

Universidad de las Ciencias Informáticas

Facultad 9



Sistema para la Secretaría Docente del Centro Politécnico del Petróleo

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

Autores:

Reinel Salgado González

Rolando Toledo Fernández

Tutor:

Alexey Díaz Domínguez

17 de junio del 2009

Año del 50 aniversario del triunfo de la Revolución

Agradecimientos

Quiero agradecer:

A mi familia en general, especialmente a mi padre por ser un guía para mí en todo momento.

A mi novia Anabel por todo el amor, comprensión y apoyo que siempre me ha brindado.

A todos mis compañeros de aulas, especialmente a Raúl y Heikel.

A mi tutor Alexey Díaz Domínguez por su apoyo.

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

---De Reinel.

Primeramente a mi familia, a todos sin excepciones y muy especialmente a mis padres, a mi hermana querida y a mi tío Jesús María que siempre me han apoyado en todo momento y han confiado ciegamente en mí.

A mis primos Reibel y Pablo que siempre están ahí en los momentos más difíciles. A nuestro tutor que nos ha ayudado para llegar a convertirnos en lo que hoy somos.

A mis amigos de todos estos años, El Villa, Helimay, Maikel Alejo y todos aquellos que estuvimos en el grupo 9101 en primero año.

A los amigos de los demás grupos en los que he estado Ronald, El Pícaro, Raul y todos los demás que han estado conmigo en los momentos más difíciles de esta escuela.

A unas muy buenas amigas que han estado conmigo en los buenos y malos momentos y que siempre me han apoyado en las decisiones que tomo y que entienden el porqué de mis actos.

Estas son Jackeline y Noraida como cariñosamente le digo.

A todas las nuevas amistades que he conocido en este 5to año y me han ayudado como Aimelys que no cumplía con Asdrúbal para cumplir conmigo.

A mi amigo Omar que aunque él no lo crea lo aprecio cantidad.

A todos los que me han hecho algún que otro favor en estos 5 años.

---De Rolando

Dedicatoria

A la memoria de mi querida Madre.

---De Reinel.

A mi familia y especialmente a mi abuela que quiero con el alma y a la memoria de aquellos que ya no están presentes físicamente pero que llevo muy dentro de mí.

---De Rolando.

Resumen

El presente trabajo de diploma tiene como objetivo desarrollar un sistema, que garantice la gestión eficiente de los procesos docentes en la secretaría del Centro Politécnico del Petróleo (CPP).

Para el desarrollo del sistema propuesto se analizaron las necesidades actuales del CPP y se tuvieron en cuenta las metodologías y tecnologías actuales propuestas por la Universidad de las Ciencias Informáticas para el desarrollo de aplicaciones Web. Se seleccionó principalmente la tecnología PHP como lenguaje de programación y PostgreSQL como sistema gestor de bases de datos.

La versión 1.0 ofrece un sistema multiplataforma, que brinda al personal del CPP un soporte para el almacenamiento de la información resultante de los procesos de negocio diarios de dicha secretaría docente. También facilita al personal, la gestión de los diferentes cursos que se imparten, la matricula de los estudiantes en los mismos; así como la generación de los diferentes reportes.

Índice

Agradecimientos.....	- 1 -
Dedicatoria.....	- 2 -
Introducción.....	- 9 -
Capítulo 1: Fundamentación Teórica.....	- 13 -
Introducción.....	- 13 -
Métodos científicos utilizados.....	- 13 -
1.1 La Gestión Académica.....	- 14 -
1.2 Análisis de soluciones existentes.....	- 14 -
1.2.1 En el mundo.....	- 14 -
MENTORVOX.....	- 15 -
1.2.2 En Cuba.....	- 15 -
Sistema Docente Integral (SDI).....	- 15 -
GESTACAD.....	- 16 -
1.2.3 En la UCI.....	- 16 -
Akademos.....	- 16 -
1.3 Sistema de Gestión Académica (SGA).....	- 17 -
1.3.1 Módulos de SGA.....	- 18 -
1.3.2 Módulo de Gestión.....	- 18 -
1.3.3 Módulo de Reporte.....	- 18 -
1.3.4 Módulo Administración.....	- 18 -
1.4 Tendencias y Tecnologías Actuales.....	- 18 -
1.4.1 Metodologías Existentes.....	- 19 -
1.4.1.1 Programación Extrema XP.....	- 19 -
1.4.1.2 SCRUM.....	- 19 -
1.4.1.3 Proceso Unificado de Rational.....	- 20 -
1.4.1.4 ¿Por qué RUP?.....	- 20 -
1.4.2 Lenguajes de Modelado.....	- 22 -
1.4.2.1 Lenguaje de Modelado OO i*.....	- 22 -

1.4.2.2	Lenguaje de Modelado OO UML	- 22 -
	Características de UML	- 22 -
1.4.2.3	¿Por qué UML?	- 23 -
1.4.3	Framework de Desarrollo	- 23 -
1.4.3.1	Symfony.	- 23 -
	Características	- 23 -
1.4.4	Lenguaje de Programación.	- 26 -
1.4.4.1	ASP	- 26 -
1.4.4.2	PHP	- 26 -
1.4.4.3	¿Por qué PHP?	- 27 -
1.4.5	Herramienta Case.	- 27 -
1.4.5.1	Rational Rose.	- 27 -
1.4.5.2	Visual Paradigm para UML	- 28 -
1.4.5.3	¿Por qué Visual Paradigm?	- 28 -
1.4.6	Gestor de Base de Datos.	- 28 -
1.4.6.1	MYSQL	- 28 -
1.4.6.2	PostgreSQL	- 28 -
1.7.6.3	¿Por qué PostgreSQL?	- 29 -
	Conclusiones	- 29 -
	Capitulo 2: Características del Sistema	- 30 -
	Introducción	- 30 -
2.1	Objeto de automatización.	- 30 -
2.2	Información que se Maneja.	- 30 -
2.3	Propuesta de Sistema.	- 32 -
2.4	Modelo del Negocio.....	- 33 -
2.5	Diagrama de Casos de Uso del Negocio	- 34 -
2.6	Descripción de los Casos de Uso del Negocio.....	- 35 -
2.6.1	CUN Gestionar Matrícula	- 35 -
	Descripción del CUN Gestionar Matrícula	- 35 -

Modelo de Objetos. CUN Gestionar Matricula.	- 39 -
2.6.2 CUN Gestionar Curso.....	- 39 -
Descripción del CUN Gestionar Curso	- 39 -
Modelo de Objetos. CUN Gestionar Curso	- 41 -
2.6.3 CUN Control de Evaluaciones de Curso.	- 41 -
Descripción del CUN Control de Evaluaciones de Curso.....	- 41 -
Modelo de Objetos. CUN Control de Evaluaciones de Curso	- 44 -
2.6.4 CUN Control de Evaluaciones de Diplomado.....	- 44 -
Descripción del CUN Control de Evaluaciones de Diplomado	- 44 -
Modelo de Objetos. CUN Control de Evaluaciones de Diplomado.....	- 46 -
2.6.5 CUN Control de Registro.	- 46 -
Descripción del CUN Control de Registro.	- 46 -
Modelo de Objetos. CUN Control de Registro.	- 48 -
2.6.6 CUN Gestionar Diplomado.	- 48 -
Descripción del CUN Gestionar Diplomado	- 48 -
Modelo de Objetos. CUN Gestionar Diplomado.	- 49 -
2.7 Especificación de los Requisitos de Software	- 49 -
2.7.1 Requerimientos Funcionales	- 50 -
2.7.2 Requerimientos No Funcionales	- 52 -
2.8 Definición de los Actores del Sistema	- 53 -
2.9 Diagrama de Casos de Uso del Sistema	- 54 -
2.10 Descripción de los Casos de Uso del Sistema	- 57 -
2.10.1 CUS Gestionar Matrícula	- 58 -
Conclusiones.....	- 60 -
Capitulo 3: Construcción de la Solución Propuesta.	- 61 -
Introducción.....	- 61 -
3.1 Arquitectura de Software.	- 61 -
Modelo Vista Controlador	- 62 -
3.2 Análisis.....	- 62 -

3.3	Diseño	- 62 -
3.3.1	Clases del diseño	- 62 -
3.3.2	Diagrama de Componentes.....	- 62 -
3.3.3	Diagrama de Despliegue.....	- 63 -
3.3.4	Diagrama de Clase Persistentes	- 63 -
3.3.5	Diseño de Base de Datos	- 63 -
3.3.5.1	Descripción de las tablas	- 64 -
	TipoCursos	- 64 -
	Curso	- 64 -
	Persona	- 65 -
	Cursista	- 65 -
	Profesor.....	- 65 -
	Conclusiones.....	- 66 -
	Capitulo #4: Estudio de Factibilidad.....	- 67 -
	Introducción.....	- 67 -
4.1	Técnica de Estimación.....	- 67 -
4.1.1	Análisis de Puntos de Función y COCOMO II.....	- 67 -
4.1.2	Análisis de Puntos de Casos de Uso.....	- 67 -
4.2	Estimación de Tiempo y esfuerzo utilizando Análisis de Punto de Casos de Uso.....	- 67 -
4.3	Ajustando los Puntos de Casos de Uso.....	- 68 -
	Conclusiones.....	- 72 -
	Conclusiones Generales.....	- 73 -
	RECOMENDACIONES	- 74 -
	Bibliografía Consultada	- 75 -
	Referencias Bibliográfica	- 76 -
	Anexos.....	- 78 -
	Glosario de Términos.....	- 78 -
	Anexo # 1	- 81 -
	DA Gestionar Matrícula	- 81 -
	DA Gestionar Curso	- 82 -

DA Control de Evaluaciones de Curso	- 83 -
DA Control de Evaluaciones de Diplomados	- 84 -
DA Control de Registro	- 85 -
DA Gestionar Diplomado	- 86 -
Anexo #2.....	- 87 -
Descripción del CUS Gestionar Curso	- 87 -
Descripción del CUS Gestionar Diploma	- 90 -
Descripción del CUS Gestionar Diplomado	- 93 -
Descripción del CUS Gestionar Especialidad	- 95 -
Descripción del CUS Gestionar Postgrado	- 99 -
Anexo #3.....	- 102 -
DCA Gestionar Matrícula	- 102 -
DCA Gestionar Tipo de Curso	- 102 -
DCA Gestionar Diploma	- 103 -
DCA Gestionar Curso	- 103 -
Anexo # 4.....	- 104 -
DCD Gestionar Matrícula	- 104 -
DCD Gestionar Tipo de Curso	- 105 -
DCD Gestionar Diploma.....	- 106 -
DCD Gestionar Curso.....	- 107 -
Anexo # 5.....	- 108 -
Diagrama de Componentes.....	- 108 -
Anexo # 6.....	- 109 -
Diagrama de Despliegue.....	- 109 -
Anexo # 7	- 110 -
Anexo # 8.....	- 111 -

Introducción

En los últimos años las tecnologías de la información y las comunicaciones (TIC) han tenido un desarrollo asombroso, al mismo tiempo han evolucionado también las formas de operar de las empresas actuales, gracias a la vinculación de las TIC en el desempeño diario de estas. “En Marzo de 1993, la Web suponía el 0,1% del tráfico total de Internet y el Centro Europeo de Investigación Nuclear (CERN), declaraba a la WWW¹ como una tecnología de acceso libre. En septiembre de 1993 el uso de la web ya alcanzaba el 1% del tráfico de Internet y existían versiones del navegador Mosaic para PC y Macintosh. En octubre de 1993, habían unos 500 servidores web activos. A partir de ahí, su crecimiento es milagroso; a finales de 1994 existían ya más de 10.000 servidores y 10 millones de usuarios. Y en 1997 la cifra superaba los 650.000 servidores”. (Baeza-Yayes, 1993). Hoy en día ya se cuenta con servidores de todo tipo, capaces de ofrecer a los usuarios hasta servicios de video en línea. Esto ha permitido que las empresas logren verter todo o parte de sus flujos de trabajos diarios hacia el mundo digital e internet. Actualmente estas empresas están insertadas en el mundo de la web utilizando tecnologías de punta que reflejan los productos y servicios que resultan de sus procesos de negocios.

Un ejemplo de informatización en disímiles esferas de la sociedad, es el uso por parte de entidades docentes, de sistemas en línea para la gestión académica; permitiendo así que profesores, estudiantes y familiares tengan pleno acceso a los resultados académicos, derribando las barreras que impone la sociedad moderna.

En esta batalla global por informatizar todos los aspectos de la vida social y laboral, las empresas cubanas no han sido la excepción, es por esto que surge la Universidad de las Ciencias Informáticas (UCI), para llevar a cabo el avance de la informática en el país, objetivo crucial en el desarrollo tecnológico de la revolución cubana.

Cuba y el sistema social elegido en 1959 han visto siempre como un sector priorizado al sector educacional, así como a todas las instituciones que de una forma u otra intervienen en la capacitación y superación de los profesionales cubanos, de ahí que se ponga tanto empeño en informatizar este sector.

El Centro Politécnico del Petróleo (CPP) es una entidad perteneciente a la UNION CubaPetróleos (CUPET) y su principal objetivo es capacitar el personal que labora para la UNION. Este centro está ubicado en Ciudad de La Habana en el municipio de Regla y lleva a cabo su tarea a través de la

¹ WWW: WORD WIDE WEB

impartición de cursos de postgrado y diplomados en los temas referentes a los distintos procesos de la producción de petróleo.

Como todo centro docente, el CPP cuenta con una secretaría donde se mantiene un control sobre sus estudiantes, cursos que se imparten y demás procesos docentes. En la actualidad todas estas actividades se realizan de forma manual, lo cual hace que el trabajo del personal que allí labora sea engorroso, dando como resultado que se atrasen las entregas de los títulos y demás documentos que deben ser emitidos por dicha secretaría. Todo el volumen de información que se genera producto al cumplimiento de las actividades correspondientes a los trabajadores de la secretaría es archivado en formato duro, estando así a pesar de las medidas de seguridad tomadas expuesto a un posible deterioro que pudiese ocasionar la pérdida total o parcial de esta información.

