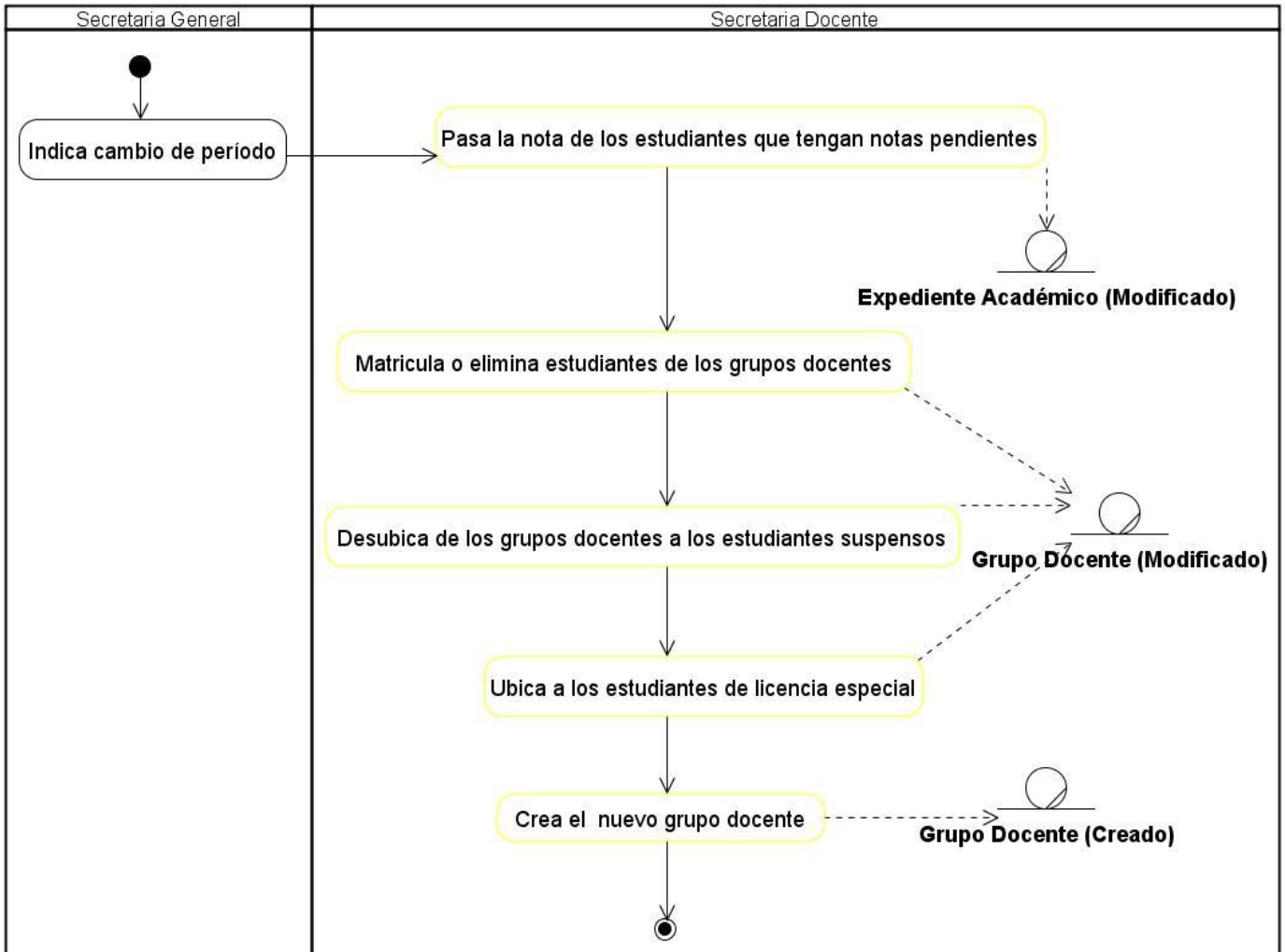


Figura 52. Diagrama de actividades del caso de uso del negocio "Crear Grupo Docente"

Anexo 3: Especificación de requisitos

RF 1. Crear Grupo Docente

- 1.1. Mostrar la interfaz visual con las opciones posibles para crear un grupo docente a partir de un grupo administrativo o no.
- 1.2. Solicitar la información para crear el grupo docente en dependencia de la opción seleccionada.
 - 1.2.1. Si es un grupo docente dependiente de un grupo administrativo, se debe almacenar la asignatura, el nombre del grupo docente y el plan de estudio.
- 1.3. Matricular en cada grupo docente creado a los estudiantes del grupo administrativo seleccionado.
 - 1.3.1. Si es un grupo docente independiente de un grupo administrativo, el sistema debe almacenar, la asignatura seleccionada, el plan de estudio, nombre del grupo docente, el período y el(los) profesor(es).
- 1.4. Almacenar la asignatura, el nombre del grupo docente, el plan de estudio, período y el(los) profesor(es) del nuevo grupo docente creado.
 - 1.4.1. Si el grupo docente ya existe, el sistema debe notificar y brindar la posibilidad de crear el grupo nuevamente.
- 1.5. Notificar que el grupo docente fue creado satisfactoriamente.

RF 2. Modificar Grupo Docente

- 2.1. Mostrar los grupos docentes creados.
 - 2.1.1. Si no hay ningún grupo docente creado el sistema debe notificarlo.
- 2.2. Seleccionar un grupo docente.
- 2.3. Mostrar un formulario con los datos modificables del grupo docente seleccionado.
 - 2.3.1. Debe permitir modificar al profesor siempre, independientemente que el grupo docente sea asociado a un grupo administrativo o no.
 - 2.3.2. En caso que sea un grupo docente independiente de un grupo administrativo debe permitir modificar, además, su nombre.
- 2.4. Notificar que se modificó el grupo docente satisfactoriamente.

RF 3. Eliminar Grupo Docente

- 3.1. Mostrar los grupos docentes creados.

3.1.1. Si no hay ningún grupo docente creado el sistema debe notificarlo.

3.2. Seleccionar un grupo docente.

3.3. Indicar una alerta que el usuario está seguro que quiere hacer válida la acción.

3.4. Notificar que se eliminó satisfactoriamente el grupo docente.

RF 4. Matricular Grupo Docente

4.1. Buscar estudiante por criterio.

4.2. Listar los estudiantes que cumplan con el criterio de búsqueda y el grupo docente al cual pertenecen.

4.2.1. Si no existen resultados asociados a la búsqueda, el sistema debe notificarlo y permitir realizar la búsqueda nuevamente.

4.3. Seleccionar un estudiante.

4.4. Matricular al estudiante en el grupo docente.

4.4.1 En caso que pertenezca a otro grupo docente mostrar un mensaje y eliminarlo del grupo.

4.5. Notificar que se matriculó satisfactoriamente al estudiante en el grupo docente.

RF 5. Desmatricular Grupo Docente

5.1. Seleccionar un grupo docente.

5.2. Listar los estudiantes asociados al grupo docente seleccionado.

5.3. Seleccionar uno o varios estudiantes.

5.4. Eliminar el(los) estudiante(s) del grupo docente.

5.5. Brindar la opción de matricular el(los) estudiante(s) en otro grupo docente.

5.6. Listar los grupos docentes con sus matrículas en los cuales puede ser adicionado el(los) estudiante(s).

5.7. Adicionar el(los) estudiante(s) en el grupo docente seleccionado.

5.8. Notificar que se desmatriculó satisfactoriamente el(los) estudiante(s) del grupo docente.

RF 6. Consultar Grupo Docente

6.1. Seleccionar un plan de estudio.

6.2. Listar las asignaturas asociadas al plan de estudio.

6.3. Seleccionar una asignatura.

6.4. Listar los grupos docentes asociados a la asignatura con sus matrículas.

6.4.1. En caso de no existir ningún grupo docente asociado a la asignatura seleccionada mostrar un mensaje.

6.5. Seleccionar un grupo docente.

6.6. Mostrar listados de estudiantes matriculados en el grupo docente seleccionado.

RF 7. Solicitar Asignatura de Adelanto

7.1. Mostrar un listado de las asignaturas que puede adelantar el estudiante separadas por año.

7.1.1. Si la lista de las asignaturas está vacía, el sistema debe notificarlo.

7.2. Seleccionar asignaturas de adelanto.

7.2.1. Si el estudiante no marcó ninguna asignatura, el sistema debe informárselo y brindarle la posibilidad de que lo haga.

7.3. Almacenar por cada asignatura seleccionada, el nombre del estudiante, el nombre de la asignatura y la fecha de la solicitud.

7.4. Listar las asignaturas seleccionadas.

RF 8. Aprobar Solicitud de Adelanto de Asignatura

8.1. Listar el nombre de los estudiantes con las asignaturas que hicieron solicitud.

8.1.1. Si la lista de las asignaturas está vacía, el sistema debe informarlo y terminar la acción.

8.2. Seleccionar una asignatura.

8.3. Listar los grupos docentes relacionados con esa asignatura y sus matrículas.

8.3.1. Si la lista de los grupos docentes está vacía el sistema debe informarlo y terminar la acción.

8.4. Seleccionar un grupo docente.

8.5. Matricular al estudiante en el grupo docente seleccionado.

8.6. Eliminar las solicitudes que fueron valoradas.

RF 9. Rechazar Solicitud de Adelanto de Asignatura

9.1. Listar el nombre de los estudiantes con la asignatura en la cual hicieron la solicitud.

9.1.1. En caso de no haber ninguna solicitud mostrar mensaje.

9.2. Marcar las solicitudes que serán rechazadas.

9.3. Eliminar solicitud de la base de datos.

9.4. Mostrar mensaje de confirmación.

RF 10. Controlar Asistencia

10.1. Mostrar un formulario con el listado de estudiantes asociados a un grupo docente donde se encuentren los campos: nombre del estudiante, fecha de la actividad, tipo de asistencia, argumento, la cantidad de horas clases y las opciones “Registrar” y “Cancelar”.

10.2. Permitir cambiar la fecha.

10.3. Validar que la fecha introducida coincida con la fecha actual.

10.4. Indicar por cada estudiante el estado de su asistencia, la cantidad de horas clases y el argumento en caso necesario.

10.5. Almacenar la fecha por cada estudiante, el estado de su asistencia, la cantidad de horas clases y el argumento si lo tiene.

10.6. Mostrar el registro de asistencia del grupo docente actualizado.

RF 11. Modificar Control Asistencia

11.1. Mostrar un formulario para modificar una asistencia en el registro del grupo docente con el listado de los estudiantes.

11.2. Permitir modificar la fecha.

11.3. Validar que la fecha introducida coincida con la fecha actual.

11.4. Modificar por cada estudiante el estado de su asistencia y el argumento en caso necesario.

11.5. Almacenar los cambios.

11.6. Actualizar el registro de asistencia.

RF 12. Eliminar Control Asistencia

12.1. Mostrar las actividades y la asistencia de cada actividad.

12.2. Seleccionar la actividad a la que se le va a eliminar el pase de asistencia.

12.3. Mostrar confirmación de la acción solicitada.

12.4. Eliminar el pase de asistencia seleccionado.

12.5. Mostrar el registro de asistencia actualizado.

RF 13. Mostrar Registro de Asistencia

13.1. Listar los grupos docentes asociados al usuario.

13.2. Seleccionar un grupo docente.

13.3. Mostrar un listado con los estudiantes del grupo docente seleccionado, las asistencias registradas para cada estudiante, la fecha en que fueron realizadas y el por ciento de asistencia en los estados presente, ausente e injustificado.

RF 14. Controlar Evaluación

14.1. Listar los grupos docentes asignados al profesor.

14.2. Seleccionar un grupo docente.

14.3. Mostrar un formulario con el tipo de evaluación y la evaluación.

14.4. Seleccionar el tipo de evaluación y la evaluación correspondiente.

14.5. Listar los estudiantes del grupo docente y un campo para seleccionar la evaluación correspondiente a cada uno de los estudiantes.

14.6. Actualizar la fecha.

14.7. Validar que la fecha insertada corresponda con la fecha actual.

14.7.1. Si la fecha no es válida se muestra un mensaje indicándole el error.

14.8. Mostrar un formulario para la inserción de la evaluación a cada uno de los estudiantes.

14.9. Permitir al profesor insertar la evaluación a cada uno de los estudiantes del grupo docente correspondiente.

14.10. Almacenar la fecha, tipo de evaluación, la evaluación y la nota para cada uno de los estudiantes del grupo docente.

14.11. Mostrar el registro de evaluaciones del grupo docente actualizado.

RF 15. Modificar Control Evaluación

15.1. Listar los grupos docentes asignados al profesor.

15.2. Seleccionar un grupo docente.

15.3. Mostrar un listado con los estudiantes del grupo docente seleccionado y sus respectivas evaluaciones.

15.4. Permitir al profesor modificar la fecha de la actividad y la nota de una evaluación de uno o varios estudiantes.

15.5. Verificar que la fecha insertada corresponda con la fecha actual.

15.5.1. Si la fecha no es válida muestra un mensaje indicándole el error.

15.6. Almacenar la fecha y la nota de la evaluación para cada uno de los estudiantes del grupo docente.

15.7. Mostrar el registro de evaluaciones del grupo docente actualizado.

RF 16. Mostrar Registro de Evaluación

16.1. Listar los grupos docentes asociados al usuario.

16.2. Seleccionar un grupo docente.

16.3. Mostrar un listado con los estudiantes del grupo docente seleccionado y sus respectivas evaluaciones.

RF 17. Acreditar Crédito Premio

17.1. Seleccionar un estudiante.

17.2. Mostrar un formulario con los campos: fecha, período, asignatura y nota.

17.3. Seleccionar los datos.

17.3.1. Si se deja algún campo vacío mostrar un mensaje de alerta.

17.4. Notificar al usuario que se asignó satisfactoriamente el premio.

17.5. Brindar la posibilidad de acreditar otro premio.

RF 18. Modificar Crédito Premio

18.1. Listar todas las asignaturas en las cuales el estudiante ha obtenido premio.

18.1.1. En caso de no tener ningún premio registrado mostrar un mensaje de notificación.

18.2. Seleccionar una asignatura.

18.3. Mostrar los campos: fecha, período, asignatura y nota, donde sólo el campo nota debe estar habilitado.

18.3.1. Si los datos introducidos están correctos el sistema debe guardarlos en la base de datos.

18.4. Notificar al usuario que se modificó satisfactoriamente el premio.

RF 19. Eliminar Crédito Premio

19.1. Mostrar todos los premios que ha obtenido un estudiante.

19.1.1. En caso de no tener registrado ningún premio mostrar un mensaje de notificación.

19.2. Seleccionar uno o varios premios.

19.3. Eliminar el premio seleccionado de la base de datos.

19.4. Notificar al usuario que se eliminó satisfactoriamente el premio.

RF 20. Acreditar Crédito Bonificación

20.1. Seleccionar un estudiante.

20.2. Mostrar un formulario con los campos: fecha, período y valor.

20.3. Seleccionar los datos.

20.3.1. Si se deja algún campo vacío mostrar un mensaje de alerta.

20.4. Mostrar un mensaje de que se acreditó correctamente la bonificación.

20.5. Brindar la posibilidad de acreditar otra bonificación.

RF 21. Modificar Crédito Bonificación

21.1. Listar todas las bonificaciones que le han sido registradas al estudiante.

21.1.1. En caso de no tener ninguna bonificación registrada mostrar un mensaje de notificación.

21.2. Seleccionar una bonificación.

21.3. Mostrar los campos: fecha, período y valor, donde sólo la nota debe estar habilitado.

21.3.1. Si los datos introducidos están correctos el sistema debe guardarlos en la base de datos.

21.4. Notificar al usuario que se modificó satisfactoriamente la bonificación.

RF 22. Eliminar Crédito Bonificación

22.1. Mostrar todas las bonificaciones que ha obtenido un estudiante.

22.1.1 En caso de no tener registrada ninguna bonificación mostrar un mensaje de notificación.

22.2. Seleccionar una o varias bonificaciones.

22.3. Eliminar la bonificación seleccionada de la base de datos.

22.4. Notificar al usuario que se eliminó satisfactoriamente.

RF 23. Asignar Profesor a Grupo Docente

23.1. Seleccionar un grupo docente.

23.2. Mostrar un formulario con los campos departamento, profesor, asignatura y período.

23.2.1. En caso de ser un grupo docente creado a partir de un grupo administrativo los campos departamento, asignatura y período aparecen por defecto.

23.2.2. En caso de ser un grupo docente creado independiente de un grupo administrativo se deben introducir todos los datos.

23.3. Seleccionar los datos.

23.4. Mostrar un mensaje de notificación si el profesor fue asignado correctamente.

23.4.1. En caso que el profesor esté asignado al grupo mostrar un mensaje de alerta.

RF 24. Mostrar Expediente del Estudiante

24.1. Seleccionar un estudiante.

24.2. Mostrar los documentos de matrícula, ciclo profesional y ciclo básico del estudiante.

RF 25. Generar Grupo Docente Auto

25.1. Seleccionar un grupo docente.

25.1.1. En caso que el grupo docente sea creado independiente de un grupo administrativo, mostrar un mensaje de alerta.

25.2. Matricular automáticamente los estudiantes asociados al grupo administrativo.

25.3. Listar los estudiantes asociados al grupo administrativo.

RF 26. Mostrar Resumen Histórico

26.1. Seleccionar una estructura administrativa.

26.2. Listar los estudiantes asociados a la estructura administrativa seleccionada.

26.3. Mostrar las asignaturas por semestre con sus respectivas evaluaciones de cada uno de los estudiantes.

RF 27. Acreditar Crédito por Actividades

27.1. Seleccionar un estudiante.

27.2. Mostrar un formulario para seleccionar los campos: fecha, período, tipo de actividad, la actividad y crédito.

27. 3. Seleccionar los datos.

27.3.1. Si se deja algún campo vacío mostrar un mensaje de alerta.

27.4. Mostrar un mensaje de que se acreditó correctamente el crédito.

27.5. Brindar la posibilidad de acreditar otro crédito.

RF 28. Modificar Crédito por Actividades

28.1. Listar todas las actividades que han aportado crédito al estudiante.

28.1.1. En caso de no tener ninguna actividad registrada mostrar un mensaje de notificación.

28.2. Seleccionar una actividad.

28.3. Mostrar los campos: fecha, período, tipo de actividad, la actividad y crédito, donde sólo el valor del crédito debe estar habilitado.

28.3.1. Si los datos introducidos están correctos el sistema debe guardarlos en la base de datos.

28.4. Notificar al usuario que se modificó satisfactoriamente el crédito.

RF 29. Eliminar Crédito por Actividades

29.1. Mostrar todos los créditos que ha obtenido un estudiante.

29.1.1 En caso de no tener registrado ningún crédito mostrar un mensaje de notificación.

29.2. Seleccionar uno o varios créditos.

29.3. Eliminar el crédito seleccionado de la base de datos.

29.4. Notificar al usuario que se eliminó satisfactoriamente.

RF 30. Crear Modelo de Corte

30.1. Seleccionar un grupo docente.

30.2. Listar los estudiantes matriculados en el grupo docente seleccionado.

30.3. Generar el modelo de corte asociado al grupo docente con el listado de los estudiantes.

RF 31. Crear Registro Asistencia

31.1. Seleccionar un grupo docente.

31.2. Listar los estudiantes matriculados en el grupo docente seleccionado.

31.3. Generar el registro de asistencia asociado al grupo seleccionado con la matrícula del grupo docente.

RF 32. Crear Acta Examen Final

32.1. Seleccionar un grupo docente.

32.2. Listar los estudiantes matriculados en el grupo docente seleccionado.

32.3. Generar el Acta de Examen Final asociada al grupo seleccionado con la matrícula del grupo docente.

RF 33. Mostrar Notas Históricas Básicas

33.1. Seleccionar un estudiante.

33.2. Listar dividido por semestre y año las asignaturas con sus notas, por cual concepto fue registrada y el período en la cual se registró.

RF 34. Mostrar Notas Históricas Optativas

34.1. Seleccionar un estudiante.

34.2. Listar los cursos optativos con sus notas.

RF 35. Mostrar Notas Históricas Finales

35.1. Seleccionar un estudiante.

35.2. Seleccionar período.

35.3. Listar asignaturas asociadas al período seleccionado.

35.4. Mostrar el transcurso del estudiante en la asignatura.

RF 36. Mostrar Hoja de Resultados Académicos

36.1. Seleccionar un estudiante.

36.2. Seleccionar año, curso, facultad y grupo docente.

36.3. Listar por curso las asignaturas y sus evaluaciones en ordinario, extraordinario, especial, premio y final que ha obtenido el estudiante.

RF 37. Mostrar Certificación de Notas

37.1. Seleccionar un estudiante.

37.2. Mostrar por cada curso académico, el nivel que venció el estudiante, el curso en que lo venció, las asignaturas obligatorias que recibió con sus evaluaciones.

37.3. Mostrar las asignaturas optativas cursadas, exámenes de premios, bonificaciones y sus evaluaciones.

37.4. Mostrar la fecha de graduación, si es Título de Oro y el promedio académico.

37.5. Mostrar nombre y apellidos de la secretaria docente, el nombre de la institución, nombre y apellidos del estudiante, la carrera de la cual se gradúa y la fecha de emisión de la certificación.

RF 38. Mostrar Promoción por Asignatura

38.1. Seleccionar asignatura.

38.2. Seleccionar año.

38.3. Mostrar el por ciento de aprobados y desaprobados para cada asignatura de cada año.

RF 39. Mostrar Estudiantes Pendientes

39.1. Mostrar listado de asignaturas.

39.2. Seleccionar asignatura.

39.3. Mostrar listado de estudiantes pendientes en la asignatura seleccionada.

RF 40. Mostrar Listado de Aprobados para Asignatura de Adelanto

40.1. Mostrar listado de asignaturas.

40.2. Seleccionar una asignatura.

40.3. Mostrar listado de estudiantes aprobados para adelantar la asignatura seleccionada.

RF 41. Mostrar Promoción por Año por Asignatura

41.1. Seleccionar un año académico.

41.2. Seleccionar asignatura.

41.3. Mostrar el por ciento de aprobados y desaprobados para cada asignatura del año académico seleccionado.

RF 42. Desubicar Profesor de Grupo Docente

42.1. Seleccionar un grupo docente.

42.2. Mostrar los datos del profesor asignado al grupo docente seleccionado.

42.3. Eliminar el profesor del grupo docente.

42.3.1. Mostrar un mensaje de notificación si el profesor fue desubicado correctamente del grupo docente.

42.4. Guardar los cambios realizados en la base de datos.

RF 43. Mostrar Aprobados por Convocatorias Prueba Nivel

43.1. Seleccionar asignatura.

43.2. Seleccionar convocatoria de prueba de nivel de la asignatura seleccionada.

43.2. Mostrar un formulario donde se especifiquen los aprobados, desaprobados y pendientes de la convocatoria seleccionada.

Anexo 4: Descripción textual detallada de los casos de uso del sistema

Tabla 4.1. Especificación del caso de uso del sistema “Gestionar Grupo Docente”

Caso de uso	
CUS-1	Gestionar Grupo Docente
Propósito	Permitir realizar la acción de crear un grupo docente, modificar un grupo docente o eliminar un grupo docente.
Actor Secretaria Docente	
Resumen: El caso de uso se inicia cuando la secretaria docente necesita realizar una operación sobre determinado grupo docente. El sistema permite seleccionar las opciones crear, modificar o eliminar un grupo docente. El caso de uso finaliza al concluir las acciones correspondientes a la opción seleccionada.	
Referencias:	RF1, RF2, RF3
Precondiciones:	Para modificar y eliminar un grupo docente se debe conocer dicho grupo docente de antemano. Para crear un grupo docente los estudiantes deben estar en estado de matriculados.
Flujo Normal de Eventos	
Sección “General”	
Acción del actor	Respuesta del sistema
1. La secretaria docente solicita gestionar un grupo docente.	2. El sistema muestra “Crear Independiente de Administrativo”, “Crear Dependiente de Administrativo”, “Modificar grupo docente” y “Eliminar grupo docente”.
3. La secretaria docente selecciona una de las opciones presentadas.	4.1. Si selecciona “Crear Independiente de Administrativo” ver sección Independiente de administrativo. 4.2. Si selecciona “Crear Dependiente de Administrativo” ver sección Dependiente de administrativo. 4.3. Si selecciona “Modificar grupo docente” ver sección Modificar Grupo Docente. 4.4. Si selecciona “Eliminar grupo docente” ver

	sección Eliminar Grupo Docente.
Sección “Dependiente de administrativo”	
Acción del actor	Respuesta del sistema
1. La secretaria docente selecciona la opción “Dependiente de administrativo”.	2. El sistema muestra un mensaje indicándole seleccionar una estructura administrativa.
3. La secretaria docente selecciona un grupo administrativo.	4. El sistema muestra un listado de las asignaturas básicas y optativas asociadas al grupo administrativo.
5. La secretaria docente selecciona una asignatura.	6. El sistema automáticamente crea el grupo docente.
	7. El sistema muestra el listado de estudiantes asociados al grupo.

Prototipo de Interfaz de usuario

The screenshot shows the user interface for the 'Sistema de Gestión Universitaria'. At the top left is the 'AKADEMOS' logo. The main header reads 'Sistema de Gestión Universitaria'. Below this is a 'Bienvenido:' message and a 'Salir' button. A navigation bar contains buttons for 'Pregrado', 'Posgrado', 'Producción', and 'Investigaciones'. Below the navigation bar is a breadcrumb trail: 'Estructura | Personal | Carrera | Control Docente | Generador de Plantillas'. The current page path is 'Pregrado\Control Docente\Grupo Docente por Asignatura\Asignaturas por Grupo'. The main content area is titled 'Asignaturas por Grupo - Semestre Actual'. It features a large list of subjects, divided into 'Asignaturas Básicas' and 'Asignaturas Optativas'. The 'Asignaturas Básicas' list includes: Educación Física I, Idioma Extranjero I, Matemática I, Matemática Discreta, and Introducción a la Programación. To the right of the subject list is a sidebar with a 'Plan de Estudio' dropdown menu (currently showing 'D') and a menu of actions: Gestionar Grupo Docente, Expediente del Estudiante, Crédito por Actividades, Bonificaciones, Premio, Asignaturas de Adelanto, Solicitar Asignatura Adelanto, and Grupos Docentes por Asignatura.