En la Secretaría del CPP se llevan a cabo seis procesos fundamentales los cuales son relacionados a continuación:

- Gestión de Matrícula: proceso que rige todas las acciones que tienen que ver con la inscripción de estudiantes en los cursos que se ofertan.
- Gestión de Curso: en este proceso se lleva a cabo el control de todas las acciones que se relacionan con los cursos desde el mismo instante en que se elabora el plan, hasta las inscripciones de los estudiantes en los mismos.
- Gestión de Diplomado: con este proceso se controla la gestión de los diplomados.
- Control de Evaluaciones de Diplomados: en este proceso se lleva a cabo el control de todas evaluaciones de los diplomados.
- Control de Evaluaciones de Curso: aquí se controla el proceso de evaluación de los cursistas a partir de que los mismos se presentan a su evaluación final.
- Control de Registro: a través de este proceso se controla la elaboración, entrega y recogida del registro de asistencia y calificaciones.

En ocasiones la dirección del CPP pide a la secretaria cierta información de un año determinado. Al ser esta almacenada en formato duro hace que una búsqueda sea engorrosa y en ocasiones se realice en vano o no se logre obtener toda la información que se necesita debido al gran volumen de ella que se maneja.

Producto a la situación antes expuesta el CPP ha solicitado a la Universidad de Ciencias Informáticas, específicamente en la facultad 9 al Polo Petrosoft, la búsqueda de una solución a esta situación.

Sistema para la Secretaria Docente del Centro Politécnico del Petróleo

Con el objetivo de brindar una respuesta a esta solicitud se ha comenzado una investigación que se inicia identificando como **problema a resolver** la inexistencia de un sistema en secretaría docente del Centro Politécnico del Petróleo que permita informatizar los procesos.

Se decidió como **objetivo general** desarrollar un sistema que permita la gestión de la información de los procesos realizados en la secretaría docente del CPP. Para dar respuesta al objetivo general se determinaron los siguientes **objetivos específicos**:

1. Identificar los procesos de negocio de la secretaría docente del Centro Politécnico del Petróleo para definir el alcance y la visión del negocio a informatizar.
2. Elaborar los artefactos generados por las diferentes fases del ciclo de desarrollo del software.

Tomando como **objeto de estudio** los procesos de negocio de gestión en las secretarías docentes y como **campo de acción** los procesos de negocio de secretaría docente del Centro Politécnico del Petróleo.

El desarrollo de dicho sistema estará sustentado en la utilización de herramientas como el sistema gestor de base de datos, el framework de desarrollo y el lenguaje de programación que se seleccione, cuya selección será resultado de un estudio comparativo entre las tendencias y tecnologías actuales, dando una medida a la solución propuesta.

Idea a Defender

Si se desarrolla un sistema que garantice informatizar los principales procesos de negocio de la secretaría docente en el Centro Politécnico del Petróleo, se logrará la gestión de la información, además de dar respuesta en menor tiempo a reportes solicitados por la dirección del politécnico.

Estructuración del contenido

El trabajo está estructurado en cuatro capítulos.

- En el Capítulo 1 se aborda la fundamentación teórica de la investigación. En éste se incluye el estudio del estado del arte de los Sistemas de Gestión Académica (SGA) a nivel internacional, nacional y de la Universidad. Así como las tendencias y tecnologías relacionadas con este tema en la actualidad, que son la base para la solución del problema.

- En el Capítulo 2 se abordan las características del sistema. Dentro de estas se encuentran el objeto de automatización, información que se maneja, propuesta de sistema, modelo de negocio así como la especificación de los requerimientos del software y la definición de los casos de uso.
- En el Capítulo 3 se expone el Análisis y Diseño del sistema que se va a desarrollar a través de modelo de clases del análisis, realización de los casos de uso, diagramas de interacción, diagramas de clases y la descripción correspondiente a cada una de las clases además del diagrama entidad relación y las descripciones de cada una de las entidades contenidas en él.
- Finalmente en el Capítulo 4 se abordará sobre la Implementación y Prueba del sistema, se detallan aspectos relacionados a la arquitectura, se exponen los diagramas de implementación y despliegue así como los artefactos generados durante la puesta en marcha del sistema.
- En las páginas finales de la tesis se encuentran las secciones de las conclusiones, bibliografía, anexos y glosario de términos.

Capítulo 1: Fundamentación Teórica.

Introducción

En este capítulo se exponen los métodos científicos en que se basa la presente investigación, así como los principales conceptos asociados al dominio del problema, estado del arte y se hace un análisis de las tecnologías y tendencias actuales, específicamente de los sistemas de gestión académica.

Métodos científicos utilizados.

Durante todo el proceso investigativo que se realizó se utilizó un conjunto de métodos científicos de investigación. Estos métodos se clasifican en:

- Teóricos: Constituyen el enfoque general para abordar los problemas científicos, de ahí que posibiliten profundizar en las regularidades y cualidades esenciales de los fenómenos.
- Empíricos: Permiten la obtención y elaboración de los hechos fundamentales que caracterizan a los fenómenos, a la par que facilitan confirmar hipótesis y teorías.

Dentro de los teóricos se emplearon los siguientes:

- Análisis histórico lógico: para estudiar el estado del arte sobre las aplicaciones informáticas de este tipo implementadas en Cuba y el resto del mundo y su uso en el ámbito nacional, así como las ventajas y desventajas que poseen.
- Analítico-Sintético: para resumir, enunciar y describir los requerimientos funcionales.
- Modelación: para realizar una reproducción simplificada de la realidad. Permite descubrir nuevas relaciones y cualidades del objeto de estudio. Específicamente para generar los artefactos correspondientes a los flujos de trabajo en los que se trabajará.

Dentro de los empíricos se emplearon:

- Entrevistas individuales y colectivas: con el personal que labora en la secretaría del CPP para comprender la situación real del problema existente, así como las opiniones y sugerencias de los mismos.
- Observación: para realizar un registro visual de lo que ocurre en el entorno del problema y aportar nuevos elementos que puedan ser de interés científico.

1.1 La Gestión Académica.

La gestión académica tanto en escuelas, universidades como para otros centros dedicados a la enseñanza, es el proceso mediante el cual se controla, organiza y dirige las actividades del proceso docente educativo; entre los que resaltan la matrícula de los estudiantes a los diferentes cursos que se imparten, la gestión de programa de cursos y el control de asistencia y calificaciones de los profesores. En este proceso intervienen principalmente dirigentes, profesores y secretarías.

1.2 Análisis de soluciones existentes.

1.2.1 En el mundo.

A continuación se caracterizan algunos sistemas de gestión académica usados hoy en el mundo, Especificando puntos a favor y en contra de cada uno de ellos.

UNISIGA2 v1.1.1 (Sistema de Gestión Académico/Administrativo-Versión Universidad)

UNISIGA2 en su versión 1.1.1 es una aplicación conformada para gestionar la parte académica y administrativa de una empresa de carácter educativo. Es una aplicación bajo licencia GPL² y está desarrollada sobre entorno web.

Funciones

- Gestión de Portal [Totalmente configurable, gestión noticiosa, acceso a los diferentes calendarios creados con el operador Google, formulario de acceso al SIGA2, formulario de acceso directo al Moodle, gestión de consultas al SIGA2, Utilidades basadas en los gadget's³ creados por la comunidad Google, Acceso vía RSS⁴, Lector RSS, Estado del Clima, etc.].
- Gestión de archivos y directorio haciendo uso del PHPfileNavigator⁵.
- Gestión de estadísticas de uso del portal, basado en el Google analytics.
- Gestión de autenticación.
- Sistema de Control de acceso.
- Mantenimiento de tablas generales.
- Sub sistema de gestión empresarial [Personal, Empresas].
- Sub sistema de gestión académico [Apoderados, Alumnos, Semestre académico].

² Licencia Publica General de GNU

³ Aplicaciones pequeñas con vínculos en internet

⁴ Familia de protocolos usados en Internet

⁵ Navegador web desarrollado en php

- Sub sistema de gestión administrativo [Caja, Cuentas por rendir, Facturas, boletas, control de gastos, anulación y reemplazo de documentos]. (Zapata, 2008)

MENTORVOX

Establece un canal de comunicación entre los estudiantes, familiares y tutores y muestra toda la información que se genera día a día de cada estudiante como son notas, absentismo, indisciplinas, horarios, comedor, circulares, etc.

Características de MENTORVOX:

- Dispone de dos canales de comunicación de acceso: Internet y Teléfono.
- Segmenta la información a la que se puede acceder por los canales de comunicación en:
 - Información General del Centro: Hace referencia a horarios de Secretaría, comunicados de centros y cualquier otra información de carácter general.
 - Información de Grupo: Información relativa al grupo o clase del alumno; horario del grupo, horario de atención del tutor, circulares, fecha de la próxima reunión de padres, o fecha de la próxima entrega de notas.
 - Información del Alumno: Información relativa al seguimiento diario de cada alumno de manera personalizada; absentismo escolar (faltas y retrasos), incidencias de conducta o actitud, incidencias de rendimiento escolar (tareas, exámenes) y seguimiento de comedor.(ESENTIA, 2004)

1.2.2 En Cuba.

A continuación se describen algunos de los Sistemas de gestión académica usados en los diferentes centros educacionales cubanos.

Sistema Docente Integral (SDI).

Desarrollado por el Instituto Superior Politécnico “José Antonio Echeverría” (CUJAE). Está conformado por 8 módulos y se trabaja para aumentar sus funcionalidades.

Los módulos que están diseñados son:

- 1) Secretaría de Facultades.
- 2) Matrícula.
- 3) Estadística.
- 4) Archivo Histórico.

5) Recuperación de Información.

6) Pago de Estipendios.

7) Planificación Docente.

8) Postgrado.

De forma general el objetivo del sistema es registrar, procesar y crear mecanismos de recuperación de la información relacionada con el proceso docente, tanto de pregrado, como de postgrado, de forma automatizada. (Alfaro, 2004)

GESTACAD

“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.

El sistema consta de los siguientes módulos:

- Administración para la gestión de las tablas del sistema vía Web así como agregar nuevas consultas al sitio oficial y establecer los distintos niveles de acceso a estas.
- Web para las Secretarías Docentes para la Gestión de Estudiantes.
- Web para los Jefes de Departamentos Docentes donde se incluyen acciones relativas como la asignación de la carga docente y el control sobre los profesores del Dpto.
- Web para la Gestión de la Matrícula.
- Web para los Profesores donde estos pueden llevar el control docente de sus estudiantes, el control de las evaluaciones así como reportes relativos a su carga docente.
- Un Sitio Web con Reportes en Línea con la utilidad del registro docente para los profesores, además de la búsqueda de estudiantes la cual devuelve, algunos datos personales del estudiante. (González, 2001)

1.2.3 En la UCI.

Akademios

Es el Sistema de Gestión Académica de la UCI, y tiene como objetivos, permitir la gestión automatizada de los elementos que intervienen en la labor académica de un centro de estudios y enfrentar los cambios de forma natural, adaptándose a las nuevas condiciones y formas de hacer con el menor costo. El mismo se diseñó teniendo en cuenta los siguientes principios:

- “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 estudios es la entidad fundamental del proceso de gestión académica y rige todos sus subprocesos (matrícula, control, planificación, etc.)

Akados está dividido en siete módulos, que abarcan todos los procesos involucrados en la gestión docente. Estos módulos interactúan entre sí para llevar a cabo cada una de las tareas que automatiza el sistema”. (DALMENDRAY, 2006) Dichos módulos se relacionan a continuación:

- Matrícula.
- Estudiante.
- Registro.
- Profesores.
- Plan de estudio.
- Reportes.
- Seguridad.

1.3 Sistema de Gestión Académica (SGA)

SGA es un sistema de gestión académica que se desarrolla por diplomantes de la UCI, con el objetivo de llevar a cabo la automatización de los procesos en la Secretaría Docente del Centro Politécnico del Petróleo. El mismo estará compuesto por tres módulos principales que son Gestión, Reporte y Administración.

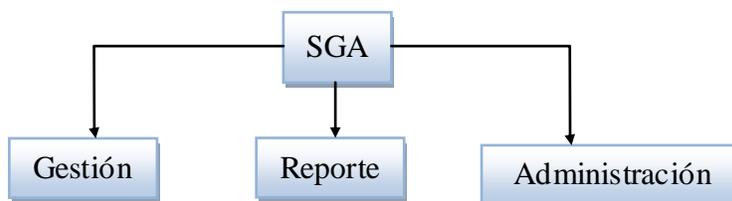
Principales Características

SGA es un sistema de gestión académico desarrollado con las bases de:

- El dinamismo de los procesos de gestión académica; que constituyen la principal fuente para un sistema que intente automatizarlos.
- Garantizar que todo el personal involucrado en los procesos de gestión académica (directivos, profesores, secretaria) tengan un papel activo en el sistema que se desarrolla.
- Garantizar la gestión eficiente de la matrícula ya que constituye el proceso más importante de las aplicaciones que automaticen los procesos de gestión académica, y rige todos los subprocesos que en el mismo se encuentran (reportes, creación de registros, etc.).

1.3.1 Módulos de SGA

SGA está compuesto por tres módulos, que son Administración, Gestión y Reporte de los cuales los dos últimos son los de mayor importancia ya que dan respuesta a los procesos que se desarrollan en la secretaría del CPP.



1.3.2 Módulo de Gestión.

El módulo de gestión es el modulo rector del trabajo que se realiza en el sistema. En él se gestiona la información de todos los cursos, estudiantes, maestrías, especialidades, diplomados y postgrados que se imparten en el CPP.

1.3.3 Módulo de Reporte.

En el módulo reporte se tienen una serie de reportes que se generan de la información que se almacena de cada uno de los cursos que se tiene en el CPP. Además se genera información entregable a los directivos del centro. Este módulo permite además que todos los reportes generados puedan ser exportados a formato PDF.

1.3.4 Módulo Administración.

Este módulo es el encargado de la seguridad de los elementos que se tienen en el sistema. Es el que permite dar privilegios a nuevos usuarios además de tener la característica de ser un módulo reutilizable pues cuenta con una base de datos independiente que le da soporte.

1.4 Tendencias y Tecnologías Actuales.

La gestión de la información mediante la utilización de hardware y software como un sistema informático se encuentran dentro de las TIC. Dichas tecnologías utilizan medios informáticos para la difusión, el procesamiento y el almacenamiento de la información. Estos procesos en la actualidad actúan cada vez más en los diferentes ámbitos de la sociedad como son en la formación pedagógica, la organización, la gestión empresarial e incluso la educación y precisamente dentro de esta se encuentra la gestión académica como el proceso principal dentro de la misma. Cada año que pasa es mayor la cantidad de

estudiantes que matriculan en centros de enseñanza de diferentes tipos y otros que después de graduarse quieren superarse aun más y por tal motivo siguen sus estudios en busca de nuevos conocimientos.

Esto trae consigo que un gran cúmulo de información sea generado y manipulado de forma manual. Provocando que sea demasiado lento el proceso para la gestión por parte del personal vinculado a estos trabajos.

1.4.1 Metodologías Existentes.

En el mundo de hoy en que el desarrollo de software ha alcanzado un alto nivel, los desarrolladores se han visto en la obligación de buscar técnicas que estandaricen el trabajo de las aplicaciones que se crean. Teniendo en cuenta lo anterior se han desarrollado un conjunto de metodologías para el desarrollo de software que hace posible que todo el personal vinculado a un proyecto determinado pueda entenderse de la misma forma.

1.4.1.1 Programación Extrema XP

La metodología Extreme Programming (XP) o Programación Extrema, es una de variantes de las metodologías ágiles con más aceptación en la comunidad internacional de desarrollo. Los fundamentos de la misma según su creador son: mejorar la comunicación, buscar la simplicidad, buscar retroalimentación en que también va el trabajo y que siempre hay que proceder con valentía.

Una de las herramientas más importantes en la metodología XP es el desarrollo orientado a pruebas, que utiliza las pruebas unitarias como eje de todo desarrollo. Las interacciones suelen ser muy cortas y se promueve a los programadores a buscar soluciones y experiencia con ellas, programar sin miedo a descomponer el sistema.

1.4.1.2 SCRUM

SCRUM es un proceso ágil y liviano que sirve para administrar y controlar el desarrollo de software. El desarrollo se realiza en forma iterativa e incremental (una iteración es un ciclo corto de construcción repetitivo). Cada ciclo o iteración termina con una pieza ejecutable que incorpora una nueva funcionalidad. En SCRUM el equipo se enfoca solamente en construir software de calidad. La gestión de proyecto se focaliza en definir las características del software a construir y remover cualquier obstáculo que pudiera entorpecer el trabajo del equipo de desarrollo. Se busca que los equipos de desarrollo sean lo más productivos posible. (Schwaber, 2004)

1.4.1.3 Proceso Unificado de Rational.

Es un proceso de ingeniería de software que provee un enfoque para asignar tareas y responsabilidades dentro de una organización de desarrollo. Su objetivo es asegurar la producción de software de alta calidad que satisfaga la necesidad del usuario final dentro de un tiempo y presupuesto previsible. El RUP mejora la productividad del equipo ya que permite que cada miembro del grupo sin importar su responsabilidad específica acceda a la misma base de datos de conocimiento. Esto hace que todos compartan el mismo lenguaje, la misma visión y el mismo proceso acerca de cómo desarrollar software. (Fernández, 2000)

1.4.1.4 ¿Por qué RUP?

RUP es un proceso para el desarrollo de un proyecto que tiene claramente quién, cómo, cuándo y qué debe hacerse en el proyecto. Tiene tres características esenciales está dirigido por casos de uso: que orientan el proyecto a la importancia para el usuario y lo que este quiere, está centrado en la arquitectura: que realiza la toma de decisiones que indica cómo tiene que ser construido el sistema y en qué orden y es iterativo e incremental: donde divide el desarrollo en mini proyectos donde los casos de uso y la arquitectura cumplen su objetivo de manera más depurada.

RUP maneja 6 claves principales:

- **Adaptación de procesos:** El proceso debe adaptarse a las características de la propia organización. El tamaño del mismo, así como las regulaciones que lo condicionen, influirán en su diseño específico. También se deberá tener en cuenta el alcance del proyecto.
- **Balancear prioridades:** Los requerimientos de los diversos inversores pueden ser diferentes, contradictorios o disputarse recursos limitados. Debe encontrarse un balance que satisfaga los deseos de todos.
- **Colaboración entre equipos:** El desarrollo de software no lo hace una única persona sino múltiples equipos. Debe haber una comunicación fluida para coordinar requerimientos, desarrollo, evaluaciones, planes, resultados, etc.
- **Demostrar valor iterativamente:** Los proyectos se entregan, aunque sea de un modo interno, en etapas iteradas. En cada iteración se analiza la opinión de los inversores, la estabilidad y calidad del producto, y se refina la dirección del proyecto así como también los riesgos involucrados
- **Elevar el nivel de abstracción:** Este principio dominante motiva el uso de conceptos reutilizables tales como patrón del software, lenguajes 4GL o esquemas (frameworks) por nombrar algunos.

Éstos se pueden acompañar por las representaciones visuales de la arquitectura, por ejemplo con UML.

- **Enfocarse en la calidad:** El control de calidad no debe realizarse al final de cada iteración, sino en todos los aspectos de la producción.

RUP divide el proceso en 4 fases principales, de las cuales se realizan varias iteraciones según el proyecto y en las que se hace un mayor hincapié en las distintas actividades. (Pablo, 2007)

Las actividades de RUP se centran crear y mantener modelos utilizando UML [epígrafe 1.10.3]. Como no existe un único proceso que sea apropiado para todos los desarrollos, RUP es un proceso configurable. Se adapta tanto a grupos pequeños de desarrollo como a grandes organizaciones. Basándose en lo que se consideran las mejores prácticas de desarrollo de software, RUP resulta apropiado para una amplia gama de proyectos y organizaciones.

RUP aplica 6 prácticas principales para el desarrollo de sistemas. Estas prácticas son:

- Desarrollo de software en forma iterativa: permite ir creciendo en el entendimiento del problema a través de refinamientos sucesivos. Esto también permite introducir cambios tácticos en los requerimientos, características del sistema o en los tiempos.
- Gestión de requerimientos: es necesario para garantizar que al final el software tenga la calidad requerida.
- Uso de arquitecturas basadas en componentes: el uso de una arquitectura basada en componentes hace que el sistema pueda reutilizar componentes desarrollados con antelación y facilitar el trabajo que se realiza.
- Modelización visual del software: El proceso le demuestra cómo modelar visualmente software para capturar la estructura y el comportamiento de arquitecturas y de componentes.
- Verificación de calidad del software: RUP le asiste en el planeamiento, el diseño, la puesta en marcha, la ejecución, y la evaluación de las pruebas de confiabilidad, funcionalidad, performance de la aplicación y el sistema
- Control de cambios: La capacidad de manejar los cambios - asegurándose que cada cambio sea aceptable, y pudiendo continuar con los mismos- es esencial en un ambiente en el cual el cambio es inevitable. (JHONATAN MINA, 2008)

1.4.2 Lenguajes de Modelado

Con el transcurso de los años se han ido desarrollando las técnicas de programación orientada a objetos (POO). Además se han desarrollado diferentes lenguajes de modelación OO. Algunos ejemplos de estos lenguajes de modelado son i*[epígrafe 1.10.1] y UML [epígrafe 1.10.3].

1.4.2.1 Lenguaje de Modelado OO i*

La notación i* fue creada a en la primera mitad de la década de los 90. Permite expresar de forma clara y sencilla los objetivos de los actores que aparecen en los modelos y la dependencia entre ellos. Consta con una notación gráfica que permite tener una visión intuitiva y unificada del entorno modelado mostrando tales actores y dependencias.

Tiene la desventaja de no tener una definición única del lenguaje. Además, las definiciones existentes no son tan claras como se desearía ya que contienen ambigüedades, contradicciones e incompletitudes.

1.4.2.2 Lenguaje de Modelado OO UML

UML es conjunto de herramientas que permiten modelar (analizar y diseñar) sistemas Orientados a Objeto.

Las herramientas que provee UML para el desarrollo de sistemas OO son:

- Diagrama de casos de uso
- Diagrama de clases
- Diagrama de estados
- Diagrama de secuencias
- Diagrama de actividades
- Diagrama de colaboraciones
- Diagrama de componentes
- Diagrama de distribución

(Torres, 2005)

Características de UML

- Permite adaptarse fácilmente a los usuarios, así como a otros usuarios de otros métodos.
- Permite modelar nuevas cosas en el proyecto que se desarrolla.

- UML provee una expresividad e integridad holística mejorada, respecto a otros lenguajes de moldeamiento visual.
- UML es fácil de aprender y usar, ya sea respecto a las técnicas más avanzadas, es decir, estereotipos y propiedades, así como algunos cambios en la anotación y semánticas.
- UML será la elección obvia para realizar nuevos proyectos, especialmente cuando se incremente la disponibilidad de herramientas, libros y cursos.

1.4.2.3 ¿Por qué UML?

Debido al estudio de las características de los lenguajes de modelado especificadas anteriormente se determino como el lenguaje de modelado que será utilizado para la construcción del SGA es UML ya que sus características se corresponde con lo que se quiere lograr del sistema y a además es el lenguaje de modelado que es usado en la UCI.

1.4.3 Framework de Desarrollo

1.4.3.1 Symfony.

Symfony es un completo framework diseñado para optimizar, gracias a sus características, el desarrollo de las aplicaciones web. Para empezar, separa la lógica de negocio, la lógica de servidor y la presentación de la aplicación web. Proporciona varias herramientas y clases encaminadas a reducir el tiempo de desarrollo de una aplicación web compleja. Además, automatiza las tareas más comunes, permitiendo al desarrollador dedicarse por completo a los aspectos específicos de cada aplicación. El resultado de todas estas ventajas es que no se debe reinventar la rueda cada vez que se crea una nueva aplicación web.

Características

Symfony se diseñó para que se ajustara a los siguientes requisitos:

- Fácil de instalar y configurar en la mayoría de plataformas (y con la garantía de que funciona correctamente en los sistemas Windows y Unix estándares)
- Independiente del sistema gestor de bases de datos
- 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 solo debe configurar aquello que no es convencional
- Sigue la mayoría de mejores prácticas y patrones de diseño para la web
- Preparado para aplicaciones empresariales y adaptable a las políticas y arquitecturas propias de cada empresa, además de ser lo suficientemente estable como para desarrollar aplicaciones a largo plazo
- Código fácil de leer que incluye comentarios de phpDocumentor y que permite un mantenimiento muy sencillo
- Fácil de extender, lo que permite su integración con librerías desarrolladas por terceros

Por otro lado se tiene aplicación web a aquellas aplicaciones que los usuarios pueden utilizar accediendo a un servidor web a través de Internet o de una intranet mediante un navegador. En otras palabras, es una aplicación software que se codifica en un lenguaje soportado por los navegadores web (HTML, JavaScript, Java, etc.) en la que se confía la ejecución al navegador.

Las aplicaciones web son populares debido a lo práctico del navegador web como cliente ligero, así como a la facilidad para actualizar y mantener aplicaciones web sin distribuir e instalar software a miles de usuarios potenciales.

En los últimos años ha aumentado con el uso de internet y los sitios web el uso de frameworks de desarrollo que hacen que se humanice el trabajo que realizan diariamente los programadores de sistemas web complejos. En la actualidad existe una gran gama de estos y cada cual tiene características en específico.

Algunos de estos frameworks de desarrollo son muy populares en nuestros días, tales como Symfony, CakePHP, CodeIgniter, Kumbia entre muchos otros.

Para el desarrollo de SGA se ha utilizado como framework de desarrollo Symfony. Después de hacer una minuciosa y exhaustiva investigación. Los resultados de la misma y el por qué se seleccionó Symfony para el desarrollo se muestra en la siguiente tabla.

Característica	Kumbia	Symfony	Cake
MVC Nativo	X	X	X
Enrutamiento Avanzado	X	X	X
Scaffold (Generadores de código)	X	X	X

Sistema para la Secretaria Docente del Centro Politécnico del Petróleo

Scaffold Avanzado (Generadores de Formularios)	X		
Correo Electrónico	X	X	X
Mapeo Objeto-Relacional	X	X	X
Asociaciones ORM	X	X	X
Eventos ORM	X	X	X
Sistema de Plantillas	X	X	X
Integración con Smarty	X	X	X
Generación de Reportes (Múltiples Formatos)	X		
HTML Helpers	X	X	X
Plug-Ins (Integración Terceros)	X	X	X
i18n (Internacionalización)		X	X
Efectos Visuales	X	X	X
Integración AJAX	X	X	X
Componente de Sesiones	X	X	X
Loggers	X	X	
Documentación en Inglés		X	X
Documentación en Español	X	X	
ACL (Access Control Lists)	X	X	X
Soporte MySQL	X	X	X
Soporte PostgreSQL	X	X	X
Soporte Oracle	X	X	
Soporte SQL Server		X	
Soporte SQLite		X	X

Esto puede demostrar la superioridad que tiene Symfony con los demás framework de desarrollo de PHP que han sido objetos de la investigación.

1.4.4 Lenguaje de Programación.

1.4.4.1 ASP

El ASP es una tecnología dinámica funcionando del lado del servidor, lo que significa que cuando el usuario solicita un documento ASP, las instrucciones de programación dentro del script son ejecutadas para enviar al navegador únicamente el código HTML resultante. La ventaja principal de las tecnologías dependientes del servidor radica en la seguridad que tiene el programador sobre su código, ya que éste se encuentra únicamente en los archivos del servidor que al ser solicitado a través del web, es ejecutado, por lo que los usuario no tienen acceso más que a la página resultante en su navegador.

1.4.4.2 PHP

PHP es un lenguaje de script interpretado en el lado del servidor utilizado para la generación de páginas Web dinámicas, similar al ASP de Microsoft o el JSP de Sun, embebido en páginas HTML y ejecutado en el servidor.