Sección "Independiente de administrativo"	
Acción del actor	Respuesta del sistema
1. La secretaria docente selecciona la opción "Independiente de administrativo".	2. El sistema muestra un formulario para indicar plan de estudio, asignatura, período y nombre.
3. La secretaria docente selecciona los datos.	4. El sistema guarda los datos en la base de datos.
	5. El sistema crea el grupo docente.

Prototipo de Interfaz de usuario

The screenshot displays the user interface for creating an administrative teaching group. At the top, the logo 'AKADEMOS' and the title 'Sistema de Gestión Universitaria' are shown. Below this is a navigation bar with tabs for 'Pregrado', 'Posgrado', 'Producción', and 'Investigaciones'. A breadcrumb trail indicates the current path: 'Pregrado \ Control Docente \ Gestionar Grupo Docente \ Crear Grupo Docente Independiente de Administrativo'. The main content area is titled 'Crear Grupo Docente Independiente de Administrativo' and contains a form with the following elements:

- Form Fields:**
 - Grupo Docente * (text input)
 - Plan de Estudio * (dropdown menu)
 - Asignatura * (dropdown menu)
 - Período * (dropdown menu)
- Buttons:** 'Aceptar' and 'Cancelar' at the bottom of the form.
- Plan de Estudio:** A separate dropdown menu above the sidebar.
- Sidebar Menu:**
 - Gestionar Grupo Docente
 - Expediente del Estudiante
 - Crédito por Actividades
 - Bonificaciones
 - Premio
 - Asignaturas de Adelanto
 - Solicitar Asignatura Adelanto
 - Grupos Docentes por Asignatura

Sección "Modificar Grupo Docente"	
Acción del actor	Respuesta del sistema
1. La secretaria docente selecciona la opción "Modificar grupo docente".	2. El sistema muestra un formulario con los datos del grupo docente.
3. La secretaria docente modifica lo que necesita y selecciona la opción "Aceptar".	4. El sistema almacena la información modificada.

	5. El sistema notifica que se modificó el grupo docente satisfactoriamente y finaliza el caso de uso.
--	---

Flujo alternativo 1

Acción del actor	Respuesta del sistema
3. Si la secretaria docente deja campos vacíos, el sistema lo notifica y regresa al paso 2.	

Prototipo de Interfaz de usuario

The screenshot shows the 'Sistema de Gestión Universitaria' interface. At the top left is the 'AKADEMOS' logo. The main header reads 'Sistema de Gestión Universitaria'. Below this is a navigation bar with 'Pregrado', 'Posgrado', 'Producción', and 'Investigaciones' tabs. A breadcrumb trail shows 'Control Docente \ Gestionar Grupo Docente \ Modificar grupo docente'. The main content area is titled 'Modificar Datos Grupo Docente' and contains a form with a user profile picture and details: 'Nombre: Damián Cervantes Rodón', 'Área: Facultad 1', and 'Cargo: Instructor Recién Graduado'. The form has four input fields: 'Grupo Docente' (empty), 'Profesor' (dropdown), 'Plan de Estudio' (dropdown with 'D' selected), and 'Asignatura' (dropdown with 'ISW' selected). There are 'Aceptar' and 'Cancelar' buttons at the bottom of the form. To the right is a sidebar menu with 'Plan de Estudio' (dropdown with 'D') and a list of options: 'Gestionar Grupo Docente', 'Expediente del Estudiante', 'Crédito por Actividades', 'Bonificaciones', 'Premio', 'Asignaturas de Adelanto', 'Solicitar Asignatura Adelanto', and 'Grupos Docentes por Asignatura'. Below the menu is a 'Buscar Estudiante' section with an input field and a 'Buscar' button.

Sección “Eliminar Grupo Docente”

Acción del actor	Respuesta del sistema
1. La secretaria docente selecciona la opción “Eliminar grupo docente”.	2. El sistema pide confirmación.

3. La secretaria docente confirma que desea eliminar el grupo docente.	4. El sistema elimina el grupo docente y notifica que la operación se realizó satisfactoriamente.
	5. El sistema guarda los cambios en la base de datos.

Prototipo de Interfaz de usuario

The screenshot shows the 'Sistema de Gestión Universitaria' interface. At the top left is the 'AKADEMOS' logo. The main navigation bar includes 'Pregrado', 'Posgrado', 'Producción', and 'Investigaciones'. Below this is a secondary navigation bar with 'Estructura', 'Personal', 'Carrera', 'Control Docente', and 'Generador de Plantillas'. The current page is 'Control Docente' > 'Eliminar Grupo Docente'. A central dialog box with a large 'X' icon asks '¿Está seguro que desea eliminar este grupo docente?' with 'Aceptar' and 'Cancelar' buttons. On the right, a sidebar menu titled 'Plan de Estudio' contains a dropdown menu and a list of actions: 'Asignar Profesor a Grupo', 'Desubicar Profesor de Grupo', 'Pasar Asistencia', 'Modificar Asistencia', 'Eliminar Asistencia', 'Ver Registro de Asistencia', 'Registrar Evaluación', 'Ver Registro de Evaluación', 'Crear Acta Examen Final', 'Matricular Estudiantes', and 'Desmatricular Estudiantes'.


Poscondiciones:	-
Prioridad:	Crítico

Tabla 4.2. Especificación del caso de uso del sistema “Gestionar Matrícula en Grupo Docente”

Caso de uso	
CUS-2	Gestionar Matrícula en Grupo Docente
Propósito	Permitir realizar la acción de matricular estudiantes en un grupo docente o desmatricular estudiantes de un grupo docente.
Actores	
Secretaria Docente	

Resumen:	
El caso de uso se inicia cuando la secretaria docente decide matricular uno o varios estudiantes en un grupo docente, cambiar estudiante de grupo docente o desmatricular estudiante de un grupo docente. El sistema permite realizar la opción seleccionada finalizando el caso de uso.	
Referencias:	RF4, RF5
Precondiciones:	-
Flujo Normal de Eventos	
Sección "General"	
Acción del actor	Respuesta del sistema
1. La secretaria docente solicita gestionar la matrícula de un grupo docente.	2. El sistema muestra "Matricular estudiantes" y "Desmatricular estudiantes".
3. La secretaria docente selecciona una de las opciones presentadas.	4.1. Si selecciona "Matricular estudiantes" ver sección Matricular Grupo Docente. 4.2. Si selecciona "Desmatricular estudiantes" ver sección Desmatricular Grupo Docente.
Sección "Matricular Grupo Docente"	
Acción del actor	Respuesta del sistema
1. La secretaria docente selecciona "Matricular estudiantes".	2. El sistema muestra un formulario para realizar la búsqueda de un estudiante.
3. La secretaria docente introduce el nombre del estudiante.	4. El sistema muestra un formulario con el listado de los estudiantes encontrados de acuerdo a la búsqueda.
5. La secretaria docente selecciona un estudiante.	6. El sistema pide confirmación para matricularlo en el grupo docente.
7. La secretaria docente selecciona la opción "Aceptar".	8. El sistema matricula al estudiante en el grupo docente y lo elimina del que estaba.
	9. El sistema actualiza los datos en la base de datos.
Flujo alternativo 1	
Acción del actor	Respuesta del sistema
7. Si la secretaria docente no selecciona que desea matricularlo, regresa al paso 4.	

Prototipo de Interfaz de usuario


Sistema de Gestión Universitaria

Bienvenido:
Salir


Pregrado
Posgrado
Producción
Investigaciones

Estructura
Personal
Carrera
Control Docente
Generador de Plantillas

Control Docente
Matricular Estudiantes

Matricular Estudiante

Ya está recibiendo Educación Física en el grupo 01102, desea matricularlo en este grupo.



Nombre: Denis Albelo

Carnet de Identidad: 86060713132

Aceptar
Cancelar

Plan de Estudio

- Asignar Profesor a Grupo
- Desubicar Profesor de Grupo
- Passar Asistencia
- Modificar Asistencia
- Eliminar Asistencia
- Ver Registro de Asistencia
- Registrar Evaluación
- Ver Registro de Evaluación
- Crear Acta Examen Final
- Matricular Estudiantes
- Desmatricular Estudiantes

Sección “Desmatricular Grupo Docente”

Acción del actor	Respuesta del sistema
1. La secretaria docente selecciona “Desmatricular estudiantes”.	2. El sistema muestra un formulario con el listado de los estudiantes del grupo docente seleccionado.
3. La secretaria docente selecciona uno o varios estudiantes a desmatricular.	4. El sistema pide confirmación para desmatricularlos del grupo docente.
5. La secretaria docente selecciona la opción “Aceptar”.	6. El sistema muestra un listado de los grupos docentes en los cuales puede matricular al(los) estudiante(s).
7. La secretaria docente selecciona un grupo docente.	8. El sistema matricula al(los) estudiante(s) en el grupo docente seleccionado y los desmatricula del anterior.
	9. El sistema muestra la matrícula del grupo

	docente actualizada.
	10. El sistema almacena la matrícula realizada y notifica que se matriculó satisfactoriamente.
Flujo alternativo 1	
Acción del actor	Respuesta del sistema
5. Si la secretaria docente no selecciona matricularlo en otro grupo docente, pasar a la acción 2.	
Prototipo de Interfaz de usuario	
<p>The screenshot shows the 'Sistema de Gestión Universitaria' interface. At the top, there is a navigation bar with 'Pregrado', 'Posgrado', 'Producción', and 'Investigaciones'. Below it, a breadcrumb trail reads 'Control Docente \ Desmatricular Estudiante'. The main content area is titled 'Desmatricular Estudiante' and contains a dialog box with the following elements:</p> <ul style="list-style-type: none"> Message: 'Dejará de recibir Educación Física en este grupo, desea matricularlo en otro grupo.' Student photo of Denis Albelo. Text: 'Nombre: Denis Albelo' and 'Carnet de Identidad: 86060713132'. Buttons: 'Aceptar' and 'Cancelar'. Right sidebar menu: 'Plan de Estudio' (with a dropdown), 'Asignar Profesor a Grupo', 'Desubicar Profesor de Grupo', 'Pasar Asistencia', 'Modificar Asistencia', 'Eliminar Asistencia', 'Ver Registro de Asistencia', 'Registrar Evaluación', 'Ver Registro de Evaluación', 'Crear Acta Examen Final', 'Matricular Estudiantes', and 'Desmatricular Estudiantes'. 	
Poscondiciones:	-
Prioridad:	Crítico

Tabla 4.3. Especificación del caso de uso del sistema “Gestionar Control Asistencia”


Caso de uso	
CUS-3	Gestionar Control Asistencia
Propósito	Permitir realizar la acción de pasar la asistencia, modificar un pase de asistencia o eliminar un pase de asistencia.
Actores Profesor	
Resumen: El caso de uso se inicia cuando el profesor decide pasar la asistencia del grupo docente al que imparte clases. Para ello se dirige al registro de asistencia, donde puede realizar las acciones de pasar asistencia, modificar asistencia y eliminar asistencia. El caso de uso finaliza cuando realice las acciones que desea y salga del sistema.	
Referencias	RF10, RF11, RF12
Flujo Normal de Eventos	
Sección “General”	
Acción del actor	Respuesta del sistema
1. El profesor solicita gestionar el control de asistencia de un grupo docente.	2. El sistema muestra “Pasar asistencia”, “Modificar asistencia” y “Eliminar asistencia”.
3. El profesor selecciona una de las opciones presentadas.	4.1. Si selecciona “Pasar asistencia” ver sección Controlar Asistencia. 4.2. Si selecciona “Modificar asistencia” ver sección Modificar Control Asistencia. 4.3. Si selecciona “Eliminar asistencia” ver sección Eliminar Control Asistencia.
Sección “Controlar Asistencia”	
Acción del actor	Respuesta del sistema
1. El profesor selecciona la opción “Pasar asistencia”.	2. El sistema muestra un formulario con el listado de los estudiantes del grupo docente, en el cual se encuentran los campos tipo de asistencia, cantidad de horas clases, fecha y el argumento.
3. El profesor indica la fecha de la actividad, el tipo de asistencia por estudiante, la cantidad de	4. El sistema verifica si la fecha indicada es igual a la actual.

horas clases, el argumento en caso de tenerlo y una vez que haya terminado pulsa el botón "Registrar".	
	5. El sistema almacena la fecha, por cada estudiante el estado de su asistencia, la cantidad de horas clases y el argumento si lo tiene.
	6. El sistema muestra el formulario con el registro de asistencia del grupo docente actualizado.

Flujo alternativo 1

Acción del actor	Respuesta del sistema
	4. Si la fecha indicada es incorrecta, el sistema muestra un mensaje de error al profesor y regresa al paso 2.

Prototipo de Interfaz de usuario


Sistema de Gestión Universitaria





Bienvenido:

Pregrado
Posgrado
Producción
Investigaciones


Estructura | Personal | Carrera | Control Docente | Generador de Plantillas

Control Docente\Pasar Asistencia

Pasar Asistencia Fecha: 17/02/09 [Cambiar](#)

	Nombre	Horas Clases	Tipo de Asistencia	Argumento
	Hector Luis	* <input type="text" value=""/>	* <input type="text" value=""/>	* <input type="text" value=""/>
	Arnaldo Ramón	* <input type="text" value=""/>	* <input type="text" value=""/>	* <input type="text" value=""/>
	Denis Albelo	* <input type="text" value=""/>	* <input type="text" value=""/>	* <input type="text" value=""/>
	Orlando Torres	* <input type="text" value=""/>	* <input type="text" value=""/>	* <input type="text" value=""/>

Sección “Modificar Control Asistencia”	
Acción del actor	Respuesta del sistema
1. El profesor selecciona la opción “Modificar asistencia”.	2. El sistema muestra el registro de asistencia del grupo docente con el listado de los estudiantes y la fecha de la asistencia.
3. El profesor selecciona un pase de asistencia.	4. El sistema muestra un formulario con el listado de los estudiantes y los campos para modificar el pase de asistencia.
5. El profesor modifica los datos que estime pertinente y pulsa la opción “Aceptar”.	6. El sistema verifica si la fecha indicada es igual a la actual.
	7. El sistema almacena los datos modificados.
	8. El sistema muestra el registro de asistencia actualizado.
Flujo alternativo 1	
Acción del actor	Respuesta del sistema
5. El profesor no modifica ningún dato, regresa al paso 2.	
	6. Si la fecha indicada es incorrecta, el sistema muestra un mensaje de error al profesor y regresa al paso 2.
Prototipo de Interfaz de usuario	


Sistema de Gestión Universitaria

Bienvenido: Salir

Pregrado
Posgrado
Producción
Investigaciones





Estructura | Personal | Carrera | Control Docente | Generador de Plantillas

Control Docente \ Modificar Asistencia

Modificar Pase de Asistencia
Cambiar

Fecha: 17/02/09 12:00:00 AM

Plan de Estudio
 D

Foto	Nombre	Tipo de Asistencia	*	*	Argumento
	Hector Luis Rodriguez	<input type="text"/>	*	*	<input type="text"/>
	Arnaldo Ramón	<input type="text"/>	*	*	<input type="text"/>
	Denis Albelo	<input type="text"/>	*	*	<input type="text"/>
	Orlando Torres	<input type="text"/>	*	*	<input type="text"/>

Registrar
Cancelar

Asignar Profesor a Grupo
 Desubicar Profesor de Grupo
 Pasar Asistencia
 Modificar Asistencia
 Eliminar Asistencia
 Ver Registro de Asistencia
 Registrar Evaluación
 Ver Registro de Evaluación
 Crear Acta Examen Final
 Matricular Estudiantes
 Desmatricular Estudiantes

Gestionar Grupo Docente
 Expediente del Estudiante
 Crédito por Actividades
 Bonificaciones

Sección “Eliminar Control Asistencia”

Acción del actor	Respuesta del sistema
1. El profesor selecciona la opción “Eliminar asistencia”.	2. El sistema muestra el registro de asistencia del grupo docente con el listado de los estudiantes y la fecha de la asistencia.
3. El profesor selecciona el pase de asistencia que desea eliminar.	4. El sistema muestra confirmación de la acción solicitada.
5. El profesor confirma que desea eliminar el pase de asistencia.	6. El sistema elimina la asistencia del registro del grupo docente seleccionado.
	7. El sistema muestra el formulario con el registro de asistencia del grupo docente actualizado.

Flujo alternativo 1

Acción del actor	Respuesta del sistema
------------------	-----------------------

5. Si el profesor no selecciona eliminar el pase de asistencia, se regresa al paso 2.

Prototipo de Interfaz de usuario

Pregrado	Posgrado	Producción	Investigaciones	
Estructura Personal Carrera Control Docente Generador de Plantillas				
Control Docente\Registro de Asistencia				
Registro de Asistencia				
Grupo Docente 01101 - Educación Física I - 72 Horas Clase				
Nombre	17/02/09	Presente	Justificado	Injustificado
Hector Luis Rodriguez	P	1.38%	0%	0%
Adrian P	P	1.38%	0%	0%
Denis Albelo	P	1.38%	0%	0%
Orlando Torres	P	1.38%	0%	0%
Sergio Miranda	P	1.38%	0%	0%
Arnaldo Perez	P	1.38%	0%	0%
Imer Olivera	P	1.38%	0%	0%
Damian Cervantes	P	1.38%	0%	0%
Poscondiciones:	-			
Prioridad:	Crítico			

Tabla 4.4. Especificación del caso de uso del sistema “Gestionar Registro Evaluación”

Caso de uso	
CUS-4	Gestionar Registro Evaluación
Propósito	Permitir realizar la acción de pasar una evaluación o modificar una evaluación.
Actores	
Profesor	





Resumen:	
El caso de uso se inicia cuando el profesor decide gestionar una evaluación, el profesor puede agregar evaluaciones al registro y modificarlas. El sistema muestra el registro de evaluaciones y el caso de uso finaliza cuando realice las acciones que desea y salga del sistema.	
Referencias:	RF14, RF15
Precondiciones:	El profesor debe tener asignado al menos un grupo docente.
Flujo Normal de Eventos	
Sección "General"	
Acción del actor	Respuesta del sistema
1. El profesor solicita gestionar el control de la evaluación de un grupo docente.	2. El sistema muestra "Registrar evaluación" y "Modificar evaluación".
3. El profesor selecciona una de las opciones presentadas.	4.1. Si selecciona "Registrar evaluación" ver sección Controlar Evaluación. 4.2. Si selecciona "Modificar evaluación" ver sección Modificar Control Evaluación.
Sección "Controlar Evaluación"	
Acción del actor	Respuesta del sistema
1. El profesor selecciona la opción "Registrar evaluación".	2. El sistema muestra un formulario para seleccionar el tipo de evaluación y la evaluación.
3. El profesor selecciona el tipo de evaluación, la evaluación y la opción "Aceptar".	4. El sistema muestra un formulario permitiéndole al profesor insertar la evaluación a cada uno de los estudiantes y registrar la fecha.
5. El profesor especifica la evaluación a uno o a varios estudiantes, la fecha y pulsa la opción "Aceptar".	6. El sistema verifica si la fecha indicada es igual a la actual.
	7. El sistema almacena la fecha, tipo de evaluación, evaluación y la nota de cada uno de los estudiantes del grupo docente.
	8. El sistema muestra el formulario con el registro del grupo docente actualizado.
Flujo alternativo 1	
Acción del actor	Respuesta del sistema
	6. Si la fecha indicada es incorrecta, el sistema

muestra un mensaje de error al profesor y regresa al paso 4.

Prototipo de Interfaz de usuario

Registrar Evaluación

Grupo Docente 01101 - Educación Física I - Juan Perez

Foto	Nombre	Evaluación
	Hector Luis Rodriguez	<input type="text"/>
	Adrian P	<input type="text"/>
	Denis Albelo	<input type="text"/>
	Orlando Torres	<input type="text"/>

Sección "Modificar Control Evaluación"

Acción del actor	Respuesta del sistema
1. El profesor selecciona la opción "Modificar evaluación".	2. El sistema muestra el registro del grupo docente con el listado de los estudiantes y la evaluación.
3. El profesor selecciona una evaluación y pulsa la opción "Aceptar".	4. El sistema muestra un formulario con el listado de los estudiantes y los campos para modificar la evaluación.
5. El profesor especifica la nota a uno o varios estudiantes y modifica la fecha en caso necesario.	6. El sistema verifica si la fecha indicada es igual a la actual.
	7. El sistema almacena la fecha, la evaluación y la nota para cada uno de los estudiantes del grupo docente.
	8. El sistema muestra el formulario con el registro del grupo docente actualizado y finaliza el caso de uso.

Flujo alternativo 1																
Acción del actor	Respuesta del sistema															
	6. Si la fecha indicada es incorrecta, el sistema muestra un mensaje de error al profesor y regresa al paso 4.															
Prototipo de Interfaz de usuario																
<p>Modificar Registro de Evaluaciones</p> <p>Primer Corte_Grupo Docente 01101 - Educación Física I - Juan Perez</p> <table border="1"> <thead> <tr> <th>Foto</th> <th>Nombre</th> <th>Evaluación</th> </tr> </thead> <tbody> <tr> <td></td> <td>Hector Luis Rodriguez</td> <td>* <input type="text"/></td> </tr> <tr> <td></td> <td>Adrian P</td> <td>* <input type="text"/></td> </tr> <tr> <td></td> <td>Denis Albelo</td> <td>* <input type="text"/></td> </tr> <tr> <td></td> <td>Orlando Torres</td> <td>* <input type="text"/></td> </tr> </tbody> </table> <p style="text-align: center;"> <input type="button" value="Aceptar"/> <input type="button" value="Cancelar"/> </p>		Foto	Nombre	Evaluación		Hector Luis Rodriguez	* <input type="text"/>		Adrian P	* <input type="text"/>		Denis Albelo	* <input type="text"/>		Orlando Torres	* <input type="text"/>
Foto	Nombre	Evaluación														
	Hector Luis Rodriguez	* <input type="text"/>														
	Adrian P	* <input type="text"/>														
	Denis Albelo	* <input type="text"/>														
	Orlando Torres	* <input type="text"/>														
Poscondiciones:	-															
Prioridad:	Crítico															

Tabla 4.5. Especificación del caso de uso del sistema “Gestionar Crédito Premio”

Caso de uso	
CUS-5	Gestionar Crédito Premio
Propósito	Permitir realizar la acción de acreditar un premio a un estudiante, modificar un premio o eliminar un premio.
Actores	
Secretaria Docente	

Resumen:	
El caso de uso se inicia cuando la secretaria docente accede a realizar una operación de acreditar, modificar o eliminar los premios de un estudiante en una asignatura. El sistema ejecuta la acción indicada por la secretaria docente finalizando el caso de uso.	
Referencias	RF17, RF18, RF19
Flujo Normal de Eventos	
Sección "General"	
Acción del actor	Respuesta del sistema
1. La secretaria docente solicita gestionar el premio de un estudiante.	2. El sistema muestra "Acreditar premio", "Modificar premio" y "Eliminar premio".
3. La secretaria docente selecciona una de las opciones presentadas.	4.1. Si selecciona "Acreditar premio" ver sección Acreditar Crédito Premio. 4.2. Si selecciona "Modificar premio" ver sección Modificar Crédito Premio. 4.3. Si selecciona "Eliminar premio" ver sección Eliminar Crédito Premio.
Sección "Acreditar Crédito Premio"	
Acción del actor	Respuesta del sistema
1. La secretaria docente selecciona la opción "Acreditar premio".	2. El sistema muestra un formulario para seleccionar los campos: fecha, período, asignatura y nota.
3. La secretaria docente selecciona los campos y pulsa la opción "Aceptar".	4. El sistema muestra un formulario con el listado de premios obtenidos por el estudiante.
	5. El sistema notifica al usuario que se asignó satisfactoriamente el premio.
Flujo alternativo 1	
Acción del actor	Respuesta del sistema
3. La secretaria docente no llena los campos regresa al paso 2.	
Prototipo de Interfaz de usuario	

Acreditar Premio



Fecha: Día * Mes * Año *

Período: *

Asignatura *

Nota: *

Nombre: Hector Luis Rodriguez

Sección “Modificar Crédito Premio”

Acción del actor	Respuesta del sistema
1. La secretaria docente selecciona la opción “Modificar premio”.	2. El sistema muestra un formulario con todas las asignaturas en las cuales el estudiante ha obtenido premio.
3. La secretaria docente selecciona una asignatura.	4. El sistema muestra un formulario donde sólo el campo nota está habilitado.
5. La secretaria docente modifica el campo nota.	6. Si los datos introducidos están correctos, el sistema guarda los cambios en la base de datos
	7. El sistema notifica al usuario que se modificó satisfactoriamente el premio.

Flujo alternativo 1

Acción del actor	Respuesta del sistema
	2. Si el estudiante no presenta asignaturas en las cuales se le ha acreditado premio, mostrar un mensaje.

Prototipo de Interfaz de usuario

Modificar Premio



Nombre: Hector Luis Rodriguez

Fecha: Día * 7 Mes * Enero Año * 2008

Periodo: * Curso 2008_2009

Asignatura * ISW

Nota: *

Sección “Eliminar Crédito Premio”

Acción del actor	Respuesta del sistema
1. La secretaria docente selecciona la opción “Eliminar premio”.	2. El sistema muestra un formulario con el listado de los premios que ha obtenido el estudiante.
3. La secretaria docente selecciona uno o varios premios.	4. El sistema muestra confirmación que desea eliminar el(los) premio(s).
5. La secretaria docente confirma que desea eliminar el(los) premio(s).	6. El sistema elimina el(los) premio(s) de la base de datos.
	7. El sistema notifica al usuario que se eliminó satisfactoriamente el(los) premio(s).