Algunas de sus principales características son:

- Es un lenguaje libre.
- Soporta gran cantidad de bases de datos: MySQL, PostgreSQL, Oracle, entre otras.
- Ofrece una solución simple y universal para las paginaciones dinámicas del Web de fácil programación.
- Perceptiblemente más fácil de mantener y poner al día que el código desarrollado en otros lenguajes.
- Soportado por una gran comunidad de desarrolladores, permitiendo que los fallos de funcionamiento se encuentren y reparen rápidamente.
- Permite el procesamiento de información en formularios, foros de discusión, manipulación de cookies y páginas dinámicas.
- Multiplataforma (Windows, Unix, Linux, Mac, etc.).
- Perfecta integración del Apache-PHP-MySQL, Oracle, MS-SQL, PostgreSQL.
- Bastante sencillo de aprender y utilizar.
- Interactúa con muchas librerías como: manejo de archivos PDF, Flash, calendarios, XML, IMAP, POP, gráficos en general. (Hinostraza, 2007)

1.4.4.3 ¿Por qué PHP?

Para el desarrollo de la aplicación fue seleccionado el lenguaje PHP por sus disímiles ventajas sobre ASP y por ser el lenguaje en el cual se encuentra implementado el framework de desarrollo que se utilizó.

1.4.5 Herramienta Case.

Las herramientas CASE (Computer Aided/Assisted Software/System Engineering) o (Ingeniería de Software Asistida por Computadora) son un conjunto de herramientas que brindan soporte a un enfoque de ingeniería en el desarrollo de software en alguna o en todas las fases de este proceso. (Kendall, 2005)

Tienen como objetivo:

- Permitir la aplicación práctica de metodologías estructuradas.
- Facilitar la realización de prototipos y el desarrollo conjunto de aplicaciones.
- Simplificar el mantenimiento de los programas.
- Mejorar y estandarizar la documentación.
- Aumentar la portabilidad de las aplicaciones.
- Facilitar la reutilización de componentes del software.
- Permitir un desarrollo y un refinamiento visual de las aplicaciones.

Existen varias herramientas de modelado UML que posibilitan el modelado visual de proyectos de software, viabilizando la comunicación entre los integrantes del equipo de desarrollo. Algunos ejemplos son Rational Rose y Visual Paradigm.

1.4.5.1 Rational Rose.

El Rational Rose es una herramienta de modelado visual con plataforma independiente, que ayuda a la comunicación entre los miembros del equipo. Tiene como ventaja que utiliza la notación estándar en la arquitectura de software (UML), la cual permite a los arquitectos y desarrolladores de software visualizar todo el sistema mediante el uso de un lenguaje común. Mantiene la consistencia de los modelos del sistema de software. Permite el chequeo de la sintaxis UML y realizar ingeniería inversa. Posibilita la generación de documentación automáticamente y la generación de código a partir de los modelos. A través de él los diseñadores pueden modelar sus componentes e interfaces en forma individual y luego unirlos con los demás componentes del proyecto.

1.4.5.2 Visual Paradigm para UML

Visual Paradigm para UML es una herramienta CASE, que utiliza UML 2.1 como lenguaje de modelaje y soporta el ciclo de vida completo del desarrollo de software. Es multiplataforma y posibilita la generación de código y documentación. Permite dibujar 13 tipos de diagramas diferentes a través de un intuitivo modelado visual y la aplicación de ingeniería inversa. Admite la importación y exportación de XML e imágenes, la administración de requerimientos, la creación de esquemas de clases a partir de una base de datos y viceversa, además cuenta con un soporte que facilita el trabajo simultáneo sobre un mismo diagrama entre dos desarrolladores en un tiempo real.

1.4.5.3 ¿Por qué Visual Paradigm?

De las herramientas case que anteriormente se mencionan fue seleccionado para el modelado de la ingeniería de Software el Visual Paradigm. Por las características antes mencionadas.

1.4.6 Gestor de Base de Datos.

1.4.6.1 MYSQL

MySQL es un sistema de gestión de bases de datos relacional, licenciado bajo la GPL de la GNU. Su diseño le permite soportar una gran carga de forma muy eficiente. MySQL fue creado por la empresa sueca MySQL AB, que mantiene el copyright del código fuente del servidor SQL, así como también de la marca.

Las principales características de este gestor de bases de datos son las siguientes:

1. Aprovecha la potencia de sistemas multiprocesador.
2. Soporta gran cantidad de tipos de datos para las columnas.
3. Dispone de API's en gran cantidad de lenguajes (C, C++, Java, PHP, etc).
4. Gran portabilidad entre sistemas.
5. Soporta hasta 32 índices por tabla.
6. Gestión de usuarios y contraseñas, manteniendo un muy buen nivel de seguridad en los datos.(Pecos, 2002)

1.4.6.2 PostgresSQL

PostgresSQL es un sistema gestor de Base de datos (BD) estable que tiene como principales características que soporta subconsultas, procedimientos almacenados, triggers y opciones de . Cuenta

con una abundante documentación y se distribuye bajo la licencia BSD, que permite, básicamente cualquier uso del código, siempre y cuando los créditos se mantengan.

A continuación se enumeran las principales características de este gestor de bases de datos:

1. Implementación del estándar SQL92/SQL99.
2. Soporta datos de tipo fecha, monetarios, elementos gráficos, datos sobre redes (MAC, IP...), cadenas de bits, etc. También permite la creación de tipos propios.
3. Incorpora la estructura de datos arreglo.
4. Permite la declaración de funciones propias, así como la definición de disparadores.
5. Soporta el uso de índices, reglas y vistas.
6. Incluye herencia entre tablas.
7. Permite la gestión de diferentes usuarios, como también los permisos asignados a cada uno de ellos. (Quiñones, 2006)

1.7.6.3 ¿Por qué PostgreSQL?

Por sus características y ventajas se seleccionó para utilizar como gestor de Base de Datos PostgreSQL.

Conclusiones

En este capítulo, se describieron los métodos científicos utilizados para la realización de la presente investigación, se trató el estado actual de las aplicaciones de gestión académica en el mundo, en Cuba y en la UCI. Se describieron las principales características del sistema a implementar y se realizó un análisis completo de las tecnologías y herramientas que serán utilizadas en el desarrollo del sistema propuesto, se describieron los lenguajes de programación y modelado, la arquitectura, el sistema gestor de bases de datos, la herramienta case y la metodología a utilizar. SGA es desarrollada en un ámbito de aplicación web haciendo uso de la tecnología PHP, el framework de desarrollo Symfony para su elaboración y Postgres SQL como gestor de Base de datos.

Capítulo 2: Características del Sistema

Introducción

En este capítulo se abordarán las principales características que presenta el sistema que se desarrolla así como el objetivo de automatización y las informaciones principales que se manejan en el ámbito de la Secretaría Docente del CPP. En el mismo se mostrará cómo se pensó el desarrollo de los principales elementos que conforman el sistema.

2.1 Objeto de automatización.

En el CPP se llevan a cabo seis procesos fundamentales para el proceso de la gestión académica en la Secretaría Docente del mismo. Estos procesos son:

- Gestionar Diplomados: En él se comprende la gestión de los diplomados que se imparten en el CPP.
- Control de Evaluaciones del Curso: En este caso de uso se controla el proceso de evaluación de los cursistas a partir de que los mismos se presentan a su evaluación final.
- Control de Evaluaciones de Diplomados: En este caso de uso se controla el proceso de evaluación de los diplomados, desde el momento en que se matriculan en el diplomado hasta el que discuten la tesina.
- Gestionar Matrícula: En este caso de uso se lleva a cabo todo el proceso que tiene que ver con el control de la inscripción de los estudiantes al curso, desde el momento en que CUPET entrega a la Secretaria del CPP el pre listado con los posibles participantes en el curso a impartir según el plan y hasta que la secretaria docente registra todos los datos que se necesita de cada uno de los cursistas.
- Gestionar Cursos: Este caso de uso abarca lo relacionado con la gestión de los cursos que se imparten ya sea inserción, actualización de los datos y eliminación en caso de ser necesario.
- Control de Registro: En este caso de uso se controla el proceso elaboración entrega y recogida del registro de asistencia y calificaciones.

2.2 Información que se Maneja.

Dentro de la Secretaría Docente del CPP se maneja un gran cúmulo de información, esta se estructura en varios reportes que son generados por los trabajadores. Estos reportes son los que se describen a continuación:

- **Certificados:** Se llenan de acuerdo a los graduados de los cursos y se le entregan a estos para tener constancia de que participaron en el curso.
- **Registro de Asistencia y Calificaciones:** Es creado y entregado a los profesores de los diferentes cursos para que se lleve un control de la asistencia de los estudiantes a los cursos y las calificaciones de los mismos.
- **Libro de Matrículas y Graduados:** Se llena uno para cada función respectivamente. En el de Matrículas se adicionan todos los estudiantes que matriculan inicialmente en el curso. Al finalizar el mismo entonces se llenan los datos del Libro de Graduados en el cual se encuentran todos los estudiantes que terminaron el curso con sus respectivas notas.
- **Informe Final:** Por cada curso que se finaliza se crea un informe que contiene los estudiantes que se graduaron del curso y las calificaciones de los mismos con la firma del profesor correspondiente.
- **Acciones Concluidas:** Al concluir un período determinado se debe llenar un registro que contiene toda la información de las diferentes acciones que se han realizado en la Secretaría Docente del CPP.
- **Acta de Evaluación:** Es generada para entregar al tribunal encargado de calificar a los estudiantes que se encuentran matriculados en Diplomados a la hora de exponer el trabajo final de los mismos.
- **Reporte:** Se crean tres reportes fundamentales que son:
 - **Reporte Mensual:** Es creado mensualmente con la relación de los cursos que se llevan a cabo en el CPP en cada uno de los meses.
 - **Reporte Semestral:** Es creado semestralmente y contiene la relación de los cursos que se dieron en un semestre dentro del CPP con respecto a los planificados.
 - **Reporte Anual:** Es creado al final del año y contiene la relación de los cursos que se han impartido en todo un año dentro del CPP con respecto a los que se habían planificado con antelación.
- **Certificados:** El sistema debe permitir además crear e imprimir los certificados que son entregados a personal que pasa el curso.

2.3 Propuesta de Sistema.

El sistema cuenta con tres módulos principales que son:

- Gestión:
 - Gestionar Cursos.
 - Gestionar Postgrados.
 - Gestionar Diplomados.
 - Gestionar Maestrías.
 - Gestionar Especialidades.
 - Gestionar Estudiantes.
- Administración.
 - Gestionar Usuario.
 - Gestionar Roles.
 - Gestionar Permisos.
 - Seguridad y Trazas.
- Reporte.
 - Certificados.
 - Registro de Asistencia y Calificaciones.
 - Libro de Matrículas y Graduados.
 - Informe Final de Cursos.
 - Libro de Graduados.
 - Acciones Concluidas.
 - Acta de Evaluación.
 - Emitir Certificación.
 - Reporte Mensual.
 - Reporte Semestral.
 - Reporte Anual.

Dicho sistema contará con un menú horizontal con submenú desplegable donde se encontrarán cada uno de los módulos de la aplicación. También contará con soporte para exportar cada uno de los diferentes reportes a formatos Word, Excel, PDF e imprimir la información que se genere en los mismos.

El módulo de administración será un subsistema a parte que permitirá sincronizarse con otros sistemas y administrarlos. Logrando así un mayor ensamblaje entre las aplicaciones usadas por los trabajadores de la secretaría del CPP.

La descripción textual de los principales casos de uso que se relacionan se encuentra en el documento de casos de uso del sistema que tienen la descripción de cada uno de estos casos de uso.

2.4 Modelo del Negocio

El modelo del negocio se centra en los límites o fronteras de las organizaciones y el establecimiento de las entidades con quien necesita comunicarse. Se encarga de definir los flujos de trabajo internos de la organización y de encontrar fórmulas para la optimización de estos. Uno de los elementos que define el modelo de negocio son los actores, los cuales son las entidades externas que interactúan con la organización. (Larman, 2004)

En la secretaría del CPP, donde se establece el modelo de negocio del SGA, los actores serían los que se describen en la tabla 1, pues son los principales beneficiados con los diferentes procesos que se realizan dentro del departamento.

Actor	Descripción
Jefe de Secretaría	Es el actor del negocio que guía el proceso que se desarrolla dentro de la Secretaría en el CPP.

Tabla 1: Actor del Negocio

Los roles dentro de las organizaciones, en este caso la secretaría de CPP, son ejecutados por los trabajadores del negocio. Habiendo definido los trabajadores del negocio será posible definir como estos interactúan y que responsabilidades deberán cumplir. La tabla 2 muestra los trabajadores del negocio y su descripción.

Trabajador	Descripción
Secretaria	Es el trabajador del negocio que interactúa directamente con él, y es la encargada de atender todas las peticiones que se hagan en los distintos procesos de negocio.
Profesor	Es el trabajador del negocio que interviene en el proceso de evaluación de los estudiantes.

Estudiante	Es el trabajador del negocio que interviene en el proceso de gestión de matrícula llenando la planilla con los datos del mismo.
Diseño Curricular	En el trabajador del negocio que interviene en el proceso de gestionar curso creando el programa del curso que se va a impartir.
CUPET	Es el trabajador del negocio que interactúa en el proceso de matrícula de los estudiantes.
Tribunal	Es el trabajador del negocio que interviene en el proceso de gestionar evaluaciones de los diplomados.

Tabla 2: Trabajadores de Negocio

2.5 Diagrama de Casos de Uso del Negocio

El diagrama de casos de uso del negocio representará los casos de uso del negocio y los actores del negocio. La figura 3 muestra el diagrama de casos de uso del negocio del SGA.

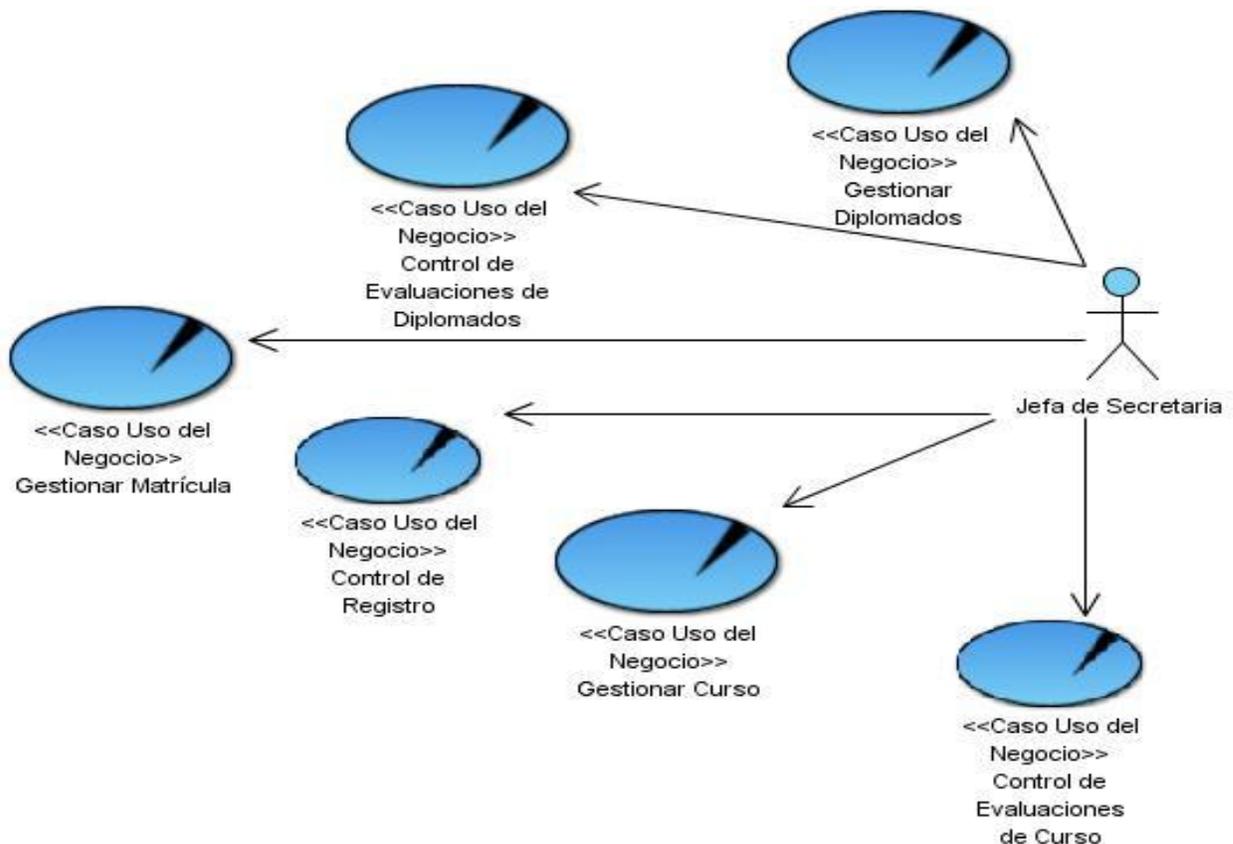


Imagen 3

2.6 Descripción de los Casos de Uso del Negocio

A continuación se realiza una breve descripción de los casos de uso del negocio (CUN). Los diagramas de actividad de los mismos pueden verse en el Anexo #1.

2.6.1 CUN Gestionar Matrícula

En este caso de uso se lleva a cabo todo el proceso que tiene que ver con el control de la inscripción de los estudiantes al curso, desde el momento en que CUPET entrega a la Secretaria del CPP el pre listado con los posibles participantes en el curso a impartir según el plan y hasta que la secretaria docente registra todos los datos que se necesita de cada uno de los cursistas.

Descripción del CUN Gestionar Matrícula

Caso de Uso:	Gestionar Matrícula	
Actores:	Jefa de Secretaria	
Trabajadores:	Secretaria, CUPET, Profesor, Estudiante	
Resumen:	En este caso de uso se lleva a cabo todo el proceso que tiene que ver con el control de la inscripción de los estudiantes al curso, desde el momento en que CUPET entrega a la Secretaria del CPP el pre listado con los posibles participantes en el curso a impartir según el plan y hasta que la secretaria docente registra todos los datos que se necesita de cada uno de los cursistas.	
Precondiciones:		
Flujo Normal de Eventos		
	Acción del Actor	Respuesta del Negocio
	<ol style="list-style-type: none"> 1. El Jefe de Secretaría pide el Listado de Pre matrícula de los estudiantes que participan en el Curso. 3. Recibe el Listado de Pre-Matricula y se lo entrega a la Secretaria. 	<ol style="list-style-type: none"> 2. CUPET crea el Listado de Pre-matricula y se lo entrega al Jefe de Secretaría. 4. La secretaria docente recibe el Listado de Pre-matrícula con los posibles participantes en el curso

5. La secretaria docente crea las Hojas de Matrícula necesarias para la inscripción de los estudiantes y se las entrega al profesor.
6. El profesor entrega a los estudiantes las Hojas de Matrícula para que los mismos llenen sus datos personales
7. Los estudiantes reciben las Hojas de Matrícula.
8. Los estudiantes llenan los datos personales requeridos en la Hoja de Matrícula.
9. Los estudiantes entregan las Hojas de Matrícula al profesor.
 - a. En caso de matricular en un diplomado, maestría o especialidad ver flujo alterno 9.1.
10. El profesor recibe las Hojas de Matrícula llenadas por los estudiantes y las Fotocopias de los Títulos.
11. El profesor entrega a la secretaria docente las Hojas de Matrícula llenadas por los estudiantes y las Fotocopias de los Títulos.
12. La secretaria docente recibe las Hojas de Matrícula y las Fotocopias de los Títulos.
13. La secretaria docente crea un Pre listado con la relación de los estudiantes que se encuentran recibiendo el curso
14. La secretaria docente entrega el Pre listado al profesor para que los estudiantes corrijan sus datos personales.
15. El profesor recibe el Pre listado con los datos

<p>23. Recibe el listado de la matrícula oficial del curso y la archiva.</p>	<p>personales de los estudiantes</p> <p>16. El profesor entrega el Pre listado a los estudiantes para que los mismos rectifiquen sus nombres.</p> <p>17. El estudiante recibe el Pre listado.</p> <p>18. Si no hay errores el estudiante entrega el Pre listado al profesor.</p> <p style="padding-left: 40px;">a. En caso de errores ver flujo alternativo 19.1.</p> <p>19. El profesor recibe el Pre listado corregido.</p> <p>20. El profesor entrega el Pre listado a la secretaria docente.</p> <p>21. La secretaria docente recibe el Pre listado corregido.</p> <p>22. La secretaria docente elabora el Listado Oficial con la matrícula del curso.</p> <p style="padding-left: 40px;">a. En caso de ser para una maestría, diplomado, especialidad o postgrado ver flujo alternativo 23.1.</p> <p>24. Entrega e Listado de la matrícula oficial del curso.</p>
Flujos Alternos	
Acción del Actor	Respuesta del Negocio
	9.1 Si el curso a impartir es un diplomado,

	<p>postgrado, maestría o especialidad Se entrega una fotocopia del título.</p> <p>19.1 El estudiante corrige los errores encontrados en el Pre listado</p> <p>19.2 El estudiante entrega el Pre listado al profesor.</p> <p>19.3 El profesor recibe el Pre listado corregido.</p> <p>19.4 El profesor entrega el Pre listado corregido a la secretaria docente.</p> <p>19.5. La secretaria docente recibe el Pre listado corregido</p> <p>19.6. La secretaria docente chequea las Fotocopias de los Títulos.</p> <p>19.7. La secretaria docente elabora el Listado Oficial con la matrícula del curso.</p> <p>23.1. Si es un diplomado, especialidad, maestría o Postgrado chequea la veracidad de la Fotocopia del Título con el original</p>
Pos condiciones	

Tabla 3

Modelo de Objetos. CUN Gestionar Matricula.

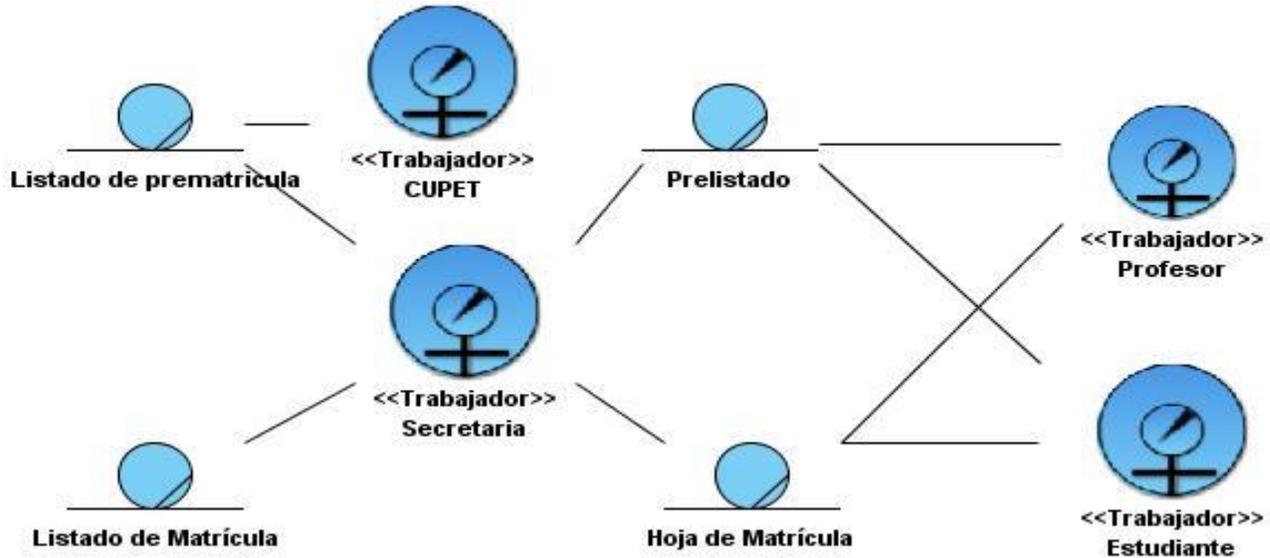


Imagen 2

2.6.2 CUN Gestionar Curso

En este caso de uso se lleva a cabo todo el proceso que tiene que ver con la gestión de los cursos que se imparten en el CPP.

Descripción del CUN Gestionar Curso

Caso de Uso:	Gestionar Cursos
Actores:	Jefe de Secretaría
Trabajadores:	Secretaria, Diseño Curricular
Resumen:	En este caso de uso se lleva a cabo todo el proceso que tiene que ver con la gestión de los cursos que se imparten en el CPP.
Precondiciones:	
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Negocio

<p>1. El caso de uso se inicializa cuando el Jefe de Secretaría pide la información referente al curso.</p> <p>12. Recibe y archiva la información referente al curso.</p>	<p>2. Diseño curricular crea el programa del curso.</p> <p>3. Diseño curricular entrega a la secretaria docente del CPP el programa del curso.</p> <p>4. La secretaria docente recibe el programa del curso elaborado por diseño curricular.</p> <p>5. La secretaria docente crea el expediente del curso a impartir.</p> <p>6. La secretaria docente crea los registros docentes para cada asignatura.</p> <p>7. La secretaria docente crea la Hoja de Presentación del curso.</p> <p>8. La secretaria docente crea los expedientes de los estudiantes registrados en el curso</p> <p>9. La secretaria docente crea las Actas de defensa de los estudiantes.</p> <p>10. La secretaria docente crea el Resumen de Calificaciones.</p> <p>11. La secretaria docente entrega la información del curso.</p>
Flujos Alternos	
Acción del Actor	Respuesta del Negocio
Pos condiciones	

Tabla 4

Modelo de Objetos. CUN Gestionar Curso

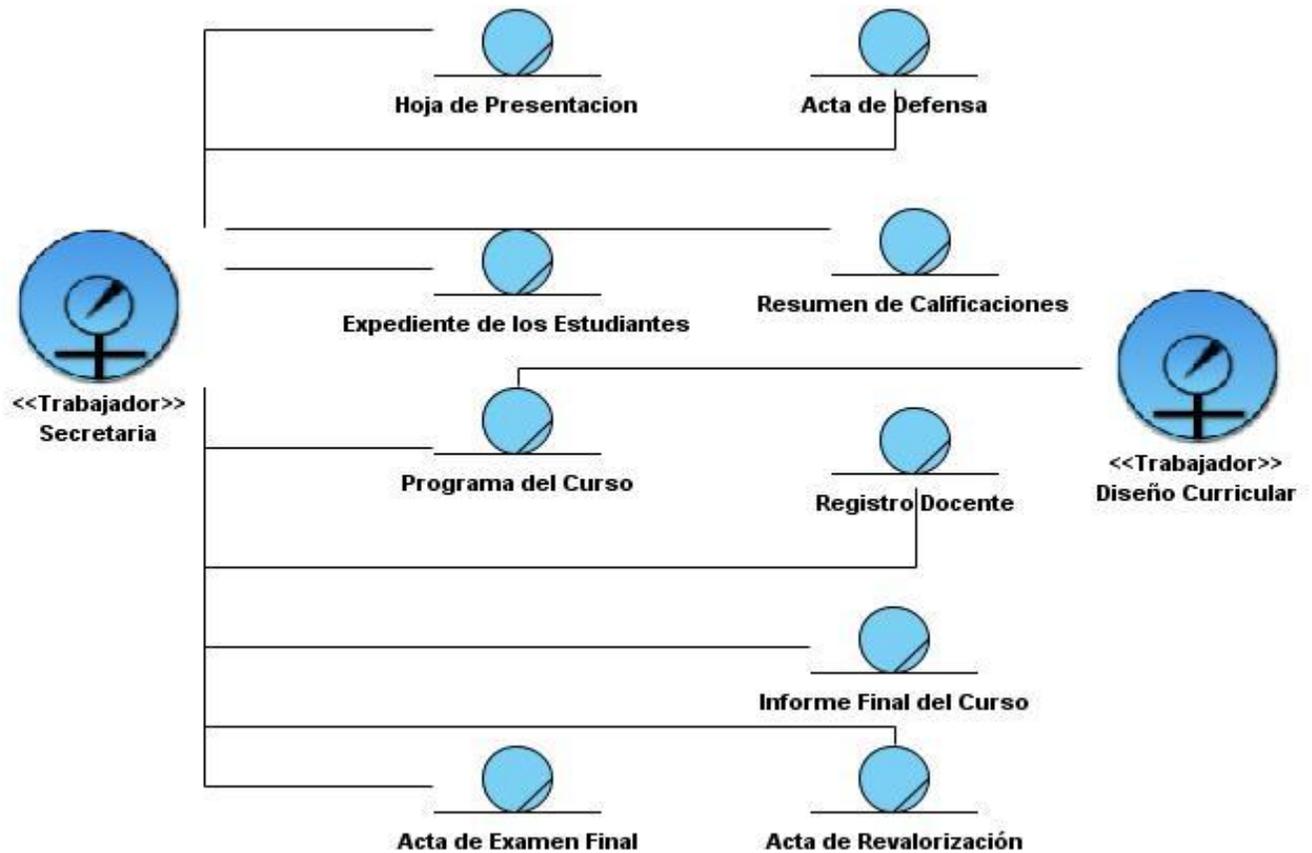


Imagen 3

2.6.3 CUN Control de Evaluaciones de Curso.

Descripción del CUN Control de Evaluaciones de Curso.