Flujo alternativo 1

Acción del actor	Respuesta del sistema
5. Si no selecciona eliminar premio, se regresa al paso 2.	

Prototipo de Interfaz de usuario

Eliminar Premio											
 <p>Nombre: Hector Luis Rodriguez</p>	<table border="1"> <thead> <tr> <th>Periodo</th> <th>Premio(s)</th> <th>Nota</th> <th>Fecha</th> <th>Seleccione</th> </tr> </thead> <tbody> <tr> <td>2008_2009</td> <td>Premio</td> <td>0.03</td> <td>19/02/09</td> <td style="text-align: center;"><input checked="" type="checkbox"/></td> </tr> </tbody> </table> <p style="text-align: center;"> <input type="button" value="Aceptar"/> <input type="button" value="Cancelar"/> </p>	Periodo	Premio(s)	Nota	Fecha	Seleccione	2008_2009	Premio	0.03	19/02/09	<input checked="" type="checkbox"/>
Periodo	Premio(s)	Nota	Fecha	Seleccione							
2008_2009	Premio	0.03	19/02/09	<input checked="" type="checkbox"/>							
Poscondiciones:	-										
Prioridad:	Crítico										

Tabla 4.6. Especificación del caso de uso del sistema “Gestionar Crédito Bonificación”

Caso de uso	
CUS-6	Gestionar Crédito Bonificación
Propósito	Permitir realizar la acción de acreditar una bonificación a un estudiante, modificar una bonificación o eliminar una bonificación.
Actores	
Secretaria Docente	
Resumen:	
El caso de uso se inicia cuando la secretaria docente accede a realizar una operación de acreditar, modificar o eliminar las bonificaciones de un estudiante. El sistema ejecuta la acción indicada por la secretaria docente terminando así el caso de uso.	
Referencias	RF20, RF21, RF22
Flujo Normal de Eventos	
Sección “General”	
Acción del actor	Respuesta del sistema
1. La secretaria docente solicita gestionar una	2. El sistema muestra “Acreditar bonificación”, “Modificar

bonificación de un estudiante.	bonificación” y “Eliminar bonificación”.
3. La secretaria docente selecciona una de las opciones presentadas.	<p>4.1. Si selecciona “Acreditar bonificación” ver sección Acreditar Crédito Bonificación.</p> <p>4.2. Si selecciona “Modificar bonificación” ver sección Modificar Crédito Bonificación.</p> <p>4.3. Si selecciona “Eliminar bonificación” ver sección Eliminar Crédito Bonificación.</p>

Sección “Acreditar Crédito Bonificación”

Acción del actor	Respuesta del sistema
1. La secretaria docente selecciona la opción “Acreditar bonificación”.	2. El sistema muestra un formulario donde se pueda seleccionar la fecha, período y el valor de la bonificación.
3. La secretaria docente selecciona los datos.	4. El sistema verifica que los campos no estén vacíos.
	5. El sistema notifica al usuario que se asignó satisfactoriamente la bonificación.

Prototipo de Interfaz de usuario

Acreditar Bonificación



Nombre: Hector Luis Rodriguez


Fecha: Día * Mes * Año *

Periodo: *

Nota: *

Sección “Modificar Crédito Bonificación”

Acción del actor	Respuesta del sistema
1. La secretaria docente selecciona la opción “Modificar bonificación”.	2. El sistema muestra un formulario con todas las bonificaciones que ha obtenido el estudiante.
3. La secretaria docente selecciona una bonificación.	4. El sistema muestra un formulario donde sólo el

	campo nota está habilitado.
5. La secretaria docente modifica la nota.	6. El sistema verifica que el dato introducido sea correcto y guarda los cambios en la base de datos.
	7. El sistema notifica al usuario que se modificó satisfactoriamente la bonificación.
Flujo alternativo 1	
Acción del actor	Respuesta del sistema
5. Si la secretaria docente no modifica la nota de una bonificación, regresa al paso 2.	
Prototipo de Interfaz de usuario	
<div style="border: 1px solid gray; padding: 5px;"> <p style="background-color: #e0e0e0; margin: -5px -5px 5px -5px; padding: 2px;">Modificar Bonificaciones</p> <div style="border: 1px solid gray; padding: 10px; margin: 5px 5px 10px 5px;">  <div style="text-align: right; margin-top: 10px;"> Fecha: Día * 19 ▼ Mes * Febrero ▼ Año * 2009 ▼ </div> <div style="text-align: right; margin-top: 10px;"> Periodo: * Curso 2008_2009 ▼ </div> <div style="text-align: right; margin-top: 10px;"> Nota: * ▼ </div> <div style="display: flex; justify-content: space-around; margin-top: 10px;"> Aceptar Cancelar </div> <p>Nombre: Hector Luis Rodriguez</p> </div> </div>	
Sección “Eliminar Crédito Bonificación”	
Acción del actor	Respuesta del sistema
1. La secretaria docente selecciona la opción “Eliminar crédito”.	2. El sistema muestra un formulario con todas las bonificaciones que ha obtenido el estudiante.
3. La secretaria docente selecciona una o varias bonificaciones.	4. El sistema muestra un mensaje para confirmar si desea eliminar la(s) bonificación(es).
5. La secretaria docente confirma que desea eliminar la(s) bonificación(es).	6. El sistema elimina la(s) bonificación(es) de la base de datos.
	7. El sistema notifica al usuario que se eliminó satisfactoriamente la(s) bonificación(es).
Flujo alternativo 1	

Acción del actor	Respuesta del sistema										
5. Si la secretaria docente no selecciona eliminar la bonificación, se regresa al paso 2.											
Prototipo de Interfaz de usuario											
<div style="border: 1px solid black; padding: 10px;"> <p style="background-color: #e0e0e0; margin: -10px -10px 10px -10px; padding: 5px;">Eliminar Bonificaciones</p> <div style="display: flex; align-items: center;">  <table border="1" data-bbox="591 684 1448 800" style="border-collapse: collapse; text-align: center;"> <thead> <tr style="background-color: #e0e0e0;"> <th>Periodo</th> <th>Premio(s)</th> <th>Nota</th> <th>Fecha</th> <th>Seleccione</th> </tr> </thead> <tbody> <tr> <td>2008_2009</td> <td>Bonificación</td> <td>0.03</td> <td>19/02/09</td> <td style="text-align: right;"><input type="checkbox"/></td> </tr> </tbody> </table> </div> <div style="margin-top: 10px;"> <p>Nombre: Hector Luis Rodriguez</p> <div style="display: flex; justify-content: center; gap: 50px;"> Aceptar Cancelar </div> </div> </div>		Periodo	Premio(s)	Nota	Fecha	Seleccione	2008_2009	Bonificación	0.03	19/02/09	<input type="checkbox"/>
Periodo	Premio(s)	Nota	Fecha	Seleccione							
2008_2009	Bonificación	0.03	19/02/09	<input type="checkbox"/>							
Poscondiciones:	-										
Prioridad:	Crítico										

Tabla 4.7. Especificación del caso de uso del sistema “Asignar Profesor a Grupo Docente”

Caso de uso	
CUS-7	Asignar Profesor a Grupo Docente
Propósito	Permitir realizar la acción de asignar un profesor a un grupo docente.
Actores Secretaria Docente	
Resumen: El caso de uso se inicia cuando la secretaria docente accede a realizar la operación de asignar un profesor a un grupo docente dada una asignatura y un departamento. Si el grupo docente ya le fue asignado ese profesor, se muestra un mensaje de notificación y no se le asigna. En caso contrario, el sistema ejecuta la acción indicada terminando así el caso de uso.	
Referencias	RF23


Flujo Normal de Eventos	
Acción del actor	Respuesta del sistema
1. La secretaria docente selecciona la opción "Asignar profesor".	2. El sistema muestra un formulario para seleccionar los campos departamento, profesor, período y asignatura.
3. La secretaria docente selecciona los datos y pulsa la opción "Asignar".	4. El sistema verifica que el profesor no haya sido asignado a ese grupo docente.
	5. El sistema muestra un mensaje de notificación si el profesor fue asignado correctamente.
Flujo alternativo 1	
Acción del actor	Respuesta del sistema
	4. Si el profesor ya le fue asignado al grupo docente, el sistema muestra un mensaje y regresa al paso 2.
Prototipo de Interfaz de usuario	
<div style="border: 1px solid gray; padding: 10px; margin: 10px auto; width: 80%;"> <p style="text-align: center; margin: 0;">Asignar Profesor a Grupo Docente</p> <div style="border: 1px solid gray; padding: 10px; margin: 10px auto; width: 90%;"> <p>Grupo Docente * <input style="width: 80px;" type="text" value="01101"/></p> <p>Departamento * <input style="width: 150px;" type="text"/></p> <p>Profesor * <input style="width: 150px;" type="text"/></p> <p>Asignatura * <input style="width: 150px;" type="text"/></p> <p>Período * <input style="width: 150px;" type="text"/></p> <div style="text-align: right; margin-top: 10px;">  <input style="margin-right: 20px;" type="button" value="Asignar"/> <input type="button" value="Cancelar"/> </div> </div> </div>	
Poscondiciones:	-
Prioridad:	Crítico

Tabla 4.8. Especificación del caso de uso del sistema “Desubicar Profesor de Grupo Docente”

Caso de uso	
CUS-8	Desubicar Profesor de Grupo Docente
Propósito	Permitir realizar la acción de desubicar un profesor de un grupo docente.
Actores Secretaria Docente	
Resumen: El caso de uso se inicia cuando la secretaria docente accede a realizar la operación de eliminar a un profesor de un grupo docente. El sistema ejecuta la acción indicada por la secretaria docente terminando el caso de uso.	
Referencias	RF42
Flujo Normal de Eventos	
Acción del actor	Respuesta del sistema
1. La secretaria docente selecciona la opción “Desubicar profesor”.	2. El sistema muestra un formulario con los datos del profesor asignado al grupo docente.
3. La secretaria docente selecciona la opción de “Desubicar”.	4. El sistema muestra confirmación que el profesor será eliminado del grupo docente.
5. La secretaria docente confirma que desea eliminarlo.	6. El sistema elimina al profesor del grupo docente.
	7. El sistema muestra un mensaje de notificación si el profesor fue desubicado correctamente.
Flujo alternativo 1	
Acción del actor	Respuesta del sistema
5. Si la secretaria docente no selecciona realizar la eliminación, se regresa al paso 2.	
Prototipo de Interfaz de usuario	

Desubicar Profesor de Grupo Docente	
	
Poscondiciones:	-
Prioridad:	Crítico

Tabla 4.9. Especificación del caso de uso del sistema “Gestionar Crédito por Actividades”

Caso de uso	
CUS-9	Gestionar Crédito por Actividades
Propósito	Permitir realizar la acción de acreditar un crédito por una actividad, modificar el crédito o eliminar el crédito.
Actores	
Secretaria Docente	
El caso de uso se inicia cuando la secretaria docente accede a realizar una operación de acreditar, modificar o eliminar los créditos por actividades de un estudiante. El sistema ejecuta la acción indicada por la secretaria docente finalizando el caso de uso.	
Referencias	RF27, RF28, RF29
Flujo Normal de Eventos	
Sección “General”	
Acción del actor	Respuesta del sistema
1. La secretaria docente solicita gestionar el crédito por actividades de un estudiante.	2. El sistema muestra “Acreditar crédito”, “Modificar crédito” y “Eliminar crédito”.
3. La secretaria docente selecciona una de las opciones presentadas.	4.1. Si selecciona “Acreditar crédito” ver sección Acreditar Crédito por Actividades. 4.2. Si selecciona “Modificar crédito” ver sección Modificar Crédito por Actividades.

	4.3. Si selecciona "Eliminar crédito" ver sección Eliminar Crédito por Actividades.
Sección "Acreditar Crédito por Actividades"	
Acción del actor	Respuesta del sistema
1. La secretaria docente selecciona la opción "Acreditar crédito".	2. El sistema muestra un formulario para seleccionar los campos: fecha, período, tipo de actividad, la actividad y el crédito.
3. La secretaria docente selecciona los datos.	4. El sistema verifica que todos los datos estén seleccionados.
	5. El sistema notifica al usuario que se asignó satisfactoriamente el crédito.
Flujo alternativo 1	
Acción del actor	Respuesta del sistema
	4. Si los datos no están correctos el sistema muestra un mensaje a la secretaria docente y regresa al paso 2.
Prototipo de Interfaz de usuario	
<div style="border: 1px solid black; padding: 10px;"> <p style="text-align: center;">Acreditar Crédito por Actividades</p> <div style="border: 1px solid black; padding: 10px; margin: 10px 0;"> <div style="display: flex; align-items: center;">  <div> <p>Nombre: Hector Luis Rodriguez</p> </div> </div> <div style="margin-top: 20px;"> <p>Fecha: Día * <input type="text"/> Mes * <input type="text"/> Año * <input type="text"/></p> <p>Período: * <input type="text"/></p> <p>Tipo de Actividad: * <input type="text"/></p> <p>Actividades: * <input type="text"/></p> <p>Crédito * <input type="text"/></p> <div style="display: flex; justify-content: center; gap: 20px; margin-top: 10px;"> <input type="button" value="Aceptar"/> <input type="button" value="Cancelar"/> </div> </div> </div> </div>	

Sección "Modificar Crédito por Actividades"	
Acción del actor	Respuesta del sistema
1. La secretaria docente selecciona la opción "Modificar crédito".	2. El sistema muestra un formulario con todas las actividades en las cuales ha obtenido crédito el estudiante.
3. La secretaria docente selecciona una actividad.	4. El sistema muestra un formulario donde sólo el campo crédito está habilitado.
5. La secretaria docente modifica el valor del crédito.	6. Si los datos introducidos están correctos, el sistema guarda en la base de datos los cambios.
	7. El sistema notifica al usuario que se modificó satisfactoriamente el crédito.
Flujo alternativo 1	
Acción del actor	Respuesta del sistema
	6. Si los datos introducidos no están correctos el sistema muestra un mensaje a la secretaria docente y regresa al paso 4.
Prototipo de Interfaz de usuario	
<div style="border: 1px solid black; padding: 10px;"> <div style="background-color: #e0e0e0; padding: 5px; border: 1px solid black;"> Modificar Crédito por Actividades </div> <div style="border: 1px solid black; padding: 10px; margin-top: 10px;"> <div style="display: flex; align-items: center;">  <div> <p>Nombre: Hector Luis Rodriguez</p> </div> </div> <div style="margin-top: 20px;"> <p>Fecha: Día * <input type="text" value="7"/> Mes * <input type="text" value="Abril"/> Año * <input type="text" value="2007"/></p> <p>Periodo: * <input type="text" value="Curso 2007-2008"/></p> <p>Tipo de Actividad: * <input type="text" value="Proyecto Productivo"/></p> <p>Actividades: * <input type="text" value="PP5"/></p> <p>Crédito * <input type="text"/></p> </div> <div style="margin-top: 20px; display: flex; justify-content: center; gap: 50px;"> <input type="button" value="Aceptar"/> <input type="button" value="Cancelar"/> </div> </div> </div>	

Sección “Eliminar Crédito por Actividades”																
Acción del actor	Respuesta del sistema															
1. La secretaria docente selecciona la opción “Eliminar crédito”.	2. El sistema muestra un formulario con todas las actividades en las cuales el estudiante ha obtenido crédito.															
3. La secretaria docente selecciona una o varias actividades.	4. El sistema muestra un mensaje para confirmar si desea eliminar el(los) crédito(s).															
5. La secretaria docente confirma que desea eliminar el(los) crédito(s).	6. El sistema elimina el(los) crédito(s) de la base de datos.															
	7. El sistema notifica al usuario que se eliminó satisfactoriamente el(los) crédito(s).															
Flujo alternativo 1																
Acción del actor	Respuesta del sistema															
5. Si la secretaria docente no confirma que desea eliminar el crédito, se regresa al paso 2.																
Prototipo de Interfaz de usuario																
<div style="border: 1px solid black; padding: 10px;"> <p style="background-color: #e0e0e0; padding: 5px;">Eliminar Crédito por Actividades</p> <div style="border: 1px solid black; padding: 10px; margin-top: 10px;"> <div style="display: flex; align-items: center;">  <div> <p>Nombre: Hector Luis Rodriguez</p> </div> </div> <table border="1" style="width: 100%; border-collapse: collapse; margin-top: 10px;"> <thead> <tr style="background-color: #e0e0e0;"> <th>Periodo</th> <th>Actividad(s)</th> <th>Nota</th> <th>Fecha</th> <th>Selecione</th> </tr> </thead> <tbody> <tr> <td>2008_2009</td> <td>Exponer Trabajo</td> <td>0.03</td> <td>19/02/09</td> <td style="text-align: center;"><input checked="" type="checkbox"/></td> </tr> <tr> <td>2007_2008</td> <td>PP5</td> <td>0.01</td> <td>7/04/2007</td> <td style="text-align: center;"><input type="checkbox"/></td> </tr> </tbody> </table> <div style="display: flex; justify-content: space-around; margin-top: 10px;"> Aceptar Cancelar </div> </div> </div>		Periodo	Actividad(s)	Nota	Fecha	Selecione	2008_2009	Exponer Trabajo	0.03	19/02/09	<input checked="" type="checkbox"/>	2007_2008	PP5	0.01	7/04/2007	<input type="checkbox"/>
Periodo	Actividad(s)	Nota	Fecha	Selecione												
2008_2009	Exponer Trabajo	0.03	19/02/09	<input checked="" type="checkbox"/>												
2007_2008	PP5	0.01	7/04/2007	<input type="checkbox"/>												
Poscondiciones:	-															
Prioridad:	Crítico															

Tabla 4.10. Especificación del caso de uso del sistema “Consultar Grupo Docente”

Caso de uso	
CUS-10	Consultar Grupo Docente
Propósito	Permitir realizar la acción de listar los grupos docentes y ver sus matrículas.
Actores Usuario	
Resumen: El caso de uso se inicia cuando la secretaria docente, el profesor o el estudiante acceden a realizar la operación de buscar los grupos docentes dada una asignatura y ver su matrícula. El sistema muestra los grupos docentes finalizando el caso de uso.	
Referencias	RF6
Flujo Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El usuario selecciona la opción “Consultar grupo docente”.	2. El sistema muestra un formulario para seleccionar un plan de estudio y una asignatura.
3. El usuario selecciona un plan de estudio.	4. El sistema lista todas las asignaturas asociadas al plan de estudio.
5. El usuario selecciona una asignatura.	6. El sistema muestra una lista de los grupos docentes en los cuales se imparte la asignatura seleccionada.
7. El usuario selecciona un grupo docente.	8. El sistema muestra los datos del grupo docente seleccionado.
Flujo alternativo 1	
Acción del actor	Respuesta del sistema
	6. Si la lista de los grupos docentes está vacía, regresar al paso 2.
Prototipo de Interfaz de usuario	

Grupos Docentes por Asignatura	
Plan de Estudio: *	Asignatura: *
<input type="text"/>	<input type="text"/>
<input type="button" value="Aceptar"/>	<input type="button" value="Cancelar"/>
Grupo	Matricula Actual
01102	29
01103	31
01104	29
01105	25
Poscondiciones:	-
Prioridad:	Crítico

Tabla 4.11. Especificación del caso de uso del sistema “Mostrar Registro de Asistencia”

Caso de uso	
CUS-11	Mostrar Registro de Asistencia
Propósito	Permitir realizar la acción de mostrar el registro de asistencia de un grupo docente.
Actores	
Usuario	
Resumen:	
El caso de uso se inicia cuando la secretaria docente, el profesor o el estudiante acceden a realizar la operación de ver el registro de asistencia de un grupo docente. El sistema ejecuta la acción indicada terminando así el caso de uso.	
Referencias	RF13
Precondición:	Se deben mostrar los grupos docentes asociados al usuario autenticado.
Flujo Normal de Eventos	

Acción del actor	Respuesta del sistema
1. El usuario selecciona la opción "Ver Registro de Asistencia".	2. El sistema muestra un listado de los grupos docentes.
3. El usuario selecciona un grupo docente y la opción "Aceptar".	4. El sistema muestra un listado con los estudiantes, las fechas de las actividades realizadas, el por ciento de asistencia en los estados presente, ausente e injustificado.

Flujo alternativo 1

Acción del actor	Respuesta del sistema
	2. Si el usuario no tiene ningún grupo docente asociado el sistema muestra un mensaje de alerta.

Prototipo de Interfaz de usuario

Pregrado	Posgrado	Producción	Investigaciones																																													
Estructura Personal Carrera Control Docente Generador de Plantillas																																																
Control Docente\Registro de Asistencia																																																
Registro de Asistencia																																																
<p>Grupo Docente 01101 - Educación Física I - 72 Horas Clase</p> <table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <thead> <tr> <th style="width: 30%;">Nombre</th> <th style="width: 10%;">17/02/09</th> <th style="width: 15%;">Presente</th> <th style="width: 15%;">Justificado</th> <th style="width: 15%;">Injustificado</th> </tr> </thead> <tbody> <tr> <td>Hector Luis Rodriguez</td> <td>P</td> <td>1.38%</td> <td>0%</td> <td>0%</td> </tr> <tr> <td>Adrian P</td> <td>P</td> <td>1.38%</td> <td>0%</td> <td>0%</td> </tr> <tr> <td>Denis Albelo</td> <td>P</td> <td>1.38%</td> <td>0%</td> <td>0%</td> </tr> <tr> <td>Orlando Torres</td> <td>P</td> <td>1.38%</td> <td>0%</td> <td>0%</td> </tr> <tr> <td>Sergio Miranda</td> <td>P</td> <td>1.38%</td> <td>0%</td> <td>0%</td> </tr> <tr> <td>Arnaldo Perez</td> <td>P</td> <td>1.38%</td> <td>0%</td> <td>0%</td> </tr> <tr> <td>Imer Olivera</td> <td>P</td> <td>1.38%</td> <td>0%</td> <td>0%</td> </tr> <tr> <td>Damian Cervantes</td> <td>P</td> <td>1.38%</td> <td>0%</td> <td>0%</td> </tr> </tbody> </table>				Nombre	17/02/09	Presente	Justificado	Injustificado	Hector Luis Rodriguez	P	1.38%	0%	0%	Adrian P	P	1.38%	0%	0%	Denis Albelo	P	1.38%	0%	0%	Orlando Torres	P	1.38%	0%	0%	Sergio Miranda	P	1.38%	0%	0%	Arnaldo Perez	P	1.38%	0%	0%	Imer Olivera	P	1.38%	0%	0%	Damian Cervantes	P	1.38%	0%	0%
Nombre	17/02/09	Presente	Justificado	Injustificado																																												
Hector Luis Rodriguez	P	1.38%	0%	0%																																												
Adrian P	P	1.38%	0%	0%																																												
Denis Albelo	P	1.38%	0%	0%																																												
Orlando Torres	P	1.38%	0%	0%																																												
Sergio Miranda	P	1.38%	0%	0%																																												
Arnaldo Perez	P	1.38%	0%	0%																																												
Imer Olivera	P	1.38%	0%	0%																																												
Damian Cervantes	P	1.38%	0%	0%																																												
Poscondiciones:	-																																															
Prioridad:	Crítico																																															

Tabla 4.12. Especificación del caso de uso del sistema “Mostrar Registro de Evaluación”

Caso de uso	
CUS-12	Mostrar Registro de Evaluación
Propósito	Permitir realizar la acción de mostrar el registro de evaluación de un grupo docente.
Actores Usuario	
Resumen: El caso de uso se inicia cuando la secretaria docente, el profesor o el estudiante acceden a realizar la operación de ver el registro de evaluación de un grupo docente. El sistema muestra el registro de evaluaciones y finaliza el caso de uso.	
Referencias:	RF16
Precondiciones:	Se deben mostrar los grupos docentes asociados al usuario autenticado.
Flujo Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El usuario selecciona la opción “Ver Registro de Evaluación”.	2. El sistema muestra un listado de los grupos docentes.
3. El usuario selecciona un grupo docente y la opción “Aceptar”.	4. El sistema muestra un listado con los estudiantes del grupo docente seleccionado con sus respectivas evaluaciones.
Flujo alternativo 1	
Acción del actor	Respuesta del sistema
	2. Si el usuario no tiene ningún grupo docente asociado el sistema muestra un mensaje de alerta.
Prototipo de Interfaz de usuario	

Registrar Evaluación																																					
Grupo Docente 01101 - Educación Física I																																					
<table border="1"> <thead> <tr> <th>Nombre</th> <th>1er Corte</th> <th>2do Corte</th> <th>3er Corte</th> </tr> </thead> <tbody> <tr> <td>Hector Luis Rodriguez</td> <td>B</td> <td>B</td> <td>B</td> </tr> <tr> <td>Adrian P</td> <td>B</td> <td>B</td> <td>B</td> </tr> <tr> <td>Denis Albelo</td> <td>B</td> <td>B</td> <td>B</td> </tr> <tr> <td>Orlando Torres</td> <td>B</td> <td>B</td> <td>B</td> </tr> <tr> <td>Sergio Miranda</td> <td>B</td> <td>B</td> <td>B</td> </tr> <tr> <td>Arnaldo Perez</td> <td>B</td> <td>B</td> <td>B</td> </tr> <tr> <td>Imer Olivera</td> <td>B</td> <td>B</td> <td>B</td> </tr> <tr> <td>Damian Cervantes</td> <td>B</td> <td>B</td> <td>B</td> </tr> </tbody> </table>	Nombre	1er Corte	2do Corte	3er Corte	Hector Luis Rodriguez	B	B	B	Adrian P	B	B	B	Denis Albelo	B	B	B	Orlando Torres	B	B	B	Sergio Miranda	B	B	B	Arnaldo Perez	B	B	B	Imer Olivera	B	B	B	Damian Cervantes	B	B	B	
Nombre	1er Corte	2do Corte	3er Corte																																		
Hector Luis Rodriguez	B	B	B																																		
Adrian P	B	B	B																																		
Denis Albelo	B	B	B																																		
Orlando Torres	B	B	B																																		
Sergio Miranda	B	B	B																																		
Arnaldo Perez	B	B	B																																		
Imer Olivera	B	B	B																																		
Damian Cervantes	B	B	B																																		
Poscondiciones:	-																																				
Prioridad:	Crítico																																				

Tabla 4.13. Especificación del caso de uso del sistema “Generar Acta de Examen Final”

Caso de uso	
CUS-13	Generar Acta de Examen Final
Propósito	Permitir realizar la acción de generar el Acta de Examen Final de una asignatura.
Actores	
Trabajador	
Resumen:	
El caso de uso se inicia cuando la secretaria docente o el profesor acceden a realizar la operación de crear el Acta de Examen Final de una asignatura. El sistema genera el Acta de Examen Final con los integrantes del grupo docente finalizando el caso de uso.	
Referencias	RF32
Flujo Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El trabajador selecciona un grupo docente.	2. El sistema muestra el listado de los estudiantes asociados al grupo seleccionado.