Caso de Uso:	Control de Evaluaciones de Curso
Actores:	Profesor, Estudiante
Trabajadores:	Secretaria
Resumen:	En este caso de uso se controla el proceso de evaluación de los cursistas a partir de que los mismos se presentan a su evaluación final.
Precondiciones:	
Flujo Normal de Eventos	

	<p>11.1 La secretaria docente crea el Acta de Revalorización</p> <p>11.2 La secretaria docente entrega al profesor el Acta de Revalorización.</p> <p>11.3. El profesor recibe el Acta elaborada por la secretaria docente</p> <p>11.4. El profesor entrega el Acta al estudiante para su firma.</p> <p>11.5. El estudiante recibe el Acta.</p> <p>11.6. El estudiante firma el Acta.</p> <p>11.7. El estudiante entrega el Acta firmada al profesor</p> <p>11.8. El profesor recibe el Acta firmada por el estudiante.</p> <p>11.9. El profesor entrega el Acta firmada a la secretaria docente.</p> <p>11.10La secretaria docente recibe el Acta firmada por el estudiante</p>
Poscondiciones	

Tabla 5

Modelo de Objetos. CUN Control de Evaluaciones de Curso

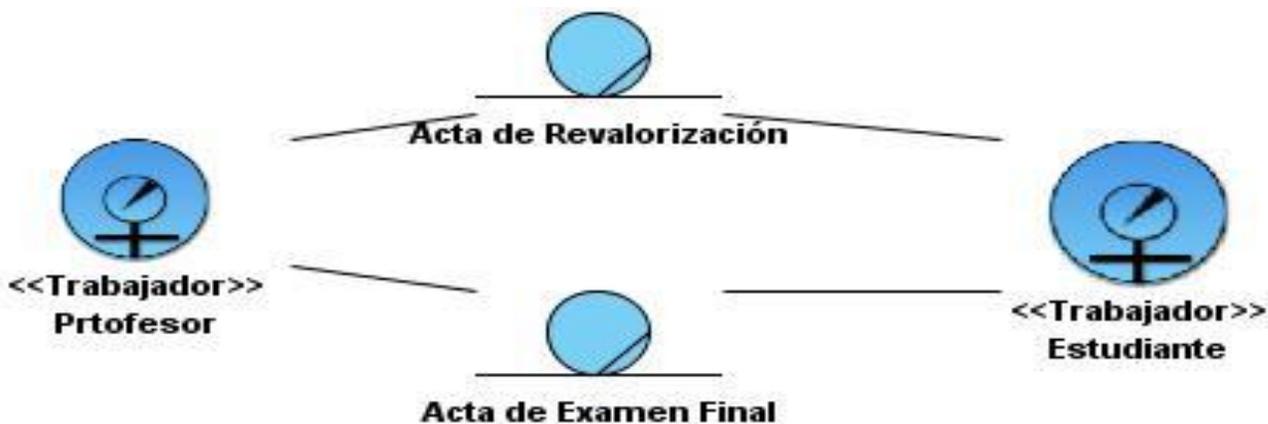


Imagen 4

2.6.4 CUN Control de Evaluaciones de Diplomado

Descripción del CUN Control de Evaluaciones de Diplomado

Caso de Uso:	Control de Evaluaciones de Diplomados
Actores:	Jefe de Secretaría
Trabajadores:	Secretaria, Tribunal, Estudiante
Resumen:	En este caso de uso se controla el proceso de evaluación de los diplomados
Precondiciones:	
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Negocio
<ol style="list-style-type: none"> 1. El caso de uso se inicia cuando el Jefe de Secretaría pide los datos necesarios para el control del curso. 	<ol style="list-style-type: none"> 2. La secretaria docente crea el resumen de Calificaciones o control Horizontal. 3. La secretaria docente crea el Acta de Defensa. 4. La secretaria docente entrega el Acta de Defensa al

	<p>tribunal.</p> <ol style="list-style-type: none"> 5. El tribunal recibe el Acta de Defensa elaborada por la secretaria docente. 6. El tribunal modifica el Acta de Defensa. 7. El tribunal entrega el Acta de Defensa a la secretaria docente con el criterio o resumen del oponente y del tutor. 8. La secretaria docente recibe el Acta de Defensa modificada por el tribunal. 9. La secretaria docente archiva el Acta de Defensa 10. La secretaria docente crea la Certificación de Calificaciones y Créditos. 11. La secretaria docente crea el Certificado de Graduado. 12. La secretaria docente entrega el Certificado de Graduado y la Certificación de Calificaciones y Créditos al estudiante. 13. El estudiante recibe el Certificado de Graduado y la Certificación de Calificaciones y Créditos.
Flujos Alternos	
Acción del Actor	Respuesta del Negocio
Poscondiciones	

Tabla 6

Modelo de Objetos. CUN Control de Evaluaciones de Diplomado.

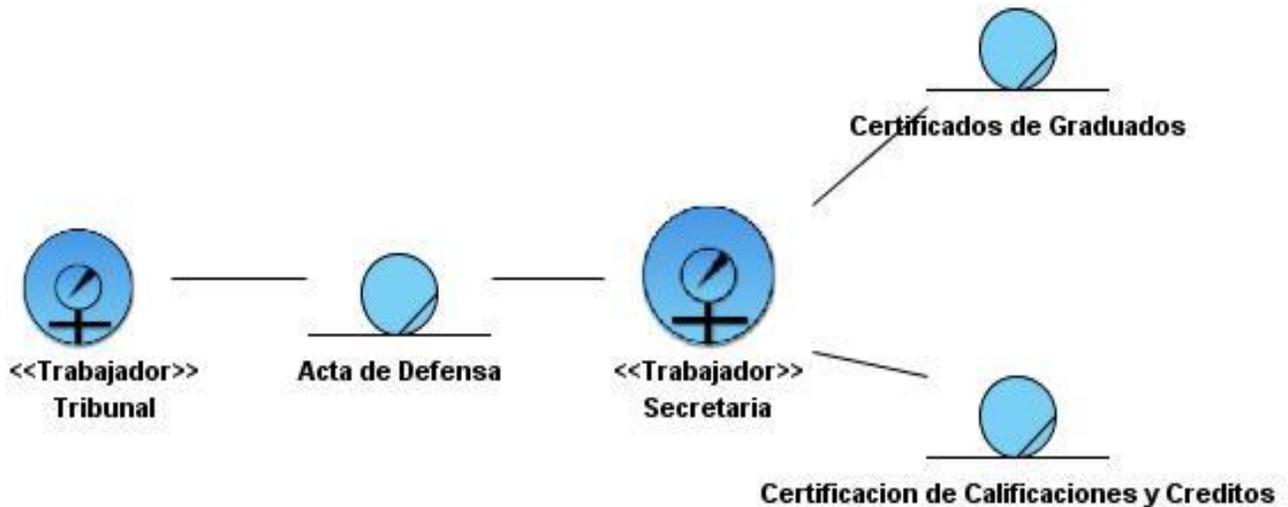


Imagen 5

2.6.5 CUN Control de Registro.

Descripción del CUN Control de Registro.

Caso de Uso:	Control de Registro	
Actores:	Jefe de Secretaría	
Trabajadores:	Secretaria, Profesor	
Resumen:	En este caso de uso se controla el proceso elaboración entrega y recogida del registro de asistencia y calificaciones	
Precondiciones:		
Flujo Normal de Eventos		
	Acción del Actor	Respuesta del Negocio
	<p>1. El caso de uso se inicializa cuando el Jefe de Secretaria pide los documentos necesarios para el control de registro.</p>	<p>2. La secretaria docente confecciona el control de acciones de capacitación desarrolladas.</p>

<p>12. Recibe y archiva los documentos.</p>	<ol style="list-style-type: none"> 3. La secretaria docente crea el Registro de Asistencia y Calificaciones y lo entrega al profesor. 4. El profesor recibe el Registro de Asistencia y Calificaciones elaborado por la secretaria docente. 5. El profesor llena el Registro de Asistencia y Calificaciones. 6. El profesor entrega a la secretaria docente el Registro de Asistencia y Calificaciones. 7. La secretaria docente recibe el Registro de Asistencia y Calificaciones modificado por el profesor. 8. La secretaria docente archiva el Registro de Asistencia y Calificaciones. 9. La secretaria docente crea registro de graduado. 10. La secretaria docente crea diplomas de los graduados. 11. La secretaria docente entrega los documentos al Jefe de Secretaría.
Flujos Alternos	
Acción del Actor	Respuesta del Negocio
Poscondiciones	

Tabla 7

Modelo de Objetos. CUN Control de Registro.



Imagen 6

2.6.6 CUN Gestionar Diplomado.

Descripción del CUN Gestionar Diplomado

Caso de Uso:	Gestionar Diplomado	
Actores:	Jefe de Secretaría	
Trabajadores:	Secretaria, Diseño Curricular	
Resumen:	En este caso de uso se controla el proceso de gestión de los diplomado	
Precondiciones:		
Flujo Normal de Eventos		
	Acción del Actor	Respuesta del Negocio
	<ol style="list-style-type: none"> 1. El caso de uso se inicializa cuando el Jefe de Secretaría pide los datos para registrar el diplomado. 3. Recibe el programa del diplomado. 	<ol style="list-style-type: none"> 2. Diseño curricular entrega el programa del diplomado al Jefe de Secretaría docente.

<p>4. Revisa la cantidad de cursos a impartir.</p> <p>5. Pide que se registren todos los cursos del diplomado.</p> <p>9. Recibe y archiva la documentación del diplomado.</p>	<p>6. La secretaria docente registra todos los cursos a impartir en el diplomado.</p> <p>7. La secretaria docente registra el diplomado.</p> <p>8. La secretaria docente entrega los documentos.</p>
Flujos Alternos	
Acción del Actor	Respuesta del Negocio
Poscondiciones	Diplomado registrado

Tabla 8

Modelo de Objetos. CUN Gestionar Diplomado.

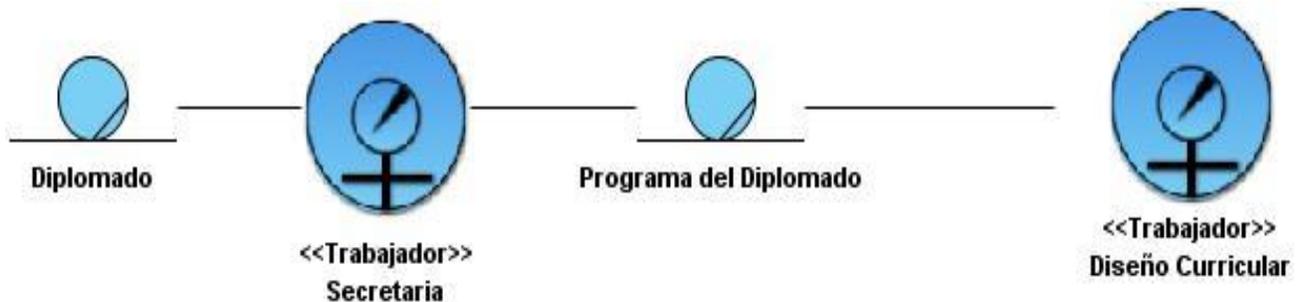


Imagen 7

2.7 Especificación de los Requisitos de Software

Con la especificación de los requisitos de software se obtiene una descripción detallada de las necesidades de un producto informático. Estos se dividen en dos grupos para una mejor especificación, los requisitos funcionales y los no funcionales. Los funcionales definen las capacidades que el sistema debe cumplir, o el qué debe hacer el sistema, mientras que los no funcionales son las propiedades que el mismo presenta, o el cómo debe hacerse. (Ivar Jacobson, 1999)

2.7.1 Requerimientos Funcionales

A continuación se muestran los requisitos funcionales de la aplicación SGA en su versión 1.0.