3. El trabajador selecciona la opción “Generar Acta de Examen Final”.	4. El sistema muestra el Acta de Examen Final con los estudiantes del grupo docente seleccionado.
---	---

Prototipo de Interfaz de usuario

UCI		ACTA DE EXAMEN E INFORME DE EVALUACION FINAL								
Facultad: Facultad 1		Curso Académico: 08-09								
Grupo: 01502		Tipo de Curso: CRD								
Año: Quinto año		Carrera: Ingeniería Informática								
Asignatura: PD		Semestre: 2do Semestre								
No	Nombre y Apellidos	Convocatoria Ordinaria del Semestre			Convocatoria Extraordinaria Final del Semestre			Convoc. Especial Final del Curso		
		Firma	Evaluación		Firma	Evaluación	Fecha	Firma	Eva-lua-ción	Fecha
			Examen	Final						
1	Yurisleis Alonso Beatón									
2	Yisel Avila Portales									
3	Enmanuel Acaharez Reyes									
4	Eduardo Basulto Uribe									
5	José Enrique Benitez Jimenez									
6	Josue Hugo Blanco Brito									
7	José Antonio Bustio Encinosa									
8	Ivan Carmenates García									
9	Aralys Ferriol Ortiz									
10	Yenier González Pérez									
11	Aniel Agustín Gutiérrez González									
ddd		Fecha	Firma Prof.		Fecha	Firma Prof.		Fecha	Firma Prof.	
Poscondiciones:		-								
Prioridad:		Crítico								

Tabla 4.14. Especificación del caso de uso del sistema “Solicitar Asignatura Adelanto”

Caso de uso	
CUS-14	Solicitar Asignatura Adelanto
Propósito	Permitir realizar la acción de solicitar asignatura de adelanto.

Actores	
Estudiante	
Resumen:	
El caso de uso se inicia cuando el estudiante accede a realizar la operación de solicitar adelantar una o varias asignaturas básicas. El sistema muestra la lista de las asignaturas que de acuerdo a la precedencia y el curso académico el estudiante puede adelantar. El estudiante selecciona una o varias asignaturas terminando así el caso de uso.	
Referencias	RF7
Flujo Normal de Eventos	
Acción del actor	Respuesta del sistema
1. El estudiante selecciona la opción "Solicitar asignatura".	2. El sistema muestra un formulario con el listado de las asignaturas separadas por año que el estudiante pueda adelantar.
3. El estudiante selecciona las asignaturas que desea matricular y selecciona la opción "Aceptar".	4. El sistema almacena por cada asignatura seleccionada, el nombre del estudiante, el nombre de la asignatura y la fecha de la solicitud.
	5. El sistema muestra la lista de las asignaturas solicitadas.
Flujo alternativo 1	
Acción del actor	Respuesta del sistema
3. Si el estudiante no selecciona ninguna asignatura regresa al paso 2.	
Prototipo de Interfaz de usuario	

Solicitar Asignatura de Adelanto		
Asignaturas por Año	Crédito	Seleccionar
Primer Año		
- Educación Física 1	0.01	<input checked="" type="checkbox"/>
- Matemática 1	0.01	<input type="checkbox"/>
<hr/>		
Segundo Año		
<hr/>		
Tercer Año		
<hr/>		
Cuarto Año		
<hr/>		
Quinto Año		
<hr/>		
<input type="button" value="Aceptar"/>		<input type="button" value="Cancelar"/>

Poscondiciones:	-
Prioridad:	Crítico

Tabla 4.15. Especificación del caso de uso del sistema “Aprobar_Rechazar Solicitud de Asignatura”

Caso de uso	
CUS-15	Aprobar_Rechazar Solicitud de Asignatura
Propósito	Permitir realizar las acciones de aprobar las solicitudes para asignaturas de adelanto o rechazarlas.
Actores	
Secretaria Docente	

Resumen:	
El caso de uso se inicia cuando la secretaria docente accede a realizar la operación de aprobar o rechazar la(s) solicitud(es) de las asignaturas de adelanto. El sistema muestra el listado de las solicitudes y la secretaria docente ejecuta la acción indicada finalizando el caso de uso.	
Referencias	RF8, RF9
Flujo Normal de Eventos	
Sección “General”	
Acción del actor	Respuesta del sistema
1. La secretaria docente selecciona la opción “Asignaturas de adelanto”.	2. El sistema muestra un listado con las solicitudes de asignaturas de adelanto y la opción de “Aprobar” o “Rechazar”.
3. La secretaria docente selecciona una de las opciones presentadas.	4.1. Si selecciona “Aprobar” ver sección Aprobar Solicitud de Adelanto de Asignatura. 4.2. Si selecciona “Rechazar” ver sección Rechazar Solicitud de Adelanto de Asignatura.
Sección “Aprobar Solicitud de Adelanto de Asignatura”	
Acción del actor	Respuesta del sistema
1. La secretaria selecciona la opción “Aprobar”.	2. El sistema muestra un formulario con el listado de los estudiantes que hicieron solicitudes y las asignaturas solicitadas.
3. La secretaria docente selecciona una asignatura.	4. El sistema muestra un formulario con la lista de los grupos docentes relacionados con esa asignatura y sus matrículas.
5. La secretaria docente selecciona un grupo docente y pulsa la opción “Aceptar”.	6. El sistema matricula al estudiante en el grupo docente seleccionado.
	7. El sistema elimina las solicitudes que fueron valoradas.
	8. El sistema notifica que la operación se realizó satisfactoriamente.
Flujo alternativo 1	
Acción del actor	Respuesta del sistema
	2. Si la lista de solicitudes está vacía el sistema muestra un mensaje de alerta.

5. Si la secretaria no selecciona ningún grupo y pulsa la opción “Cancelar”, regresa al paso 2.																
Sección “Rechazar Solicitud de Adelanto de Asignatura”																
Acción del actor	Respuesta del sistema															
1. La secretaria selecciona la opción “Rechazar”.	2. El sistema muestra un formulario con la lista de las asignaturas de adelanto solicitadas y de los estudiantes que las solicitaron.															
3. La secretaria docente selecciona una o varias solicitudes y pulsa la opción “Aceptar”.	4. El sistema pide confirmación de la acción.															
	5. El sistema elimina las solicitudes rechazadas.															
	6. El sistema notifica que la operación se realizó satisfactoriamente.															
Prototipo de Interfaz de usuario																
<div style="border: 1px solid black; padding: 10px;"> <p>Solicitud de Asignaturas de Adelanto</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 15%;">Foto</th> <th style="width: 25%;">Nombre</th> <th style="width: 30%;">Asignatura Solicitada</th> <th style="width: 15%;">Aprobar</th> <th style="width: 15%;">Rechazar</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;"></td> <td>Hector Luis Rodriguez</td> <td>Matemática</td> <td style="text-align: center;"><input checked="" type="checkbox"/></td> <td style="text-align: center;"><input type="checkbox"/></td> </tr> <tr> <td style="text-align: center;"></td> <td>Denis Albelo</td> <td>Inglés V</td> <td style="text-align: center;"><input type="checkbox"/></td> <td style="text-align: center;"><input checked="" type="checkbox"/></td> </tr> </tbody> </table> <div style="display: flex; justify-content: center; gap: 20px; margin-top: 20px;"> Aceptar Cancelar </div> </div>		Foto	Nombre	Asignatura Solicitada	Aprobar	Rechazar		Hector Luis Rodriguez	Matemática	<input checked="" type="checkbox"/>	<input type="checkbox"/>		Denis Albelo	Inglés V	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Foto	Nombre	Asignatura Solicitada	Aprobar	Rechazar												
	Hector Luis Rodriguez	Matemática	<input checked="" type="checkbox"/>	<input type="checkbox"/>												
	Denis Albelo	Inglés V	<input type="checkbox"/>	<input checked="" type="checkbox"/>												
Poscondiciones:	-															
Prioridad:	Crítico															

Anexo 5: Diagramas de colaboración

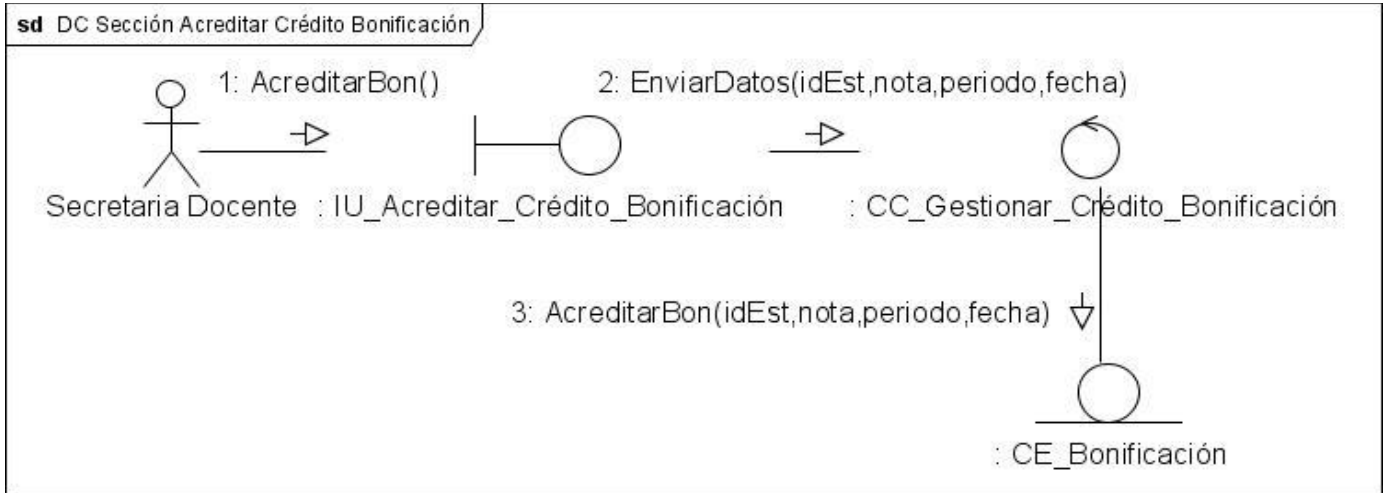


Figura 53. Diagrama de colaboración sección “Acreditar Crédito Bonificación”

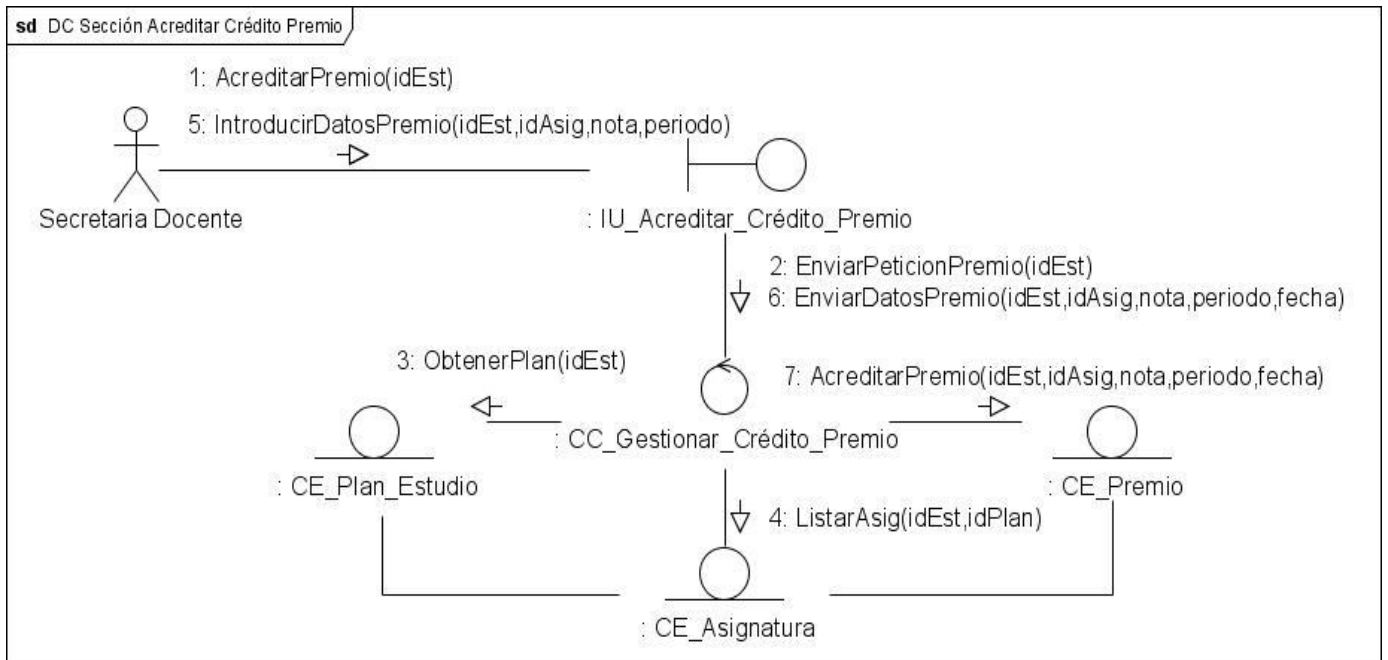


Figura 54. Diagrama de colaboración sección “Acreditar Crédito Premio”

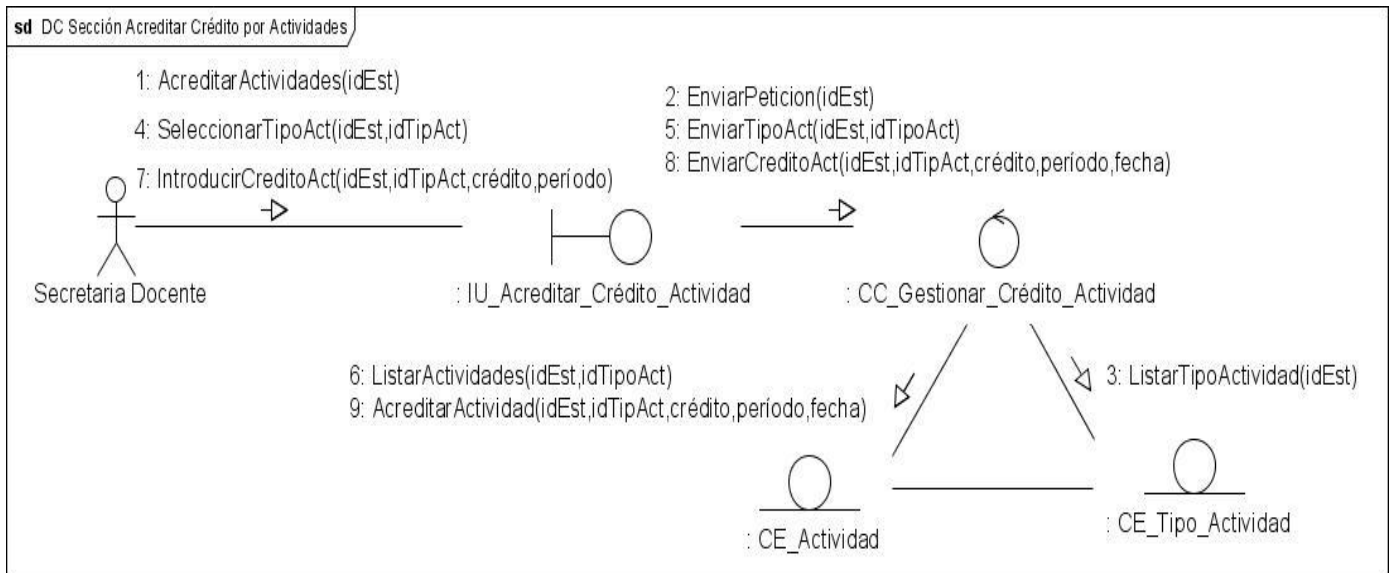


Figura 55. Diagrama de colaboración sección “Acreditar Crédito Actividades”

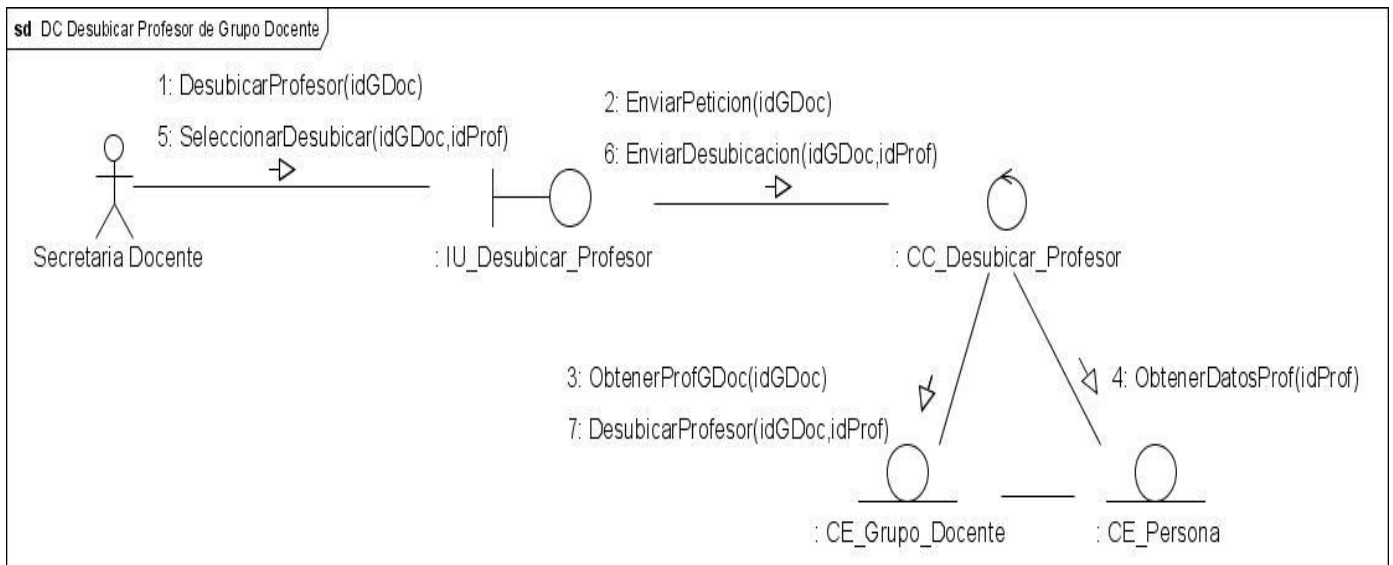


Figura 56. Diagrama de colaboración “Desubicar Profesor de Grupo Docente”

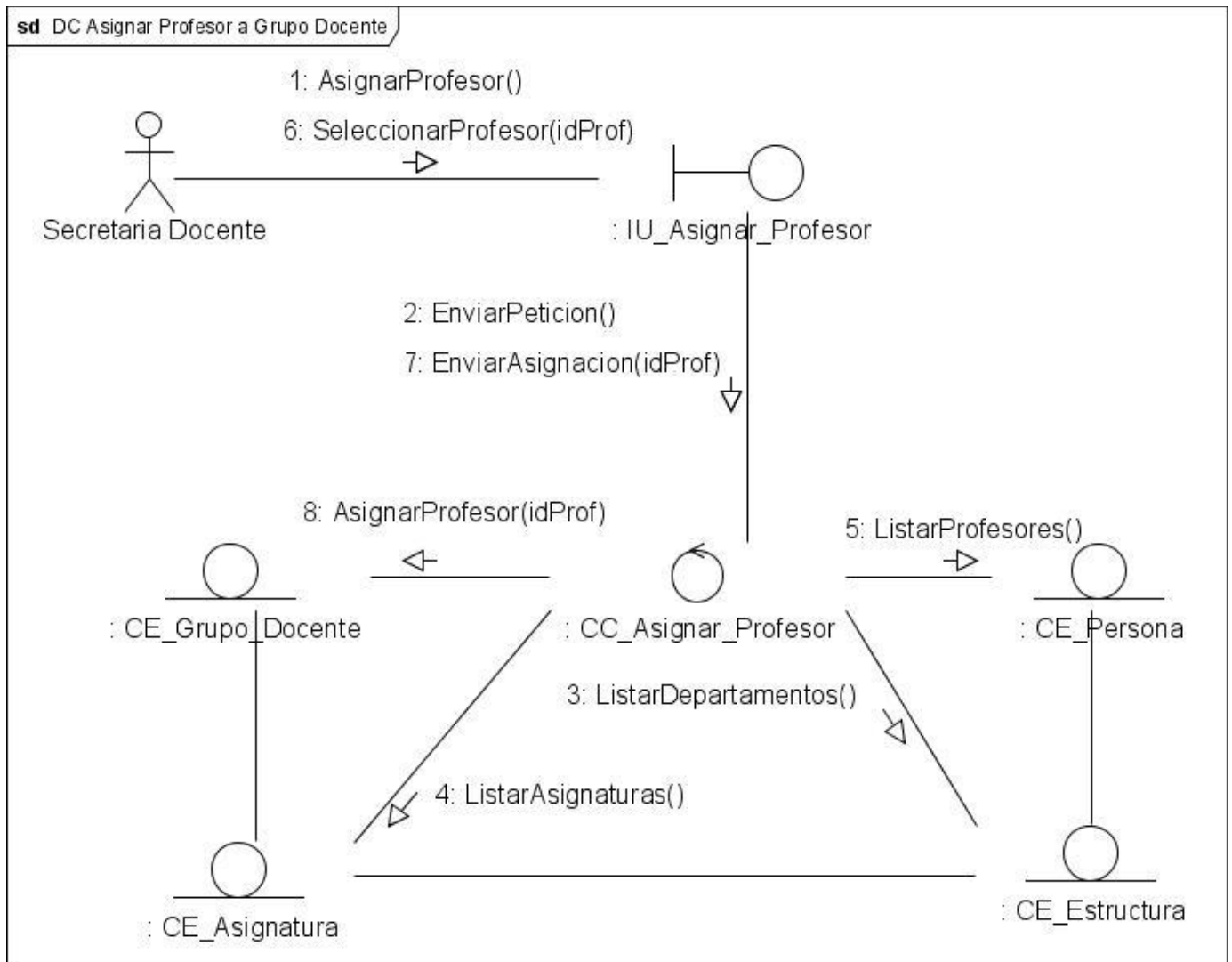


Figura 57. Diagrama de colaboración “Asignar Profesor a Grupo Docente”

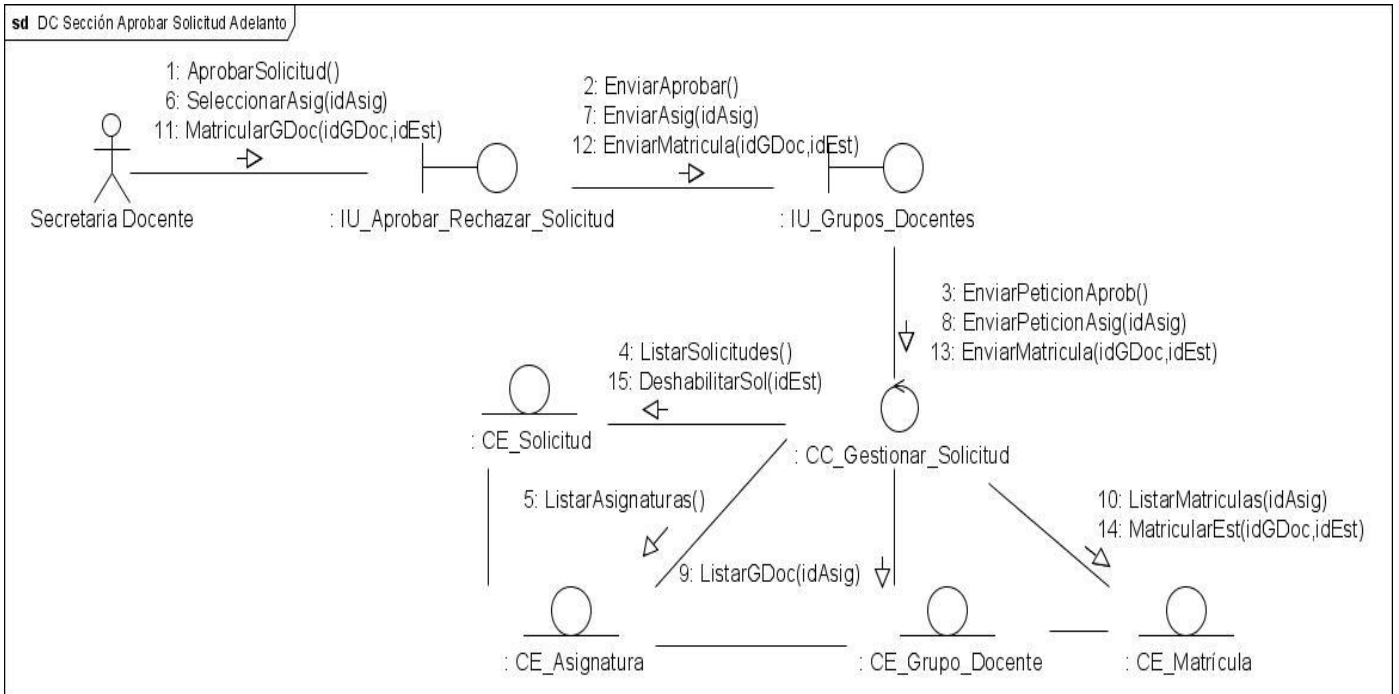


Figura 58. Diagrama de colaboración sección “Aprobar Solicitud de Adelanto de Asignatura”

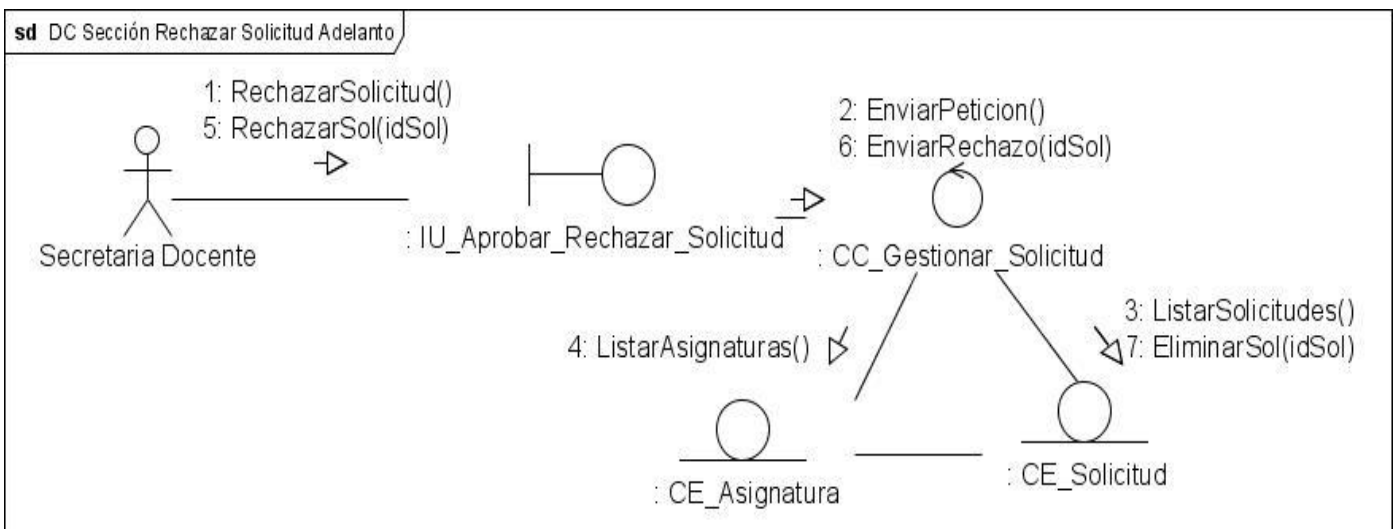


Figura 59. Diagrama de colaboración sección “Rechazar Solicitud de Adelanto de Asignatura”

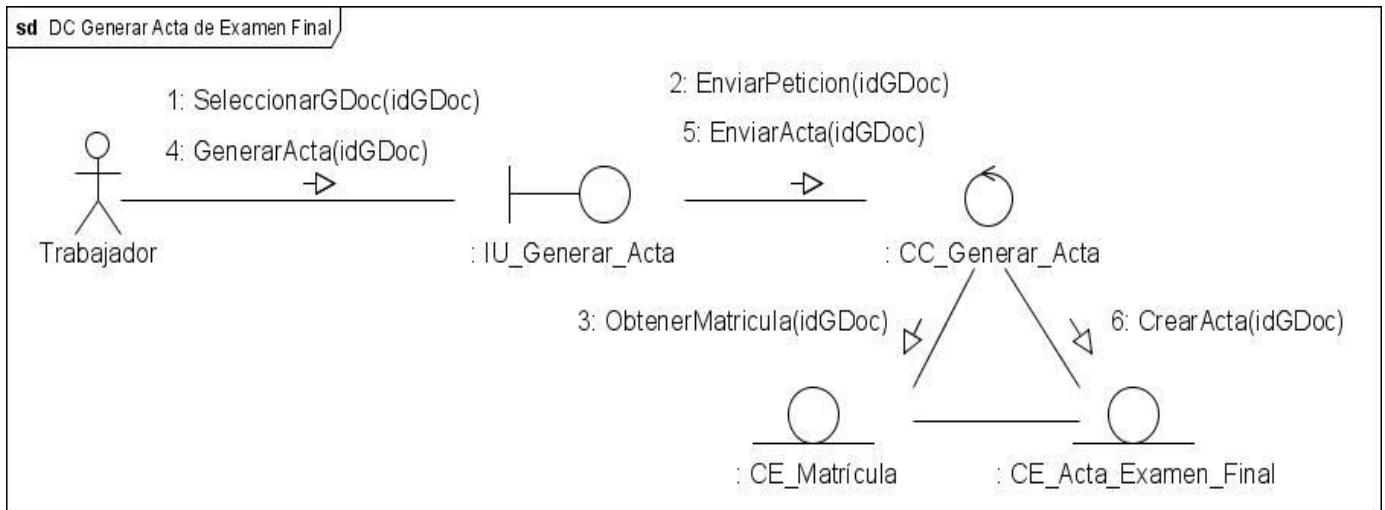


Figura 60. Diagrama de colaboración “Generar Acta de Examen Final”

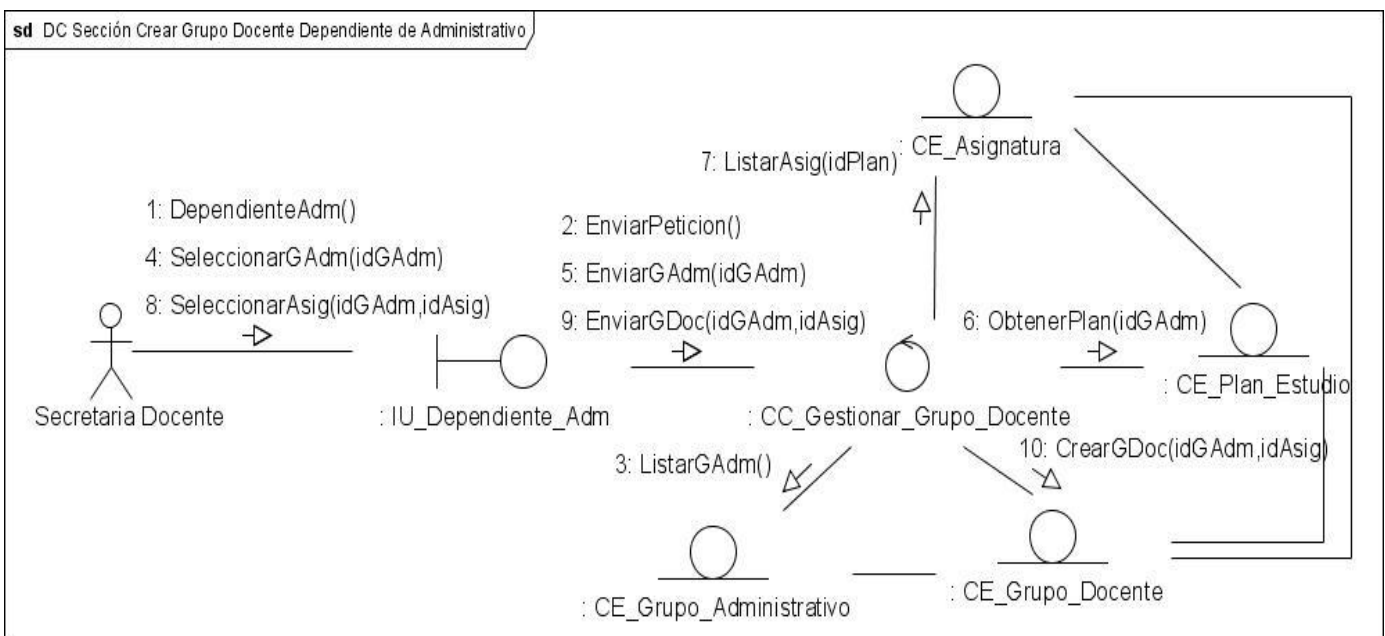


Figura 61. Diagrama de colaboración sección “Dependiente de administrativo”

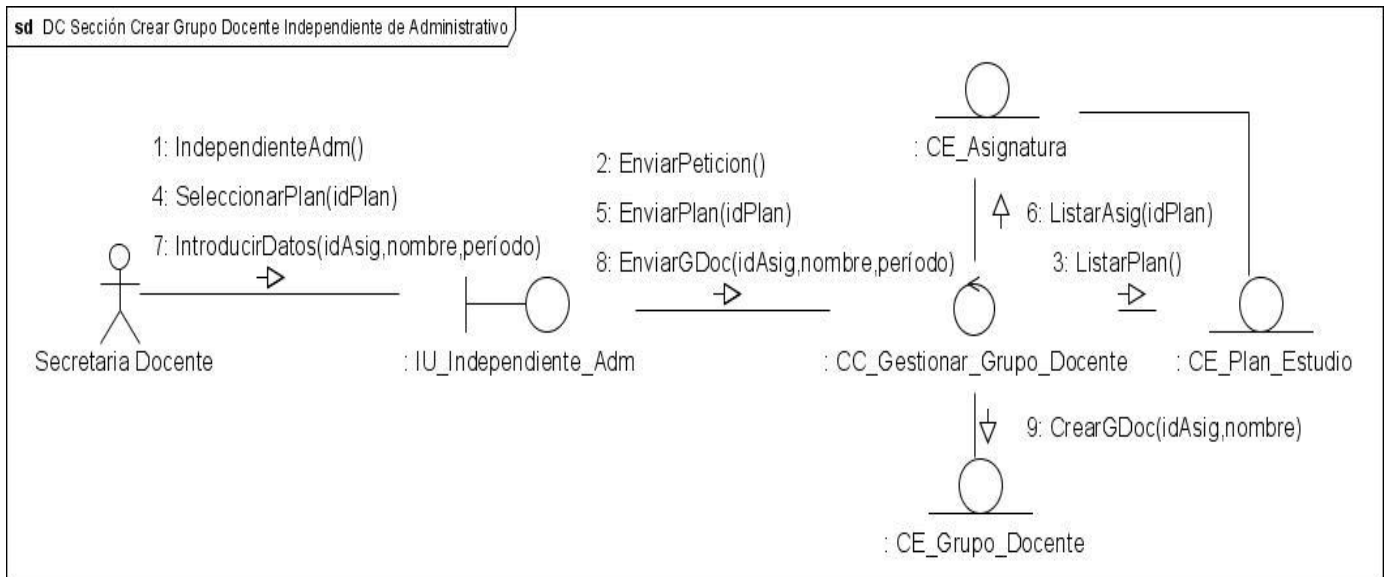


Figura 62. Diagrama de colaboración sección “Independiente de administrativo”

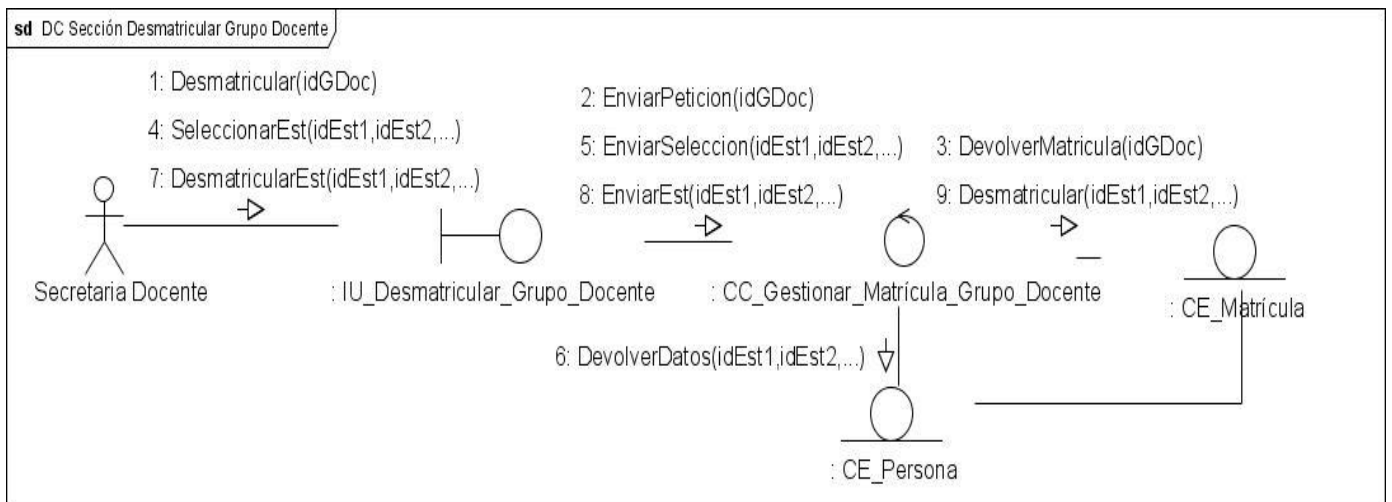


Figura 63. Diagrama de colaboración sección “Desmatricular Grupo Docente”

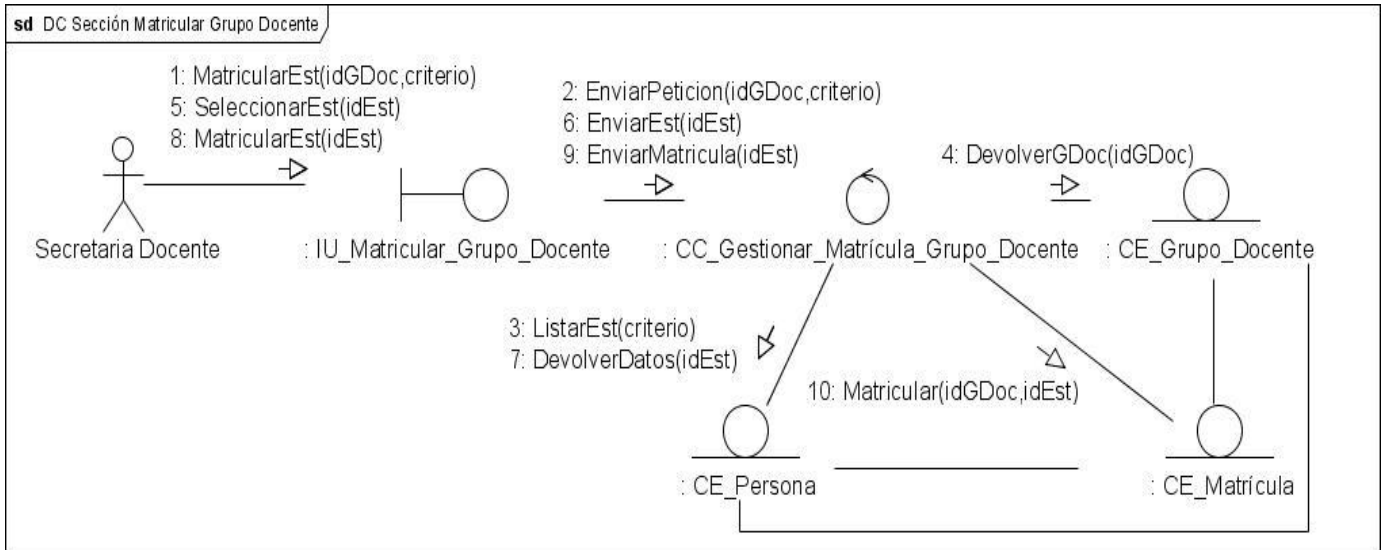


Figura 64. Diagrama de colaboración sección “Matricular Grupo Docente”

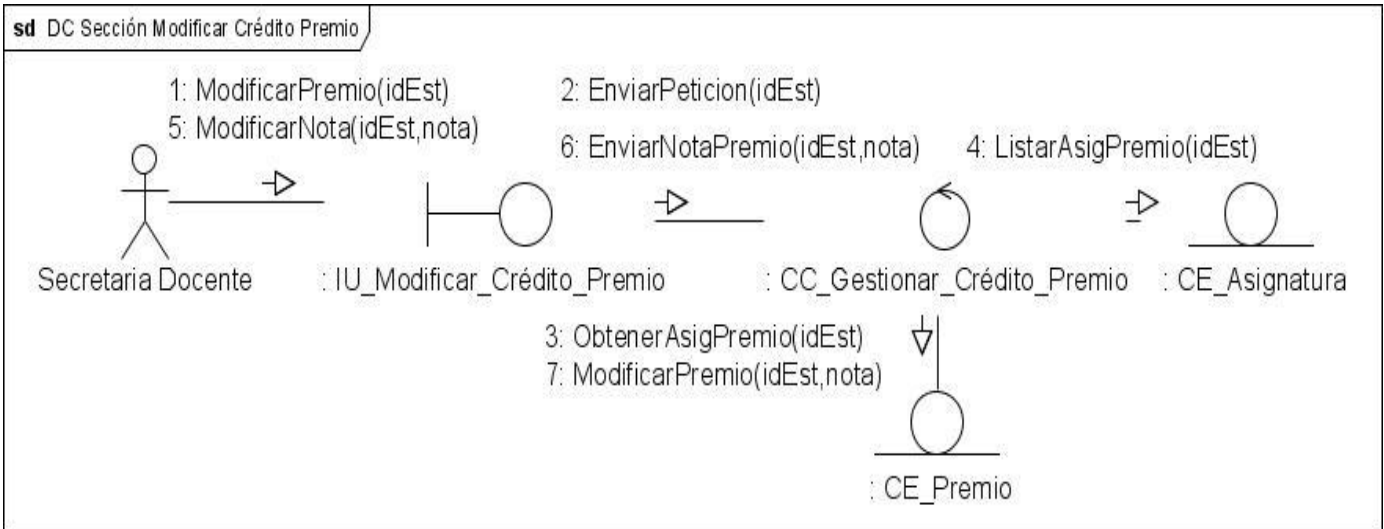


Figura 65. Diagrama de colaboración sección “Modificar Crédito Premio”

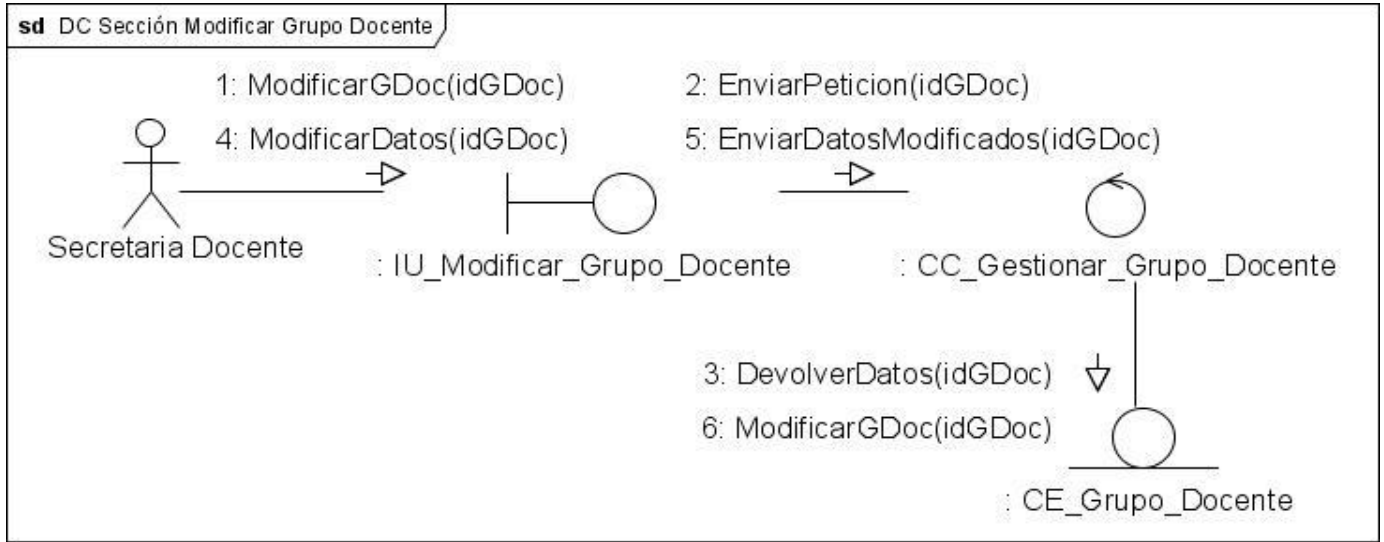


Figura 66. Diagrama de colaboración sección “Modificar Grupo Docente”

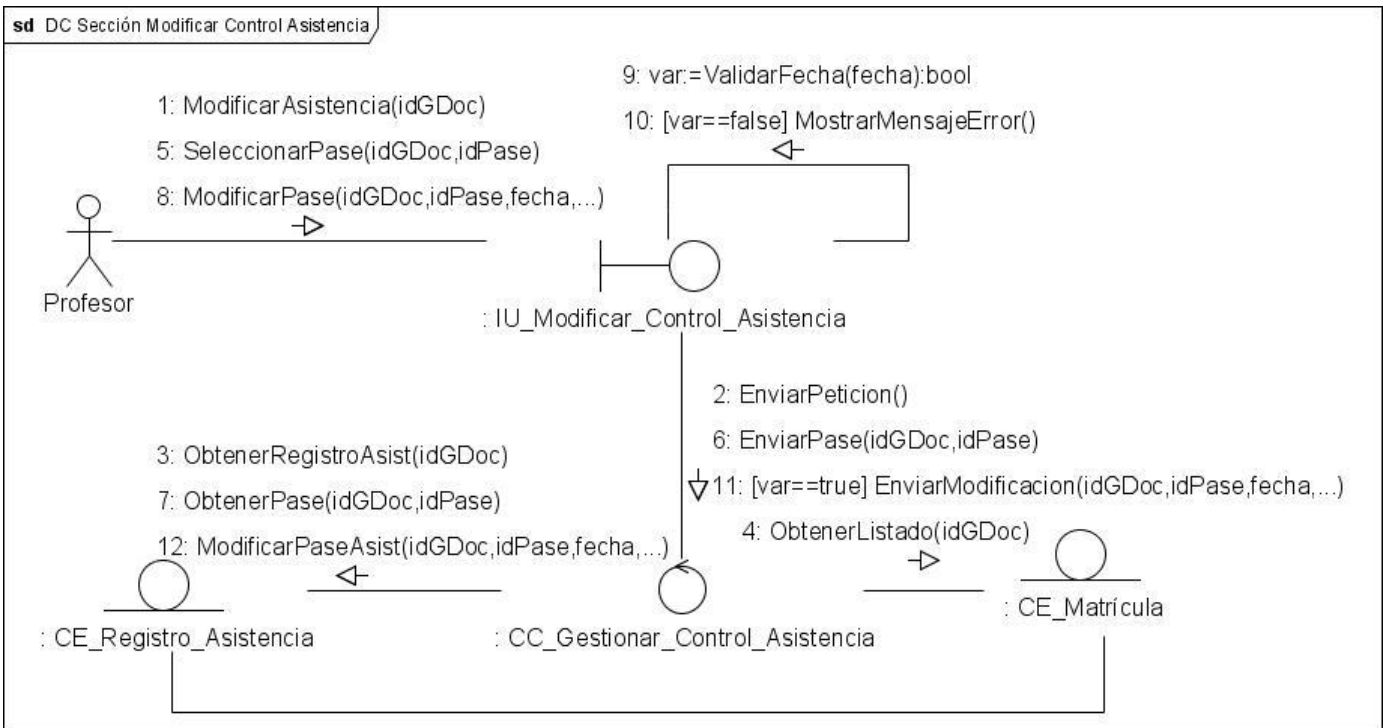


Figura 67. Diagrama de colaboración sección “Modificar Control Asistencia”

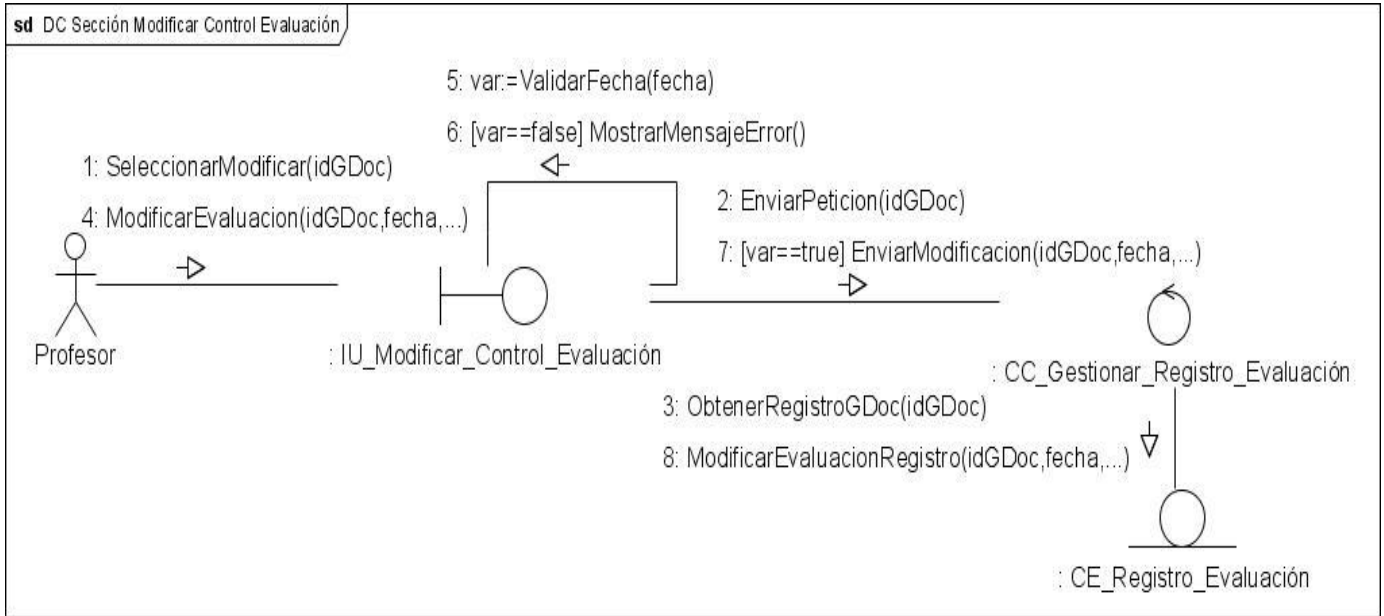


Figura 68. Diagrama de colaboración sección “Modificar Control Evaluación”