Alias	Requerimiento	Descripción
RF 1	Gestionar Matricula	Gestiona los datos que se almacenan de los estudiantes.
RF 1.1	Insertar datos del Estudiante.	Inserta los datos de un nuevo estudiante.
RF 1.2	Modificar datos del Estudiante.	Modifica los datos de un estudiante que ya se encuentra.
RF 1.3	Eliminar datos del Estudiante.	Elimina un estudiante determinado.
RF 1.4	Seleccionar datos del Estudiante.	Devuelve un listado con la información de todos los estudiantes.
RF 2.0	Gestionar Cursos.	Gestiona los datos que se almacenan de los cursos que se imparten en el CPP.
RF 2.1	Insertar datos del Curso	Inserta los datos de un nuevo curso.
RF 2.2	Modificar datos del Curso.	Modifica los datos de un curso determinado.
RF 2.3	Eliminar datos del Curso.	Elimina un curso determinado.
RF 2.4	Seleccionar datos del Curso.	Devuelve un listado con la información de todos los cursos.
RF 3.0	Emitir Certificados.	Crea los certificados que se les entregan a los estudiantes.
RF 4.0	Emitir Reporte.	Genera los reportes que se deben entregar por parte de las secretarias a la dirección de CPP.
RF 4.1	Emitir cierre mensual.	
RF 4.2	Emitir cierre trimestral.	
RF 4.3	Emitir cierre semestral.	
RF 4.4	Emitir cierre anual.	
RF 5.0	Emitir Registro de Asistencia y calificaciones.	Genera el registro de asistencia y calificaciones que utilizan los profesores para controlar las evaluaciones de los estudiantes.
RF 6.0	Emitir Informe Final.	Genera el informa final que se elabora al culminar cualquier curso en el CPP.
RF 7.0	Mostrar Acciones Concluidas.	Muestra un informe con todos los cursos que se han impartido en el CPP en un tiempo determinado.

Sistema para la Secretaria Docente del Centro Politécnico del Petróleo

RF 8.0	Mostrar Libro de Matrículas y Graduados.	Genera el libro de Matrículas y Graduados que contiene la información de todos los cursistas que empiezan en un curso y los que terminan el mismo.
RF 9.0	Mostrar Libro de graduados.	Genera el Libro de Graduados que contiene información de aquellos estudiantes que se graduaron.
RF 10.0	Emitir Acta de Evaluación.	Genera el acta de evaluación. Documento que se necesita para la evaluación de aquellos estudiantes que participan en diplomados, maestrías y especialidades.
RF 11.0	Emitir Calificaciones de Diplomados.	Emite un reporte con todas las calificaciones de los estudiantes que están pasando un determinado diplomado.
RF 12.0	Gestionar Diplomados.	Gestiona la información que se almacena de los diplomados.
RF 12.1	Insertar Diplomados.	Inserta los datos de los diplomados.
RF 12.1.1	Insertar Postgrados.	Inserta o Selecciona los postgrados que se impartirán en ese diplomado.
RF 12.2	Modificar Diplomados.	Modifica la información contenida en el diplomado.
RF 12.2.1	Modificar Postgrados.	Modifica los postgrados que se imparten en el diplomado.
RF 12.3	Eliminar Diplomados.	Elimina el diplomado.
RF 12.3.1	Eliminar Postgrados	Elimina los postgrados que se imparten en el diplomado.
RF 12.4	Seleccionar Diplomados.	Devuelve un listado con los diplomados que hay.
RF 12.4.1	Seleccionar Postgrados.	Devuelve un listado de los postgrados que tiene un diplomado determinado.
RF 13.0	Gestionar Maestrías.	Gestiona la información que se almacena de las Maestrías.
RF 13.1	Insertar Maestría.	
RF 13.1.1	Insertar Postgrados.	
RF 13.2	Modificar Maestría.	
RF 13.2.1	Modificar Postgrados.	
RF 13.3	Eliminar Maestría.	
RF 13.3.1	Eliminar Postgrados.	

RF 13.4	Seleccionar Maestría.	
RF 13.4.1	Seleccionar Postgrados.	
RF 14.0	Gestionar Especialidades.	Gestiona la información que se almacena de las Especialidades.
RF 14.1	Insertar Especialidad.	
RF 14.1.1	Insertar Postgrados.	
RF 14.2	Modificar Especialidad.	
RF 14.2.1	Modificar Postgrados.	
RF 14.3	Eliminar Especialidad.	
RF 14.3.1	Eliminar Postgrados.	
RF 14.4	Seleccionar Especialidad.	
RF 14.4.1	Seleccionar Postgrados.	

Tabla 9

2.7.2 Requerimientos No Funcionales

Categoría	Requerimiento	Descripción
Usabilidad	Fácil empleo para usuarios sin experiencia	El sistema debe tener una interfaz que le sea familiar al usuario para aprovechar sus conocimientos en el manejo de herramientas de software. Además debe ser de fácil aprendizaje para que usuarios inexpertos puedan familiarizarse rápidamente.
Usabilidad	Emplear perfiles de usuario	Diferenciar las interfaces gráficas y opciones para los usuarios que accedan al sistema con diferentes roles (secretaria, especialista, etc.).
Seguridad	Protección de la Base de Datos	La base de datos debe tener varios esquemas, dividiendo así de una forma lógica las funcionalidades, evitando la pérdida total de la información en caso de algún accidente o ataque.
Seguridad	Reglas del Negocio sobre la Aplicación	Deben estar definidas reglas del negocio para distintos tipos de flujos alternativos y lograr una mayor flexibilidad y

		robustez en el sistema.
Rendimiento	Escalabilidad	El sistema debe ser capaz de mantener el mismo rendimiento y estabilidad a medida que aumenta la cantidad de datos a gestionar.
Soporte	Servidor Apache en los sistemas operativos Linux y Windows	El sistema debe descansar sobre un Servidor Apache y no sobre un Servidor IIS ya que el sistema debe ser multiplataforma, donde el IIS no brinda esta funcionalidad y el Apache sí.
Restricciones de Diseño	Gestor de bases de datos PostgreSQL 8.3	El sistema debe descansar sobre el gestor de bases de datos PostgreSQL 8.3 ya que es un gestor potente, seguro y estable que pertenece al movimiento de software libre. El mismo soporta varios lenguajes de consulta como PL/pgSQL y SQL.
Restricciones de Diseño	Compatibilidad con los diferentes navegadores	El sistema debe ser capaz de poderse ejecutar desde diferentes navegadores como Mozilla Firefox e Internet Explorer.

Tabla 10

2.8 Definición de los Actores del Sistema

Actor	Descripción
Administrador	Es el actor que se encarga de la administración de todos los permisos dentro de la aplicación.
Secretaria	Es el actor que se encarga de gestionar la información que se maneja en el CPP de los diferentes cursos, especialidades, maestrías, postgrados y diplomados que se imparten.
Director	Es el actor que solamente puede ver los reportes que son pedidos a la Secretaria del CPP.
Subdirector Docente	Es el actor que solamente puede ver los reportes que son pedidos a la Secretaria del CPP.

Tabla 11

2.9 Diagrama de Casos de Uso del Sistema

El sistema se encuentra dividido por su envergadura en tres módulos principales los cuales se muestran en la imagen que viene a continuación.

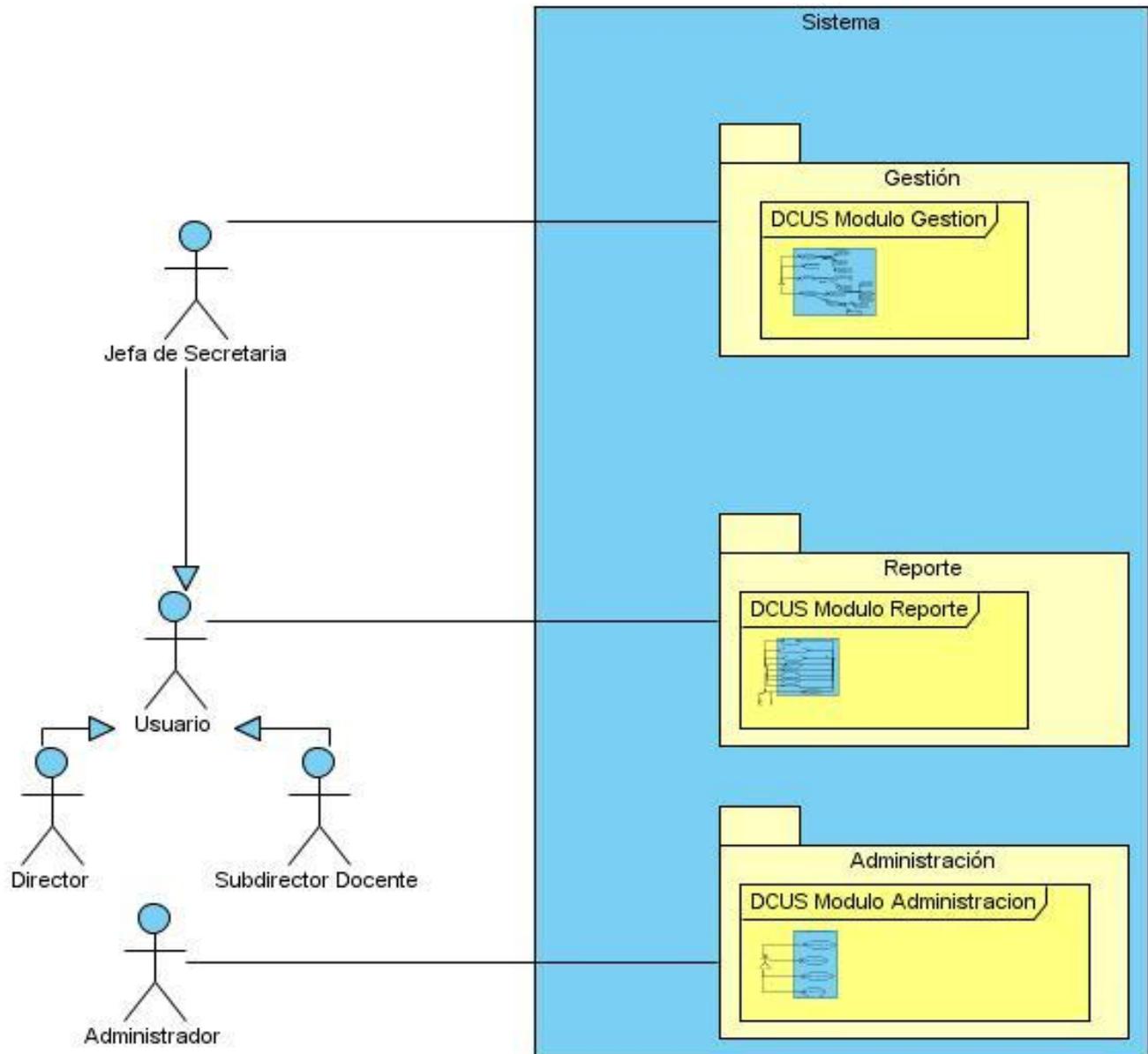


Imagen 8

En cada uno de estos módulos hay una determinada cantidad de funcionalidades que se muestran en las siguientes figuras separadas por subsistemas.

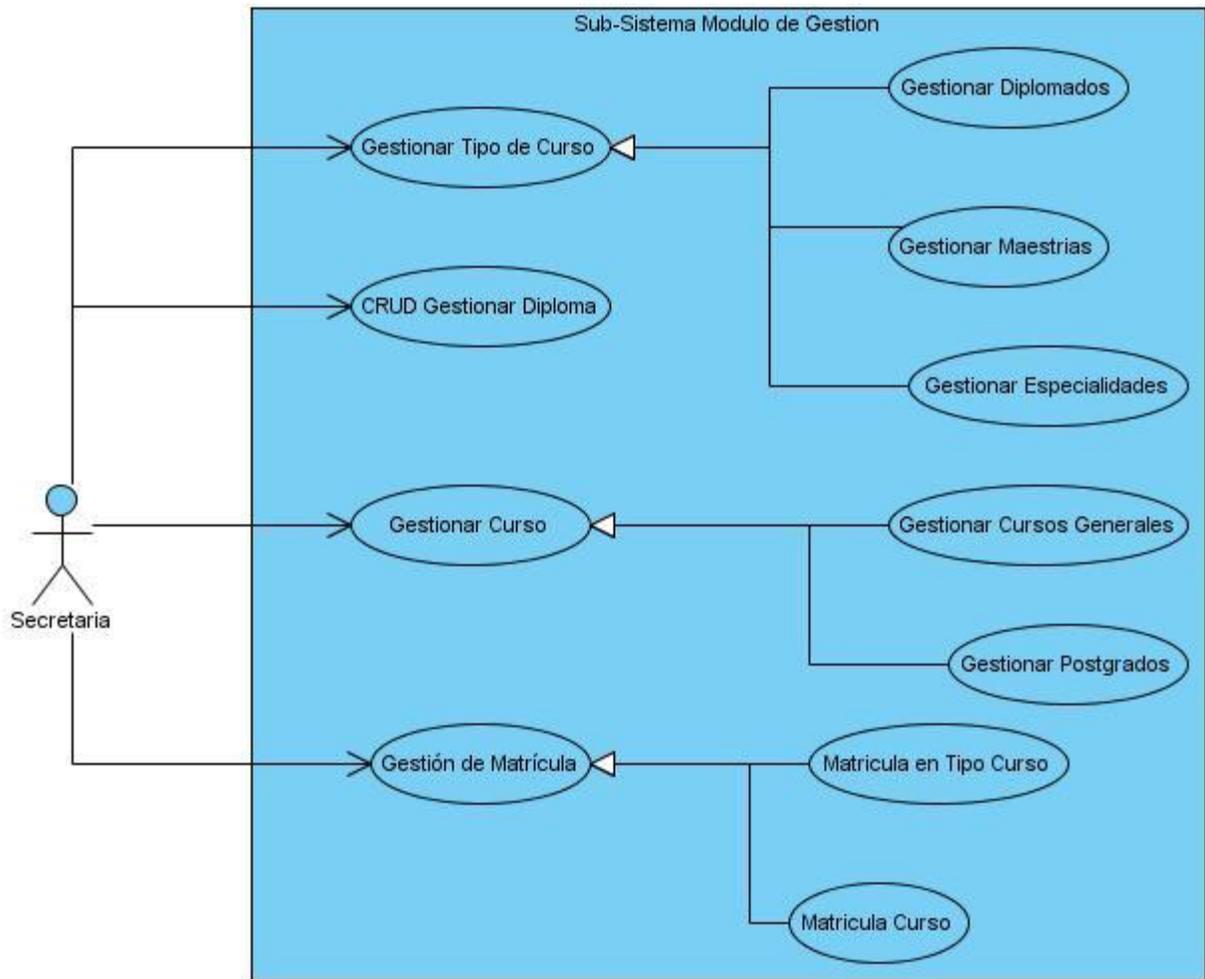


Imagen 9: Módulo de Gestión

En el Subsistema del modulo de gestión se agrupan los casos de uso principales que le dan soporte a los casos de uso críticos dentro del contenido del sistema que se desarrolla.