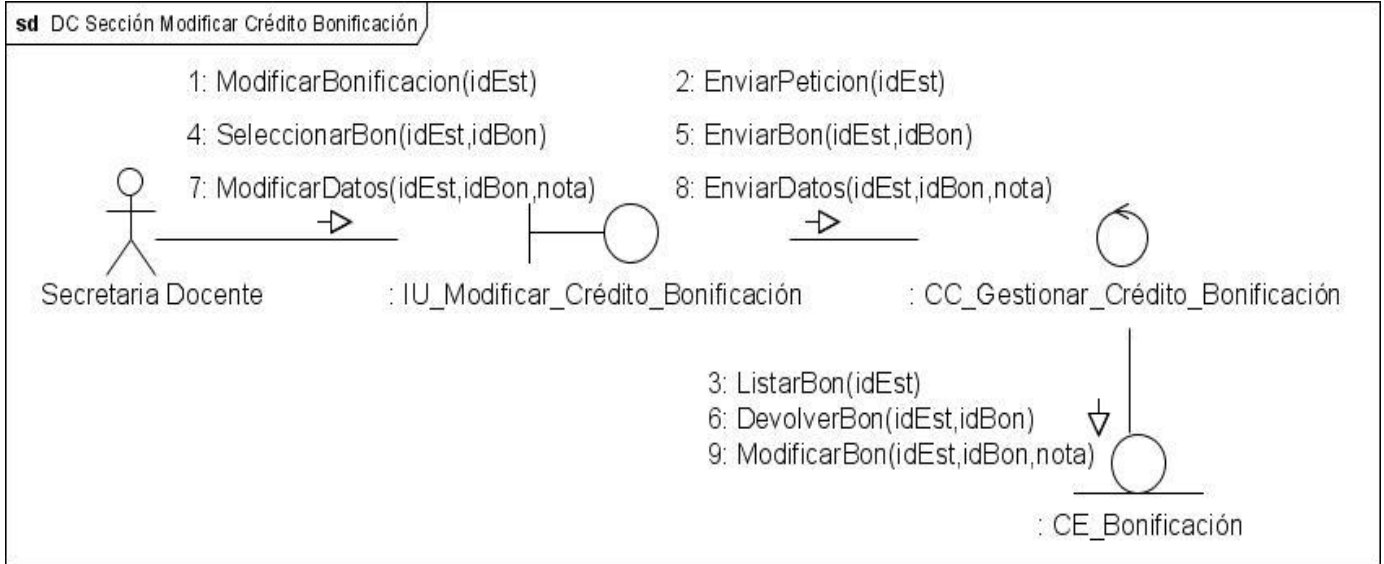


Figura 69. Diagrama de colaboración sección “Modificar Crédito Bonificación”

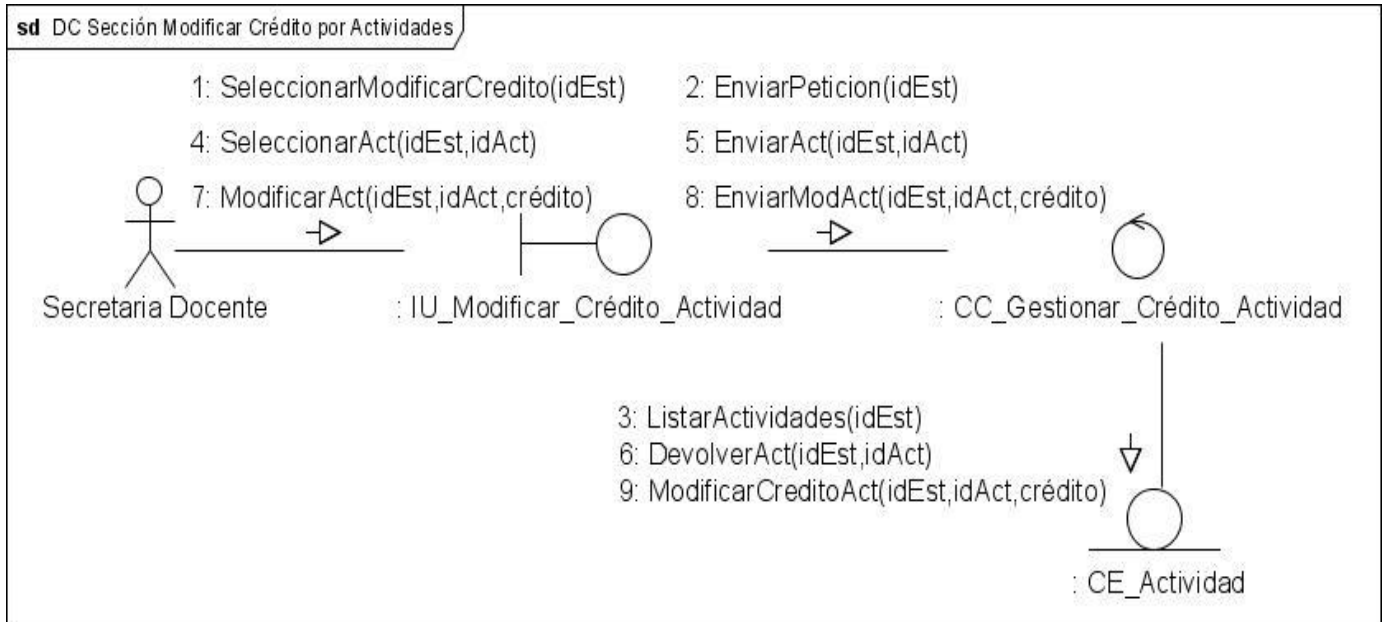


Figura 70. Diagrama de colaboración sección “Modificar Crédito por Actividades”

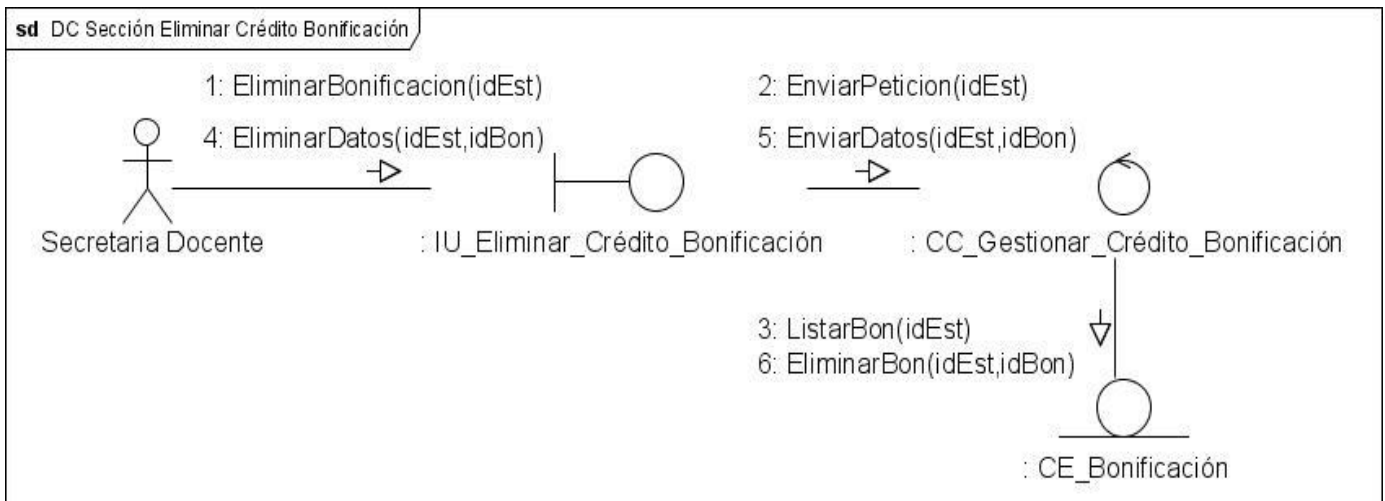


Figura 71. Diagrama de colaboración sección “Eliminar Crédito Bonificación”

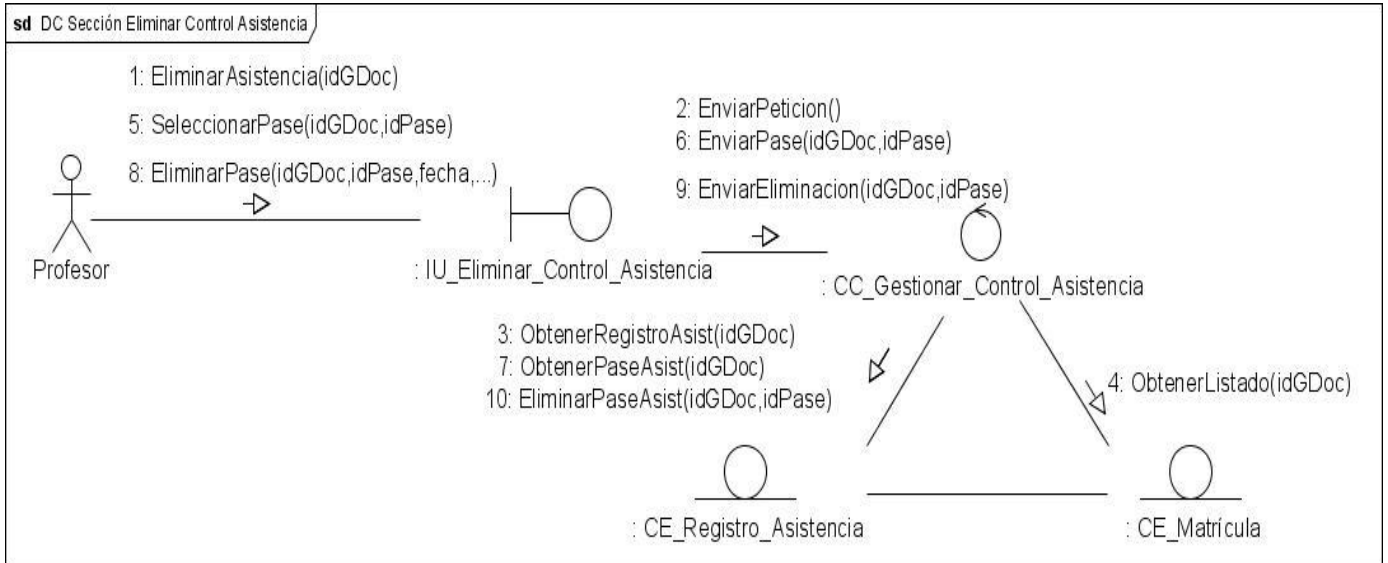


Figura 72. Diagrama de colaboración sección “Eliminar Control Asistencia”

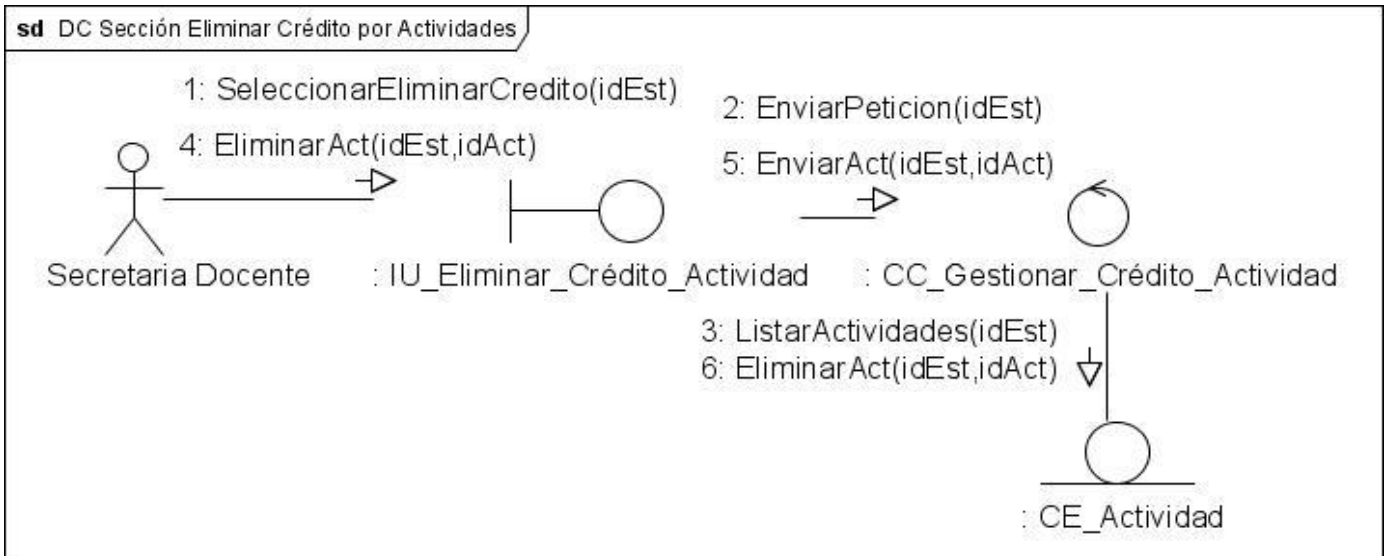


Figura 73. Diagrama de colaboración sección “Eliminar Crédito por Actividades”

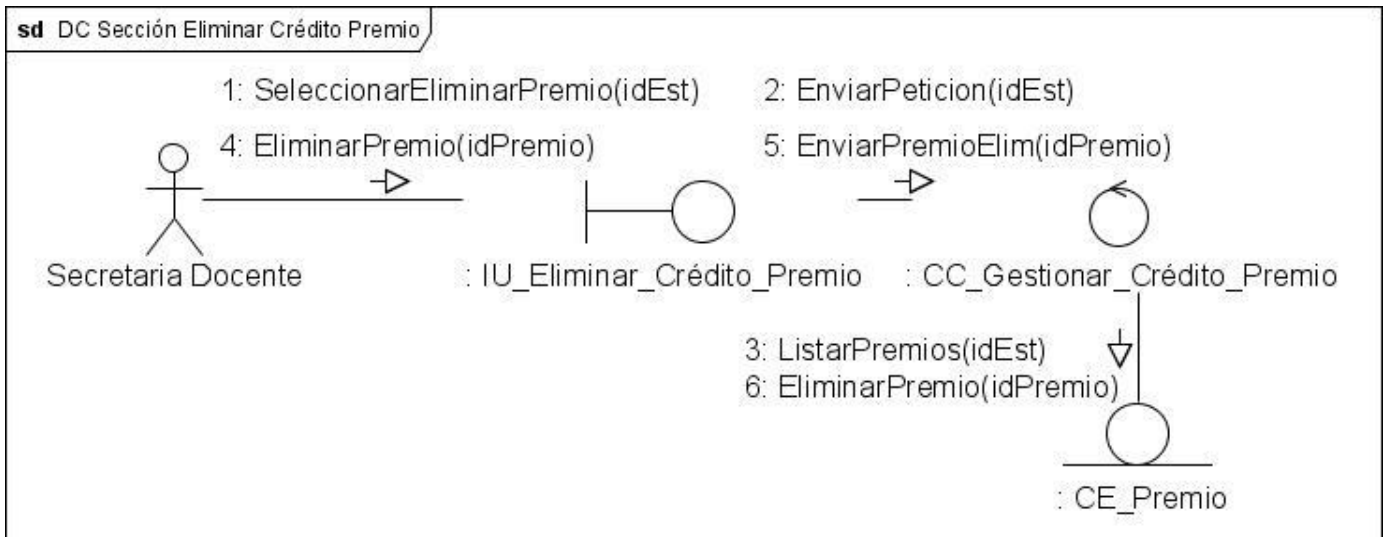


Figura 74. Diagrama de colaboración sección “Eliminar Crédito Premio”

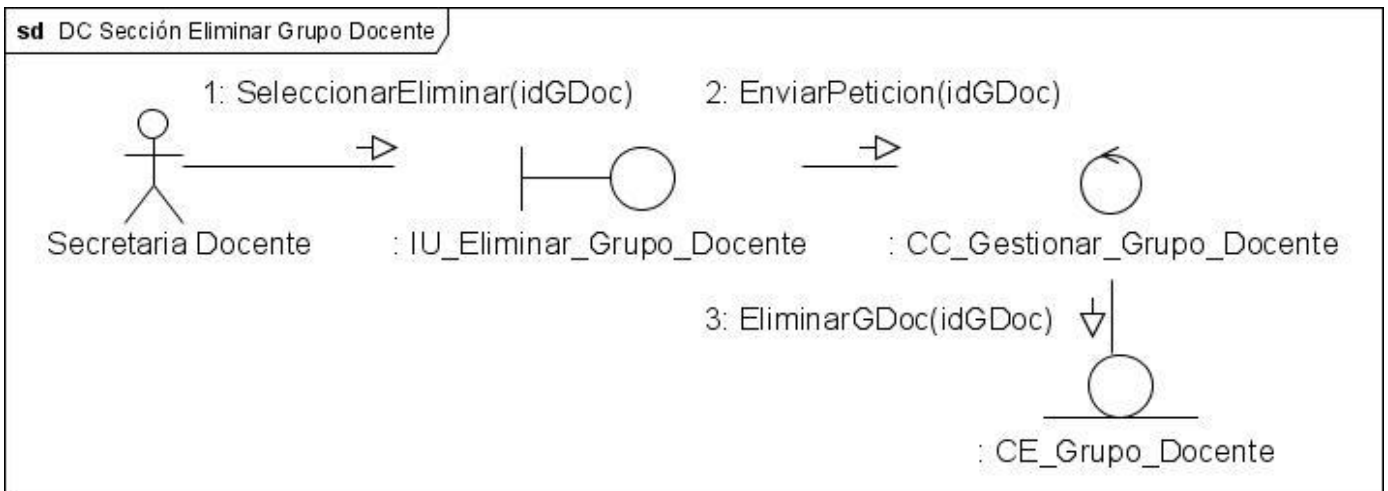


Figura 75. Diagrama de colaboración sección “Eliminar Grupo Docente”

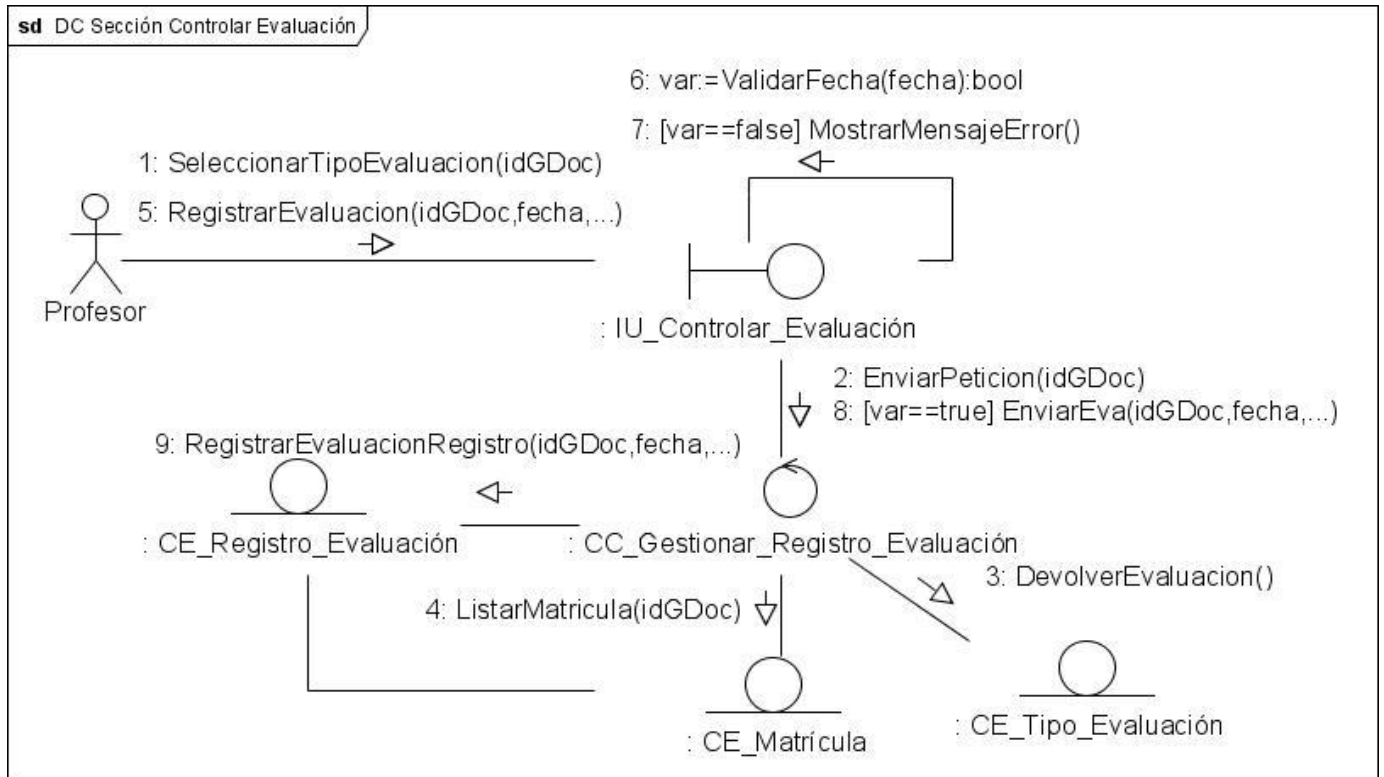


Figura 76. Diagrama de colaboración sección “Controlar Evaluación”

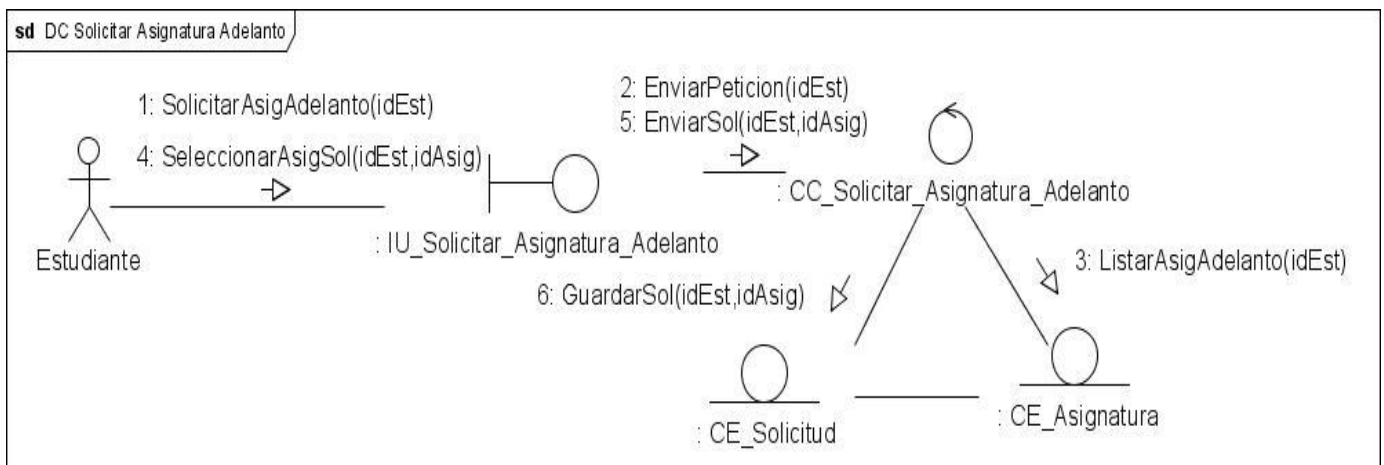


Figura 77. Diagrama de colaboración “Solicitar Asignatura Adelanto”

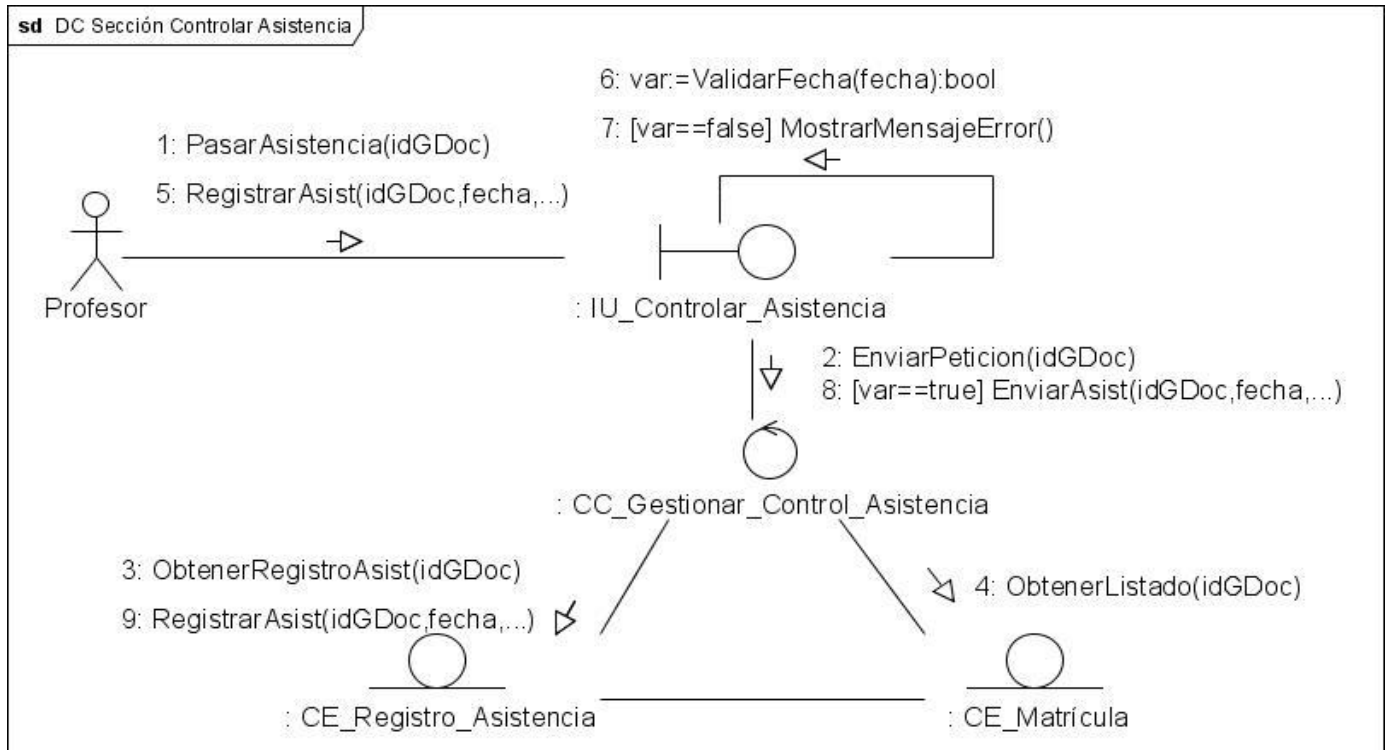


Figura 78. Diagrama de colaboración sección “Controlar Asistencia”

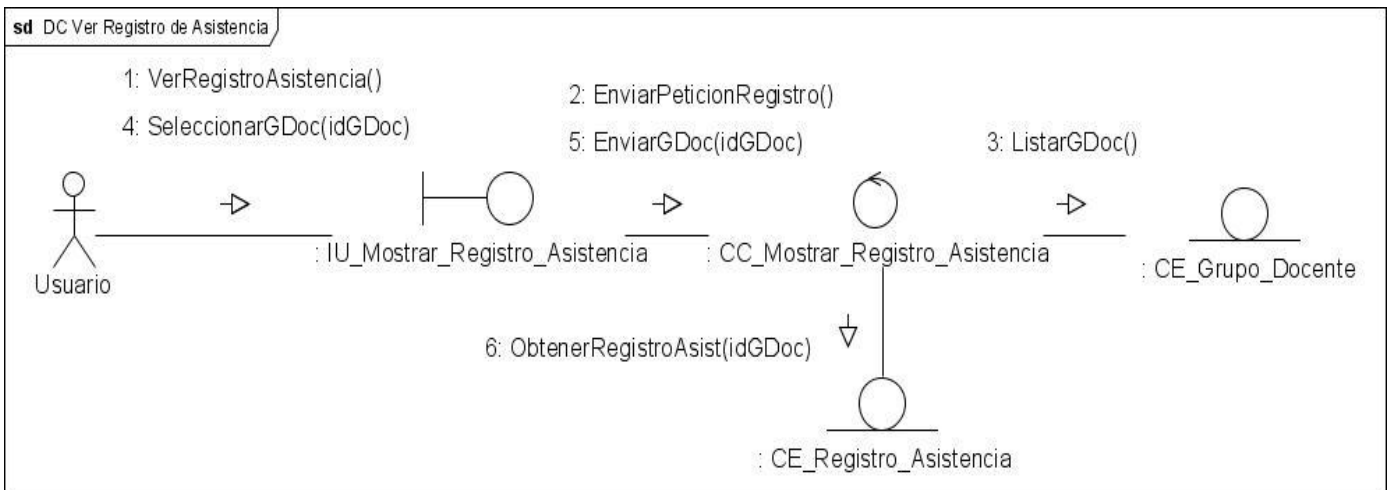


Figura 79. Diagrama de colaboración “Mostrar Registro de Asistencia”

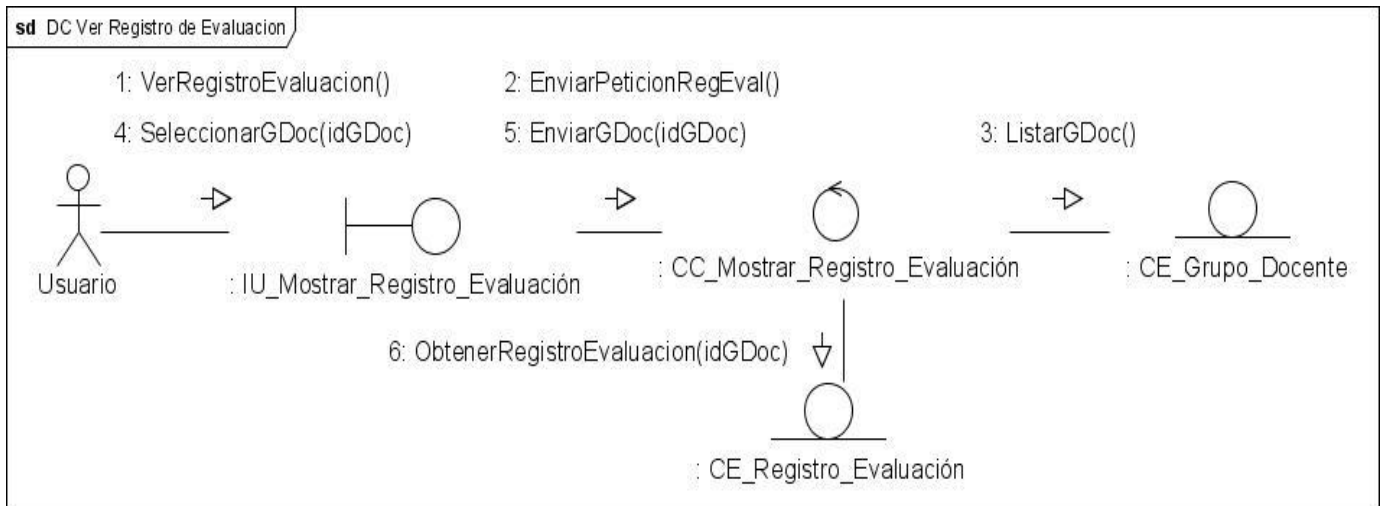


Figura 80. Diagrama de colaboración “Mostrar Registro de Evaluación”

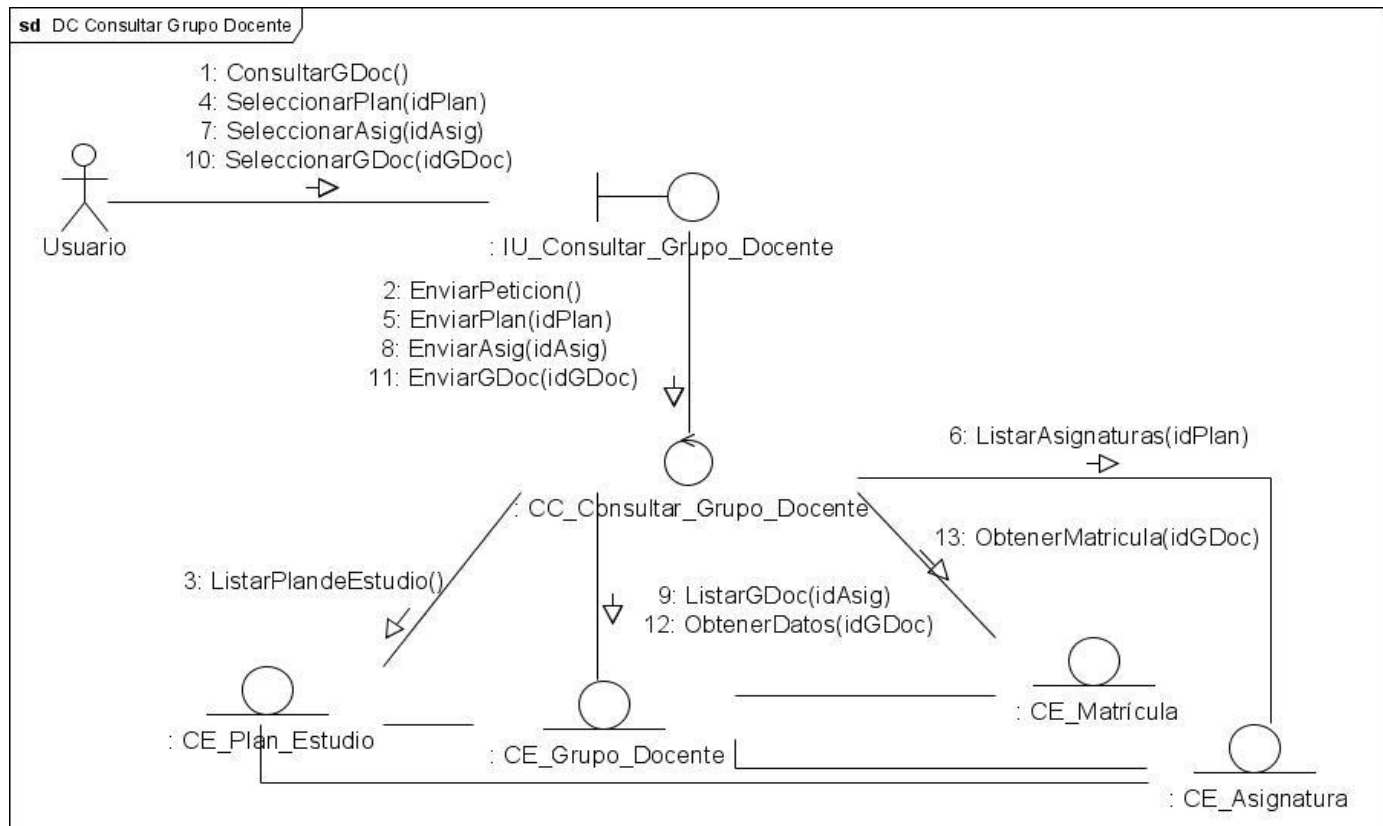


Figura 81. Diagrama de colaboración “Consultar Grupo Docente”

Anexo 6: Diagramas de secuencia

A

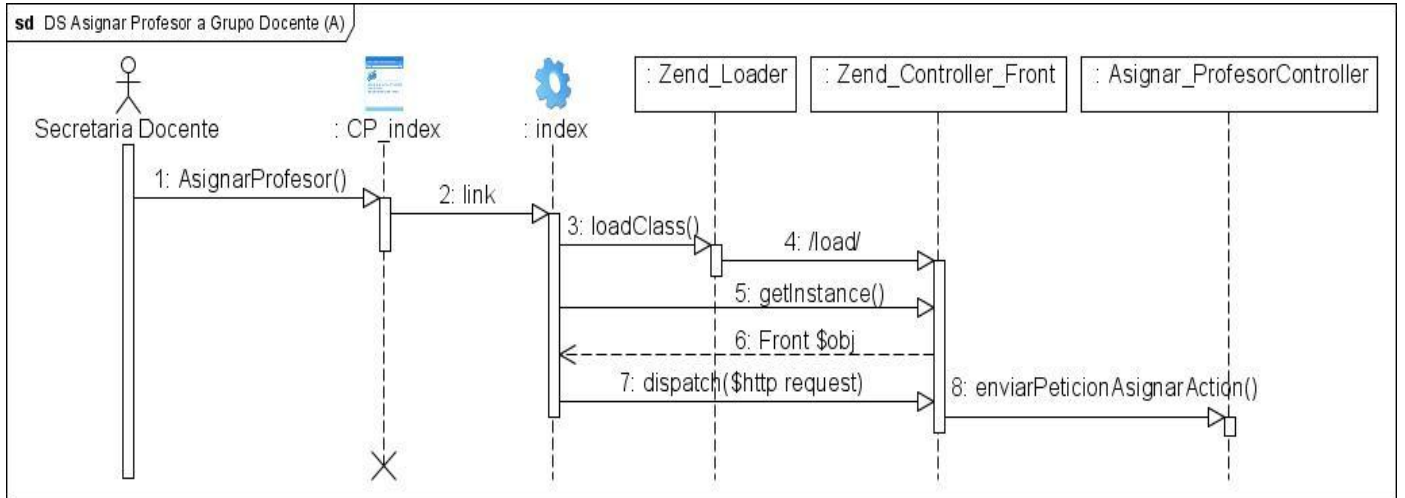


Figura 82. Diagrama de secuencia “Asignar Profesor a Grupo Docente (A)”

B

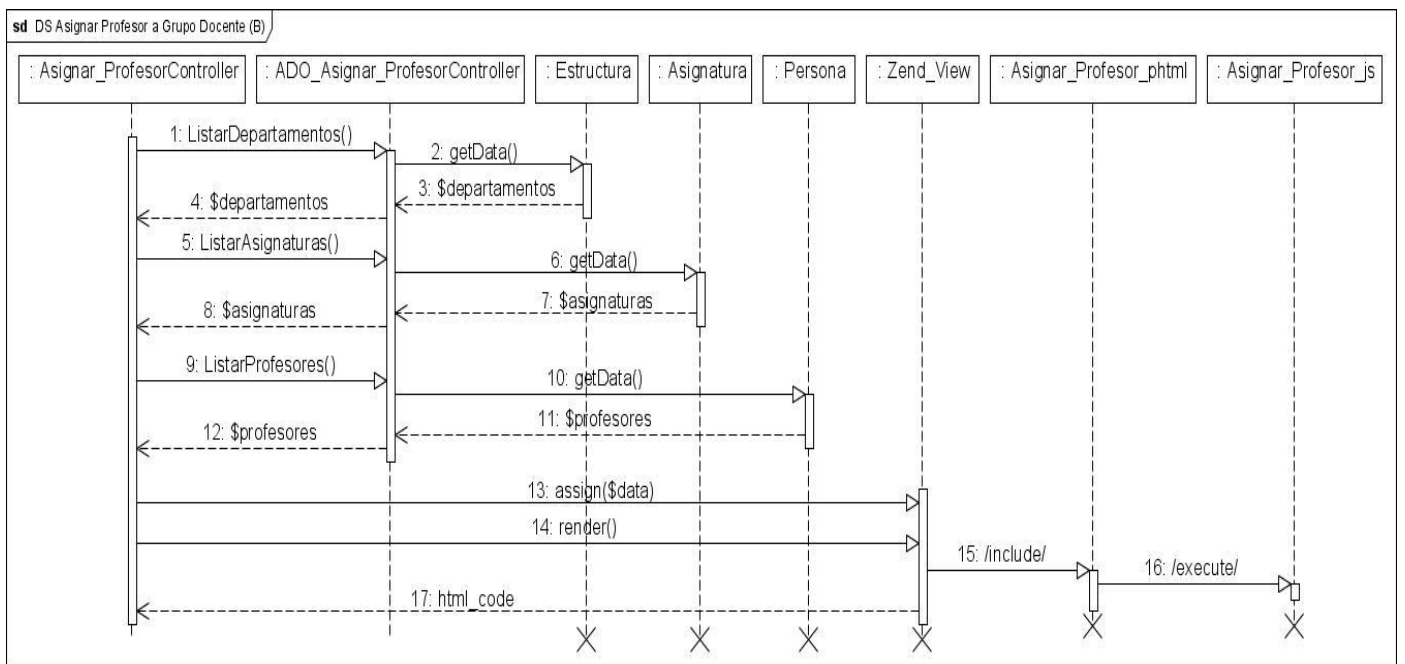


Figura 83. Diagrama de secuencia “Asignar Profesor a Grupo Docente (B)”

C

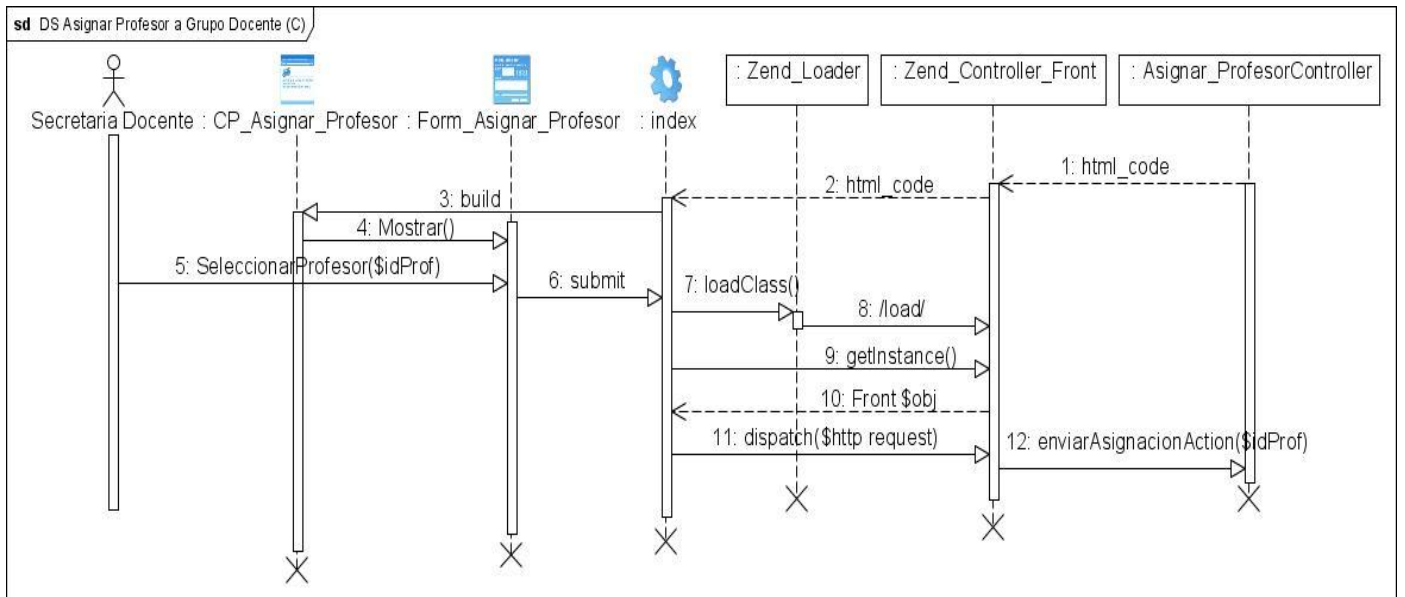


Figura 84. Diagrama de secuencia “Asignar Profesor a Grupo Docente (C)”

D

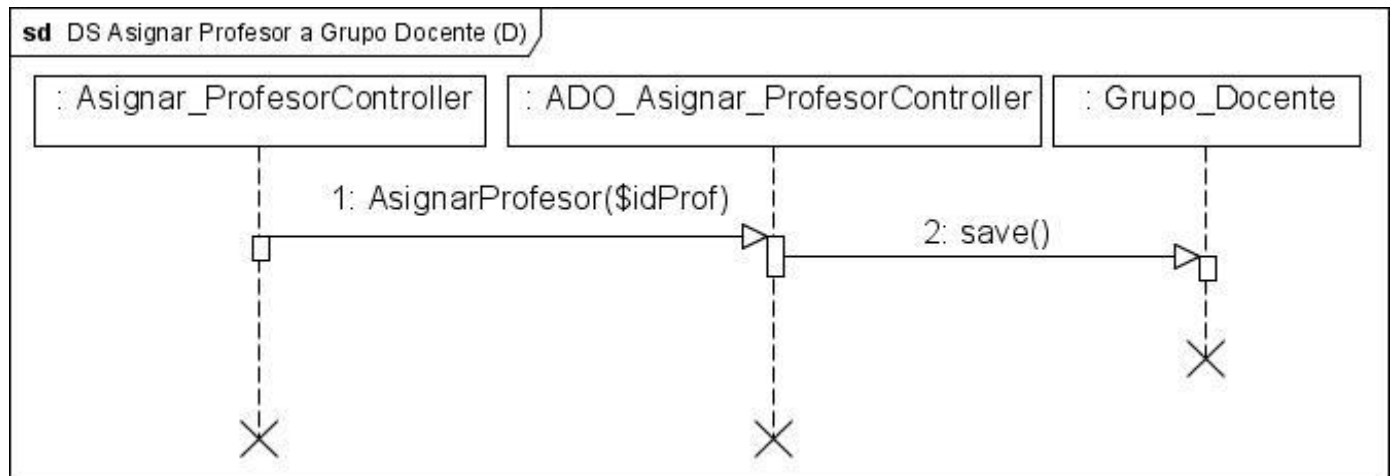


Figura 85. Diagrama de secuencia “Asignar Profesor a Grupo Docente (D)”

A

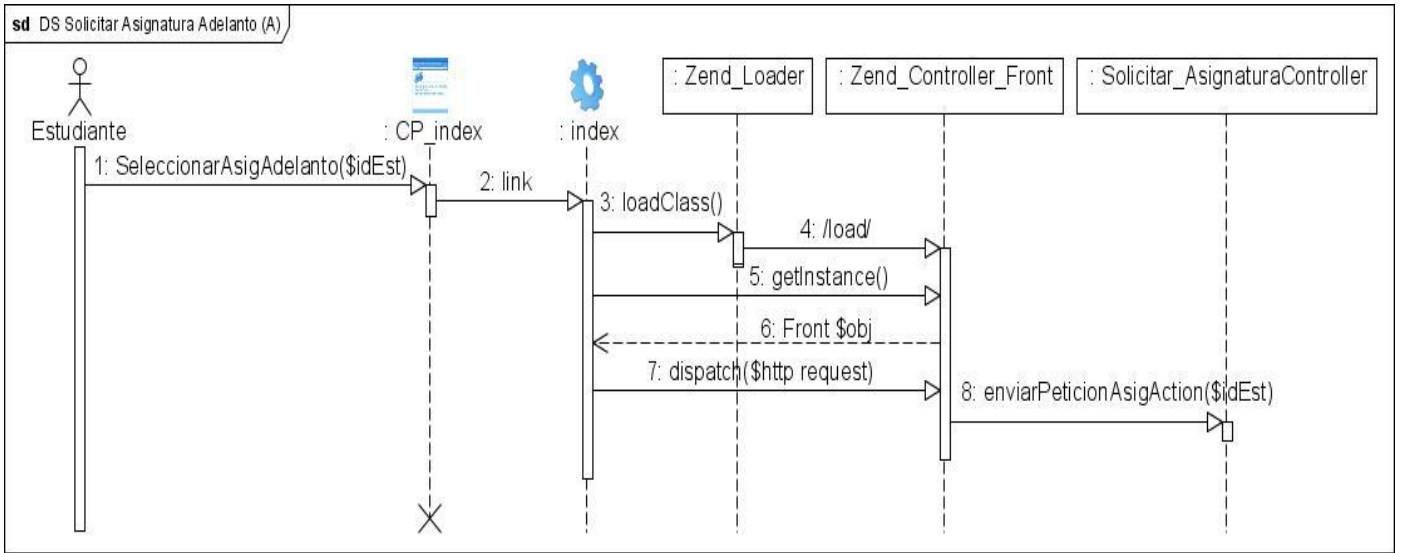


Figura 86. Diagrama de secuencia “Solicitar Asignatura Adelanto (A)”

B

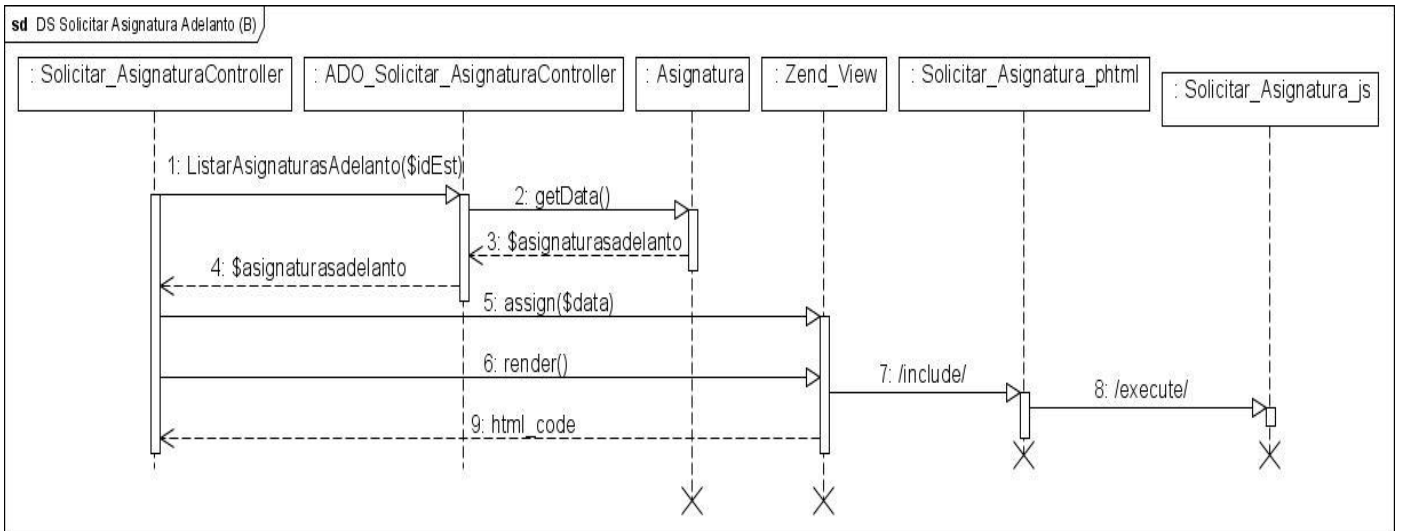


Figura 87. Diagrama de secuencia “Solicitar Asignatura Adelanto (B)”

C

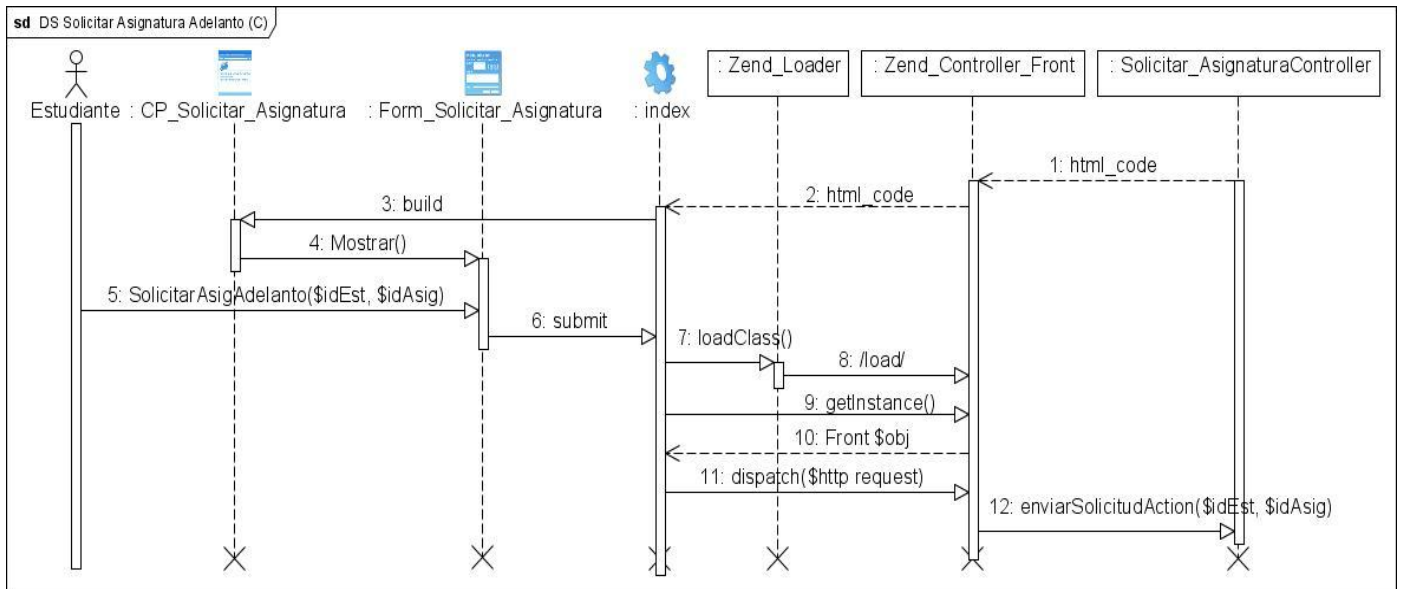


Figura 88. Diagrama de secuencia “Solicitar Asignatura Adelanto (C)”

D

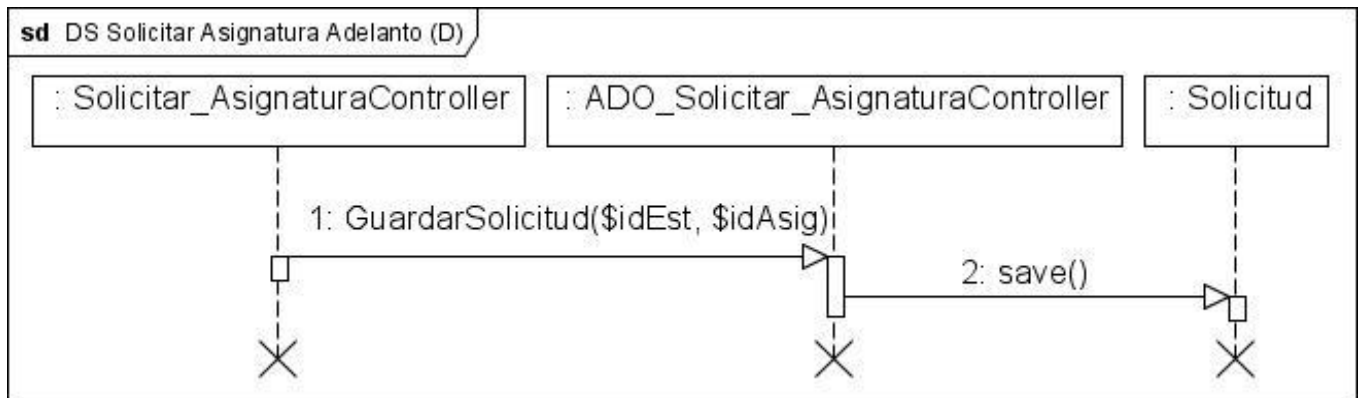


Figura 89. Diagrama de secuencia “Solicitar Asignatura Adelanto (D)”

A

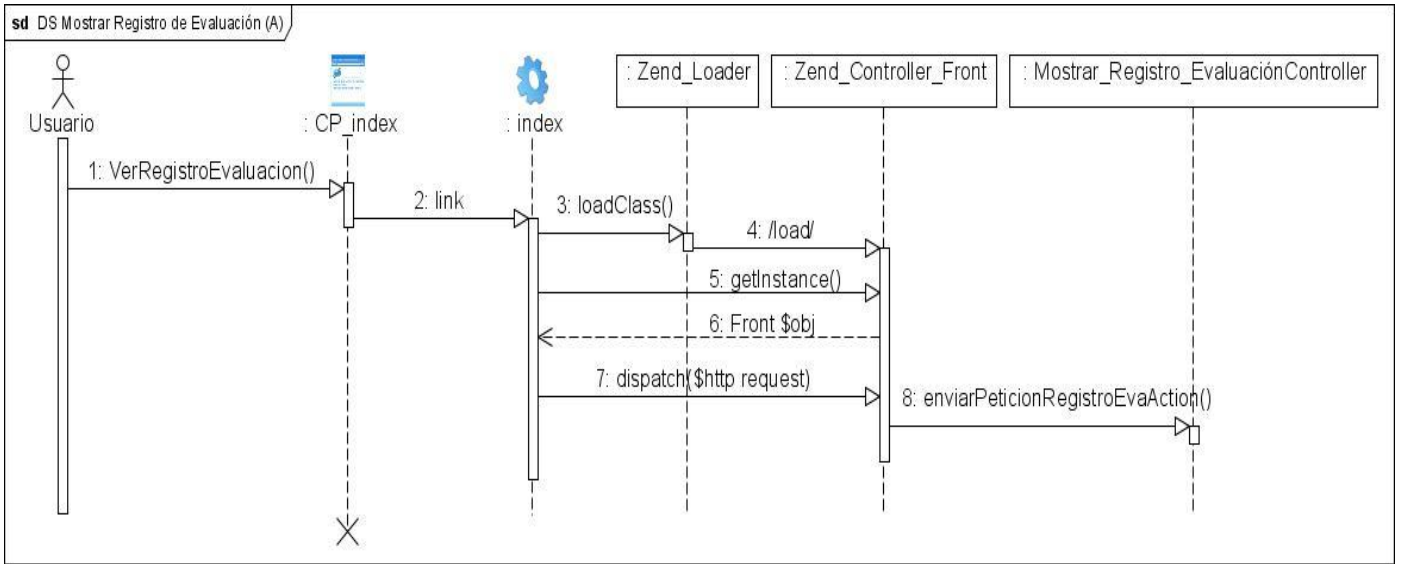


Figura 90. Diagrama de secuencia “Mostrar Registro de Evaluación (A)”

B

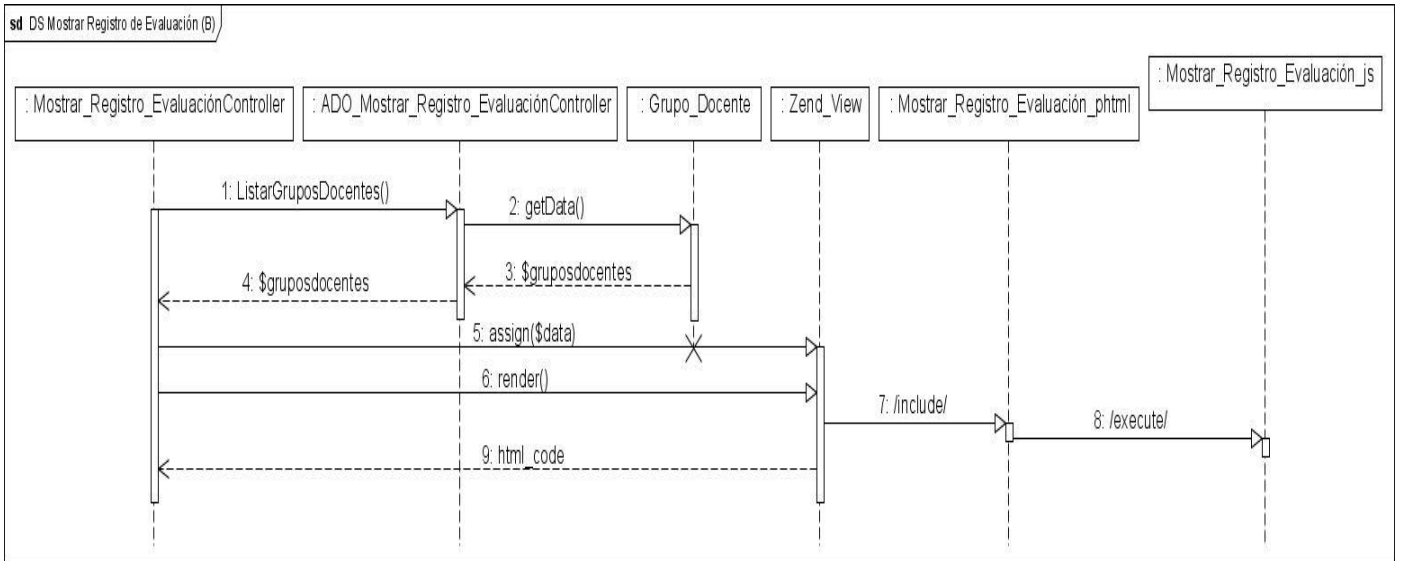


Figura 91. Diagrama de secuencia “Mostrar Registro de Evaluación (B)”

C

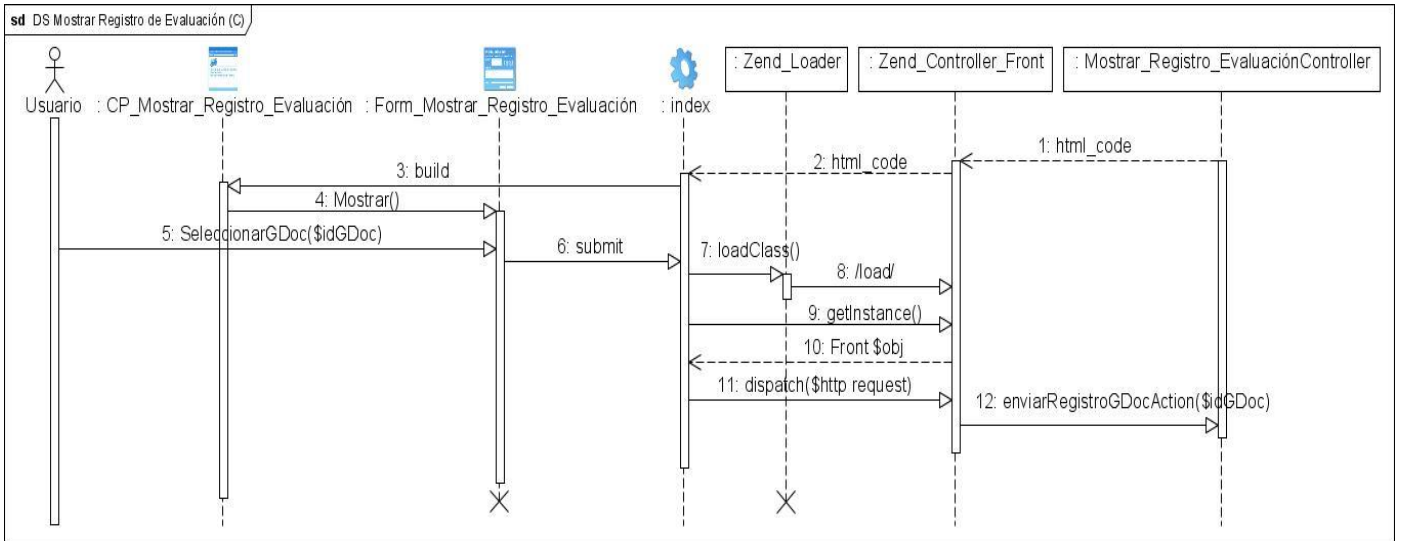


Figura 92. Diagrama de secuencia “Mostrar Registro de Evaluación (C)”

D

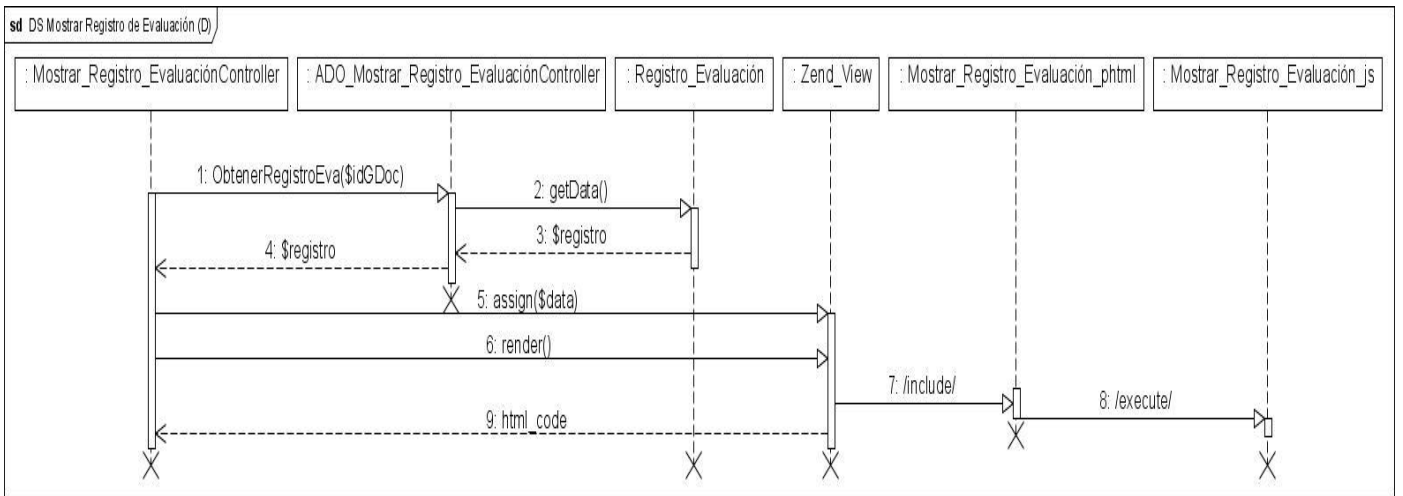


Figura 93. Diagrama de secuencia “Mostrar Registro de Evaluación (D)”

E

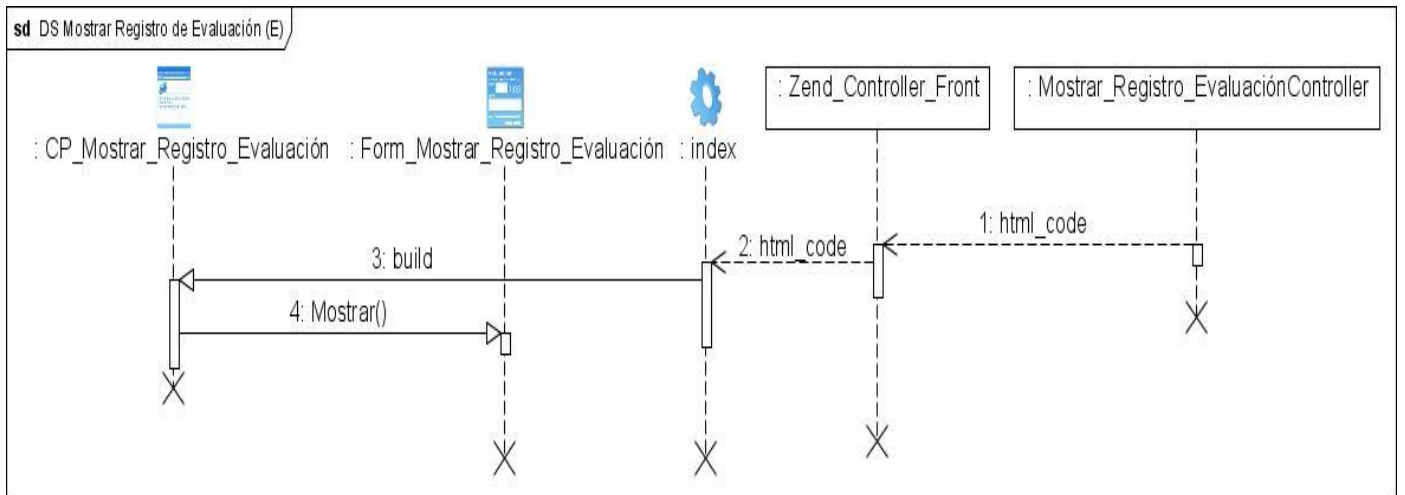


Figura 94. Diagrama de secuencia “Mostrar Registro de Evaluación (E)”

Anexo 7: Avaluos y Actas de Aceptación

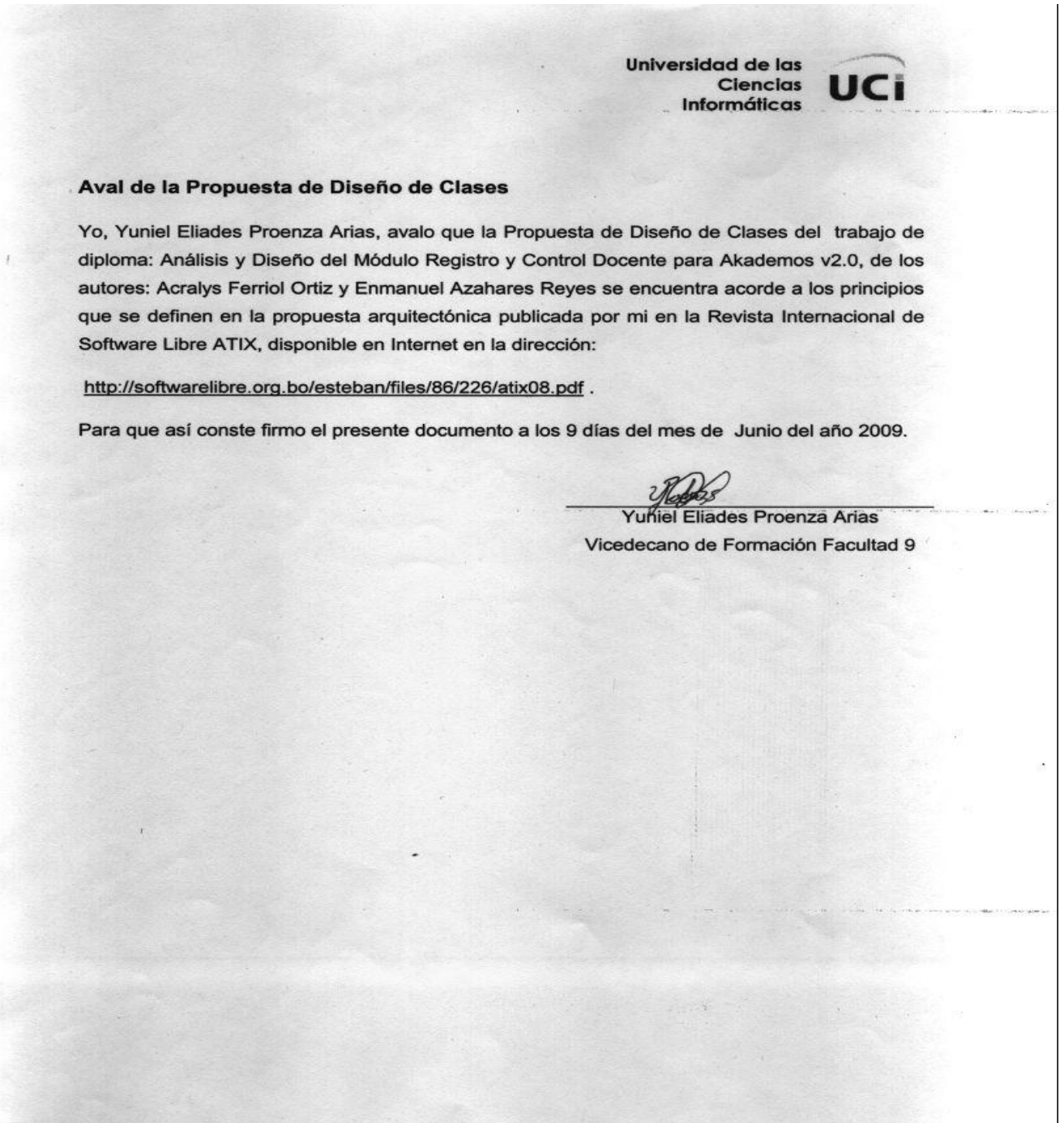


Figura 95. Aval de la propuesta de diseño de clases del Módulo Registro y Control Docente



Figura 96. Aval de la propuesta de diseño de clases del Módulo Registro y Control de los arquitectos del proyecto Akademos

ACTA DE ACEPTACIÓN

En la Ciudad de La Habana, a los 9 días del mes de junio del año 2009

De una parte, Facultad 1, representado en este acto por **Mayelin Alvarez Marichal**, quien a los fines y efectos derivados del presente documento se denominará como "**El Cliente**".

De otra parte, el Grupo de Desarrollo del Proyecto Gestión Académica, Akademos, representado en este acto por **Yanet Poza Bernal**, que a los fines y efectos derivados del presente documento se denominará como "**El Proyecto**".

Primero: Que en cumplimiento del Cronograma han sido efectuadas las actividades que se describen, **Las Partes DECLARAN:**

CONSIDERANDO: Que se han efectuado las actividades siguientes sobre el módulo **Registro y Control docente:**

1. Especificación de los requerimientos del software
2. Modelado del sistema

CONSIDERANDO: Que las actividades que se han ejecutado cumplen con los requerimientos de **El Cliente**.

CONSIDERANDO: Que **El Proyecto** ha mostrado la documentación que avala la ejecución de este acto a **El Cliente**.

POR TANTO: **Las Partes** acuerdan formalizar mediante la presente **Acta**, **Aceptadas** las actividades que han sido ejecutadas en esta fecha.

Y para que así conste, se extiende la presente **Acta** en dos ejemplares, rubricados por **Las Partes**.

Por El Cliente



Mayelin Alvarez Marichal
Secretaria Docente

Por El Proyecto



Yanet Poza Bernal
Lider del proyecto Akademos

Figura 97. Acta de Aceptación de la secretaría docente Facultad 1

ACTA DE ACEPTACIÓN

En la Ciudad de La Habana, a los 9 días del mes de junio del año 2009

De una parte, Facultad 9, representado en este acto por **Surama Borges Zayas**, quien a los fines y efectos derivados del presente documento se denominará como "**El Cliente**".

De otra parte, el **Grupo de Desarrollo del Proyecto Gestión Académica, Akademos**, representado en este acto por **Yanet Poza Bernal**, que a los fines y efectos derivados del presente documento se denominará como "**El Proyecto**".

Primero: Que en cumplimiento del Cronograma han sido efectuadas las actividades que se describen, **Las Partes DECLARAN:**

CONSIDERANDO: Que se han efectuado las actividades siguientes sobre el módulo **Registro y Control Docente:**

1. Especificación de los requerimientos del software
2. Modelado del sistema

CONSIDERANDO: Que las actividades que se han ejecutado cumplen con los requerimientos de **El Cliente**.

CONSIDERANDO: Que **El Proyecto** ha mostrado la documentación que avala la ejecución de este acto a **El Cliente**.

POR TANTO: **Las Partes** acuerdan formalizar mediante la presente **Acta, Aceptadas** las actividades que han sido ejecutadas en esta fecha.

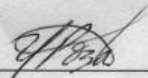
Y para que así conste, se extiende la presente **Acta** en dos ejemplares, rubricados por **Las Partes**.

Por El Cliente



Surama Borges Zayas
Secretaría Docente

Por El Proyecto



Yanet Poza Bernal
Líder del proyecto Akademos

Figura 98. Acta de Aceptación de la secretaría docente Facultad 9

ACTA DE ACEPTACIÓN

En la Ciudad de La Habana, a los 9 días del mes de junio del año 2009.

De una parte, Facultad 10, representado en este acto por **Milagros E. Ramírez Muñoz**, quien a los fines y efectos derivados del presente documento se denominará como "El Cliente".

De otra parte, el Grupo de Desarrollo del Proyecto Gestión Académica, Akademos, representado en este acto por **Yanet Poza Bernal**, que a los fines y efectos derivados del presente documento se denominará como "El Proyecto".

Primero: Que en cumplimiento del Cronograma han sido efectuadas las actividades que se describen, **Las Partes DECLARAN:**

CONSIDERANDO: Que se han efectuado las actividades siguientes sobre el módulo *Registro y Control Docente*:

1. Especificación de los requerimientos del software
2. Modelado del sistema

CONSIDERANDO: Que las actividades que se han ejecutado cumplen con los requerimientos de **El Cliente**.

CONSIDERANDO: Que **El Proyecto** ha mostrado la documentación que avala la ejecución de este acto a **El Cliente**.

POR TANTO: **Las Partes** acuerdan formalizar mediante la presente **Acta**, **Aceptadas** las actividades que han sido ejecutadas en esta fecha.

Y para que así conste, se extiende la presente **Acta** en dos ejemplares, rubricados por **Las Partes**.

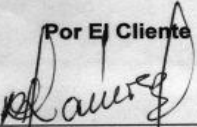

<p>Por El Cliente</p>  _____	<p>Por El Proyecto</p>  _____
<p>Milagros E. Ramírez Muñoz Secretaría Docente</p>	<p>Yanet Poza Bernal Líder del proyecto Akademos</p>

Figura 99. Acta de Aceptación de la secretaría docente Facultad 10