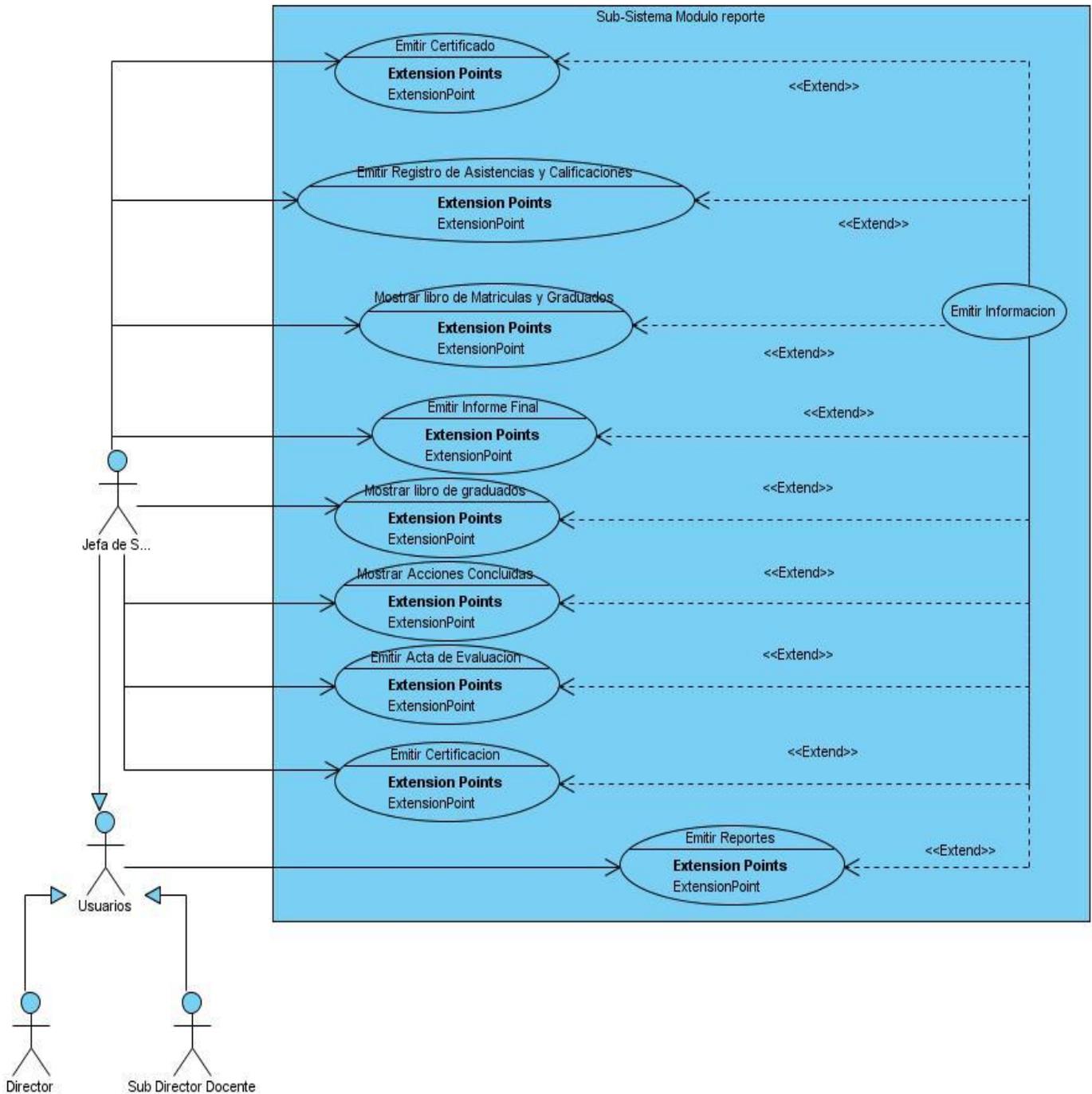


Imagen 10: Módulo de Reporte

En el modulo reporte se encuentran toda una gama de reportes que se generan con el objetivo de evaluar el trabajo que realizan las secretarias en el CPP.

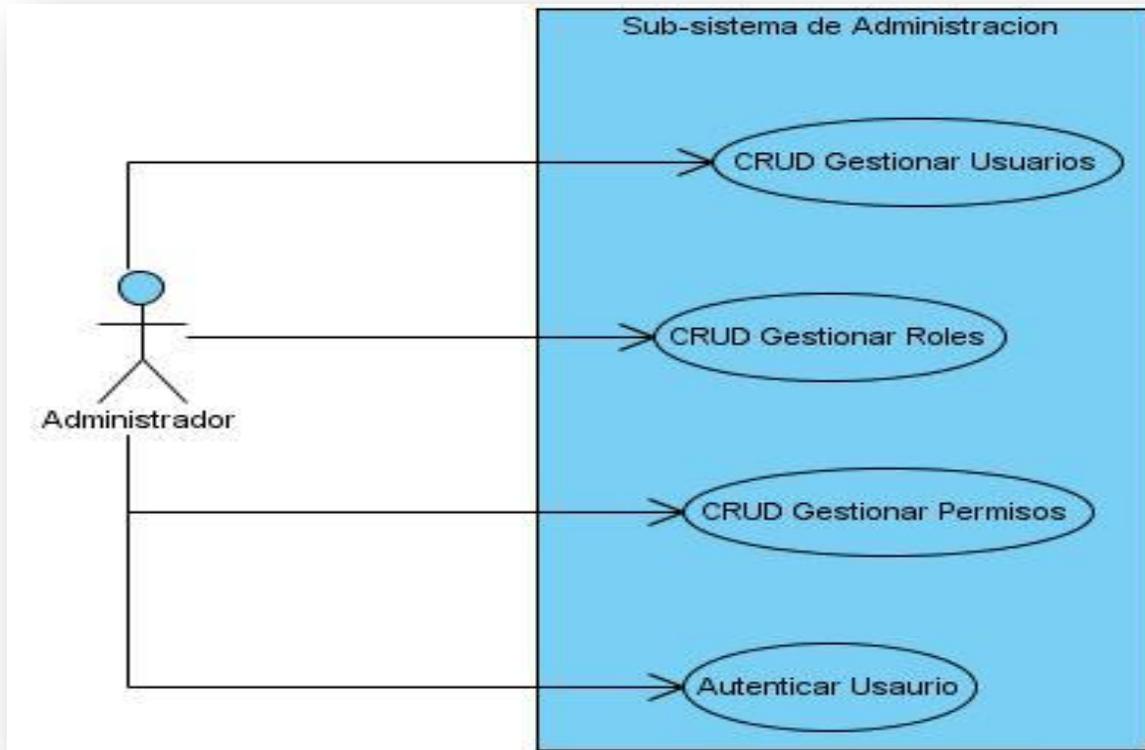


Imagen 11: Módulo de Administración

En el modulo de administración se tienen los casos de uso que dan soporte a la seguridad del sistema. En el mismo es donde se trata toda la parte del acceso y los permisos que van a tener los usuarios a lugares determinados dentro de la aplicación.

2.10 Descripción de los Casos de Uso del Sistema

A continuación se realiza una descripción de uno de los casos de uso del Sistema (CUS). Los restantes pueden verse en el Anexo # 2.

2.10.1 CUS Gestionar Matrícula

Caso de Uso:	Gestión de Matrícula	
Actores:	Secretaria	
Resumen:	El caso de uso se inicia cuando la secretaria selecciona la opción Gestión de Matrícula	
Precondiciones:	1. Ser la secretaria	
Referencias		
Prioridad	Crítico	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	
1. La secretaria selecciona la opción Gestión de Matrícula.	2. El sistema muestra las opciones: Crear Pre listado de Matrícula y Crear Hoja de Matrícula.	
<i>Prototipo de Interfaz</i>		
Sección “Crear Pre listado de Matrícula”		
Acción del Actor	Respuesta del Sistema	
1. La secretaria selecciona la opción Crear Pre listado de Matrícula. 3. La secretaria llena el formulario con los datos del Pre listado de Matrícula 4. La secretaria envía al servidor el Pre listado de Matrícula.	2. El sistema muestra un formulario para los datos a insertar. 5. El sistema valida los datos entrados. 6. En caso de estar correctos, se registra el Pre listado de Matrícula. 7. Fin del flujo.	
Flujos Alternos Sección “Crear Pre listado de Matrícula”		
Acción del Actor	Respuesta del Sistema	
4.2 La secretaria corrige los errores. 4.3 La secretaria envía el Pre listado de Matrícula rectificado al servidor	4.1 En caso de haber errores, se muestra el formulario del Pre listado de Matrícula para ser rectificado.	

	<p>5.4 El sistema ejecuta la acción 5.</p> <p>5.5 Fin del flujo alterno.</p>
<i>Prototipo de Interfaz</i>	
Sección “Crear Hoja de Matrícula”	
Acción del Actor	Respuesta del Sistema
<p>1. La secretaria selecciona la opción Crear Hoja de Matrícula.</p> <p>3. La secretaria selecciona el Pre listado de Matrícula deseado.</p> <p>6. La secretaria llena el formulario con los datos de la Hoja de Matrícula.</p> <p>7. La secretaria envía al servidor la Hoja de Matrícula.</p> <p>11. La secretaria selecciona la opción deseada.</p> <p>12. La secretaria confirma la impresión</p>	<p>2. El sistema muestra los Pre listados de Matrícula disponibles para crear la Hoja de Matrícula.</p> <p>5. El sistema muestra un formulario para llenar la Hoja de Matrícula.</p> <p>8. El sistema verifica los datos.</p> <p>9. En caso de estar correctos, el sistema crea el Listado Oficial del Curso.</p> <p>10. El sistema solicita a la secretaria permisos para imprimir.</p> <p>13. El sistema envía los datos a la impresora.</p> <p>14. Fin del flujo</p>
Flujos Alternos Sección “Crear Hoja de Matrícula”	
Acción del Actor	Respuesta del Sistema
<p>8.2 La secretaria rectifica los errores</p> <p>8.3 La secretaria envía la Hoja de Matrícula rectificada al servidor.</p>	<p>8.1 En caso de haber errores, el sistema muestra el formulario de la Hoja de Matrícula para ser rectificado.</p> <p>8.4 El sistema ejecuta la acción 8</p> <p>8.5 Fin del flujo alterno</p>

12.1 La secretaria cancela la impresión	12.2 El sistema muestra nuevamente las opciones iniciales. 12.3 Fin del flujo alterno.
<i>Prototipo de Interfaz</i>	
Pos condiciones	<ol style="list-style-type: none">1. Pre listado de Matrícula creado en la base de datos2. Hoja de Matrícula creado en la base de datos3. Listado Oficial del Curso creado en la base de datos.

Conclusiones

En el capítulo se estableció y describió el negocio en el que se enmarca el sistema, y los requisitos funcionales y no funcionales. Se identificaron y describieron los casos de uso del sistema, y se especificó que el presente trabajo de diploma mostrará el modelado, la descripción y realización de los casos de uso arquitectónicamente significativos del sistema, dada la dimensión del mismo.

Capítulo 3: Construcción de la Solución Propuesta.

Introducción

En este capítulo se exponen los diagramas de clases del diseño que participan en la realización de los casos de usos arquitectónicamente significativos. Además se representa los distintos diagramas de secuencia de dichos casos de usos. Se da una breve descripción de las clases Entidades y Controladoras empleadas y de las tablas de la Base de Datos, de la que se muestra a su vez los diagramas Entidad – Relación.

3.1 Arquitectura de Software.

Con el transcurso de los años se han ido descubriendo y desarrollando guías generales de aprendizaje para el desarrollo de programas. A estas guías se les ha bautizado con el nombre de Arquitectura de Software, porque como los planos de un edificio estas modelan la forma de desarrollar un determinado software informático.

La Arquitectura de Software, también denominada Arquitectura lógica está integrada por un conjunto de patrones y abstracciones coherentes que proporcionan el marco de referencias necesarias para guiar el desarrollo de un software. Establece los fundamentos para que un equipo de desarrollo trabaje sobre una misma línea común que permita alcanzar los objetivos específicos de un sistema informático.

A la hora de crear un software informático se selecciona o se crea una arquitectura que rige el desarrollo del mismo. Algunas de las diferentes arquitecturas de software que existen son:

- Monolítica: Donde el software se estructura en grupos funcionales muy acoplados.
- Cliente-Servidor: Donde el software reparte su carga de cómputo en dos partes independientes pero sin reparto claro de funciones.
- Arquitectura de tres niveles: Especialización de la arquitectura cliente-servidor donde la carga se divide en tres partes (o capas) con un reparto claro de funciones: una capa para la presentación (interfaz de usuario), otra para el cálculo (donde se encuentra modelado el negocio) y otra para el almacenamiento (persistencia). Una capa solamente tiene relación con la siguiente.

De las arquitecturas antes mencionada se tomó para el desarrollo de la aplicación la Arquitectura en tres Niveles o Capas.

Modelo Vista Controlador

Modelo Vista Controlador (Model-View-Controller) Patrón arquitectónico desarrollado para interfaces gráfica que resalta la importancia de una separación clara entre la presentación de datos y la lógica de negocio de una aplicación.

3.2 Análisis.

Como resultado del desarrollo de cada uno de los casos de usos más fundamentales se han desarrollado los diagramas de clases de análisis. Estos representan como se lleva a cabo el desarrollo de los diferentes casos de uso dentro de la aplicación. Para un mayor entendimiento de los mismos se recomienda ver el anexo # 3 del documento.

3.3 Diseño

3.3.1 Clases del diseño

El Modelo de Diseño es un proceso para la definición detallada de un sistema con el fin de la realización física de los casos de uso para cubrir las funciones que realizará el sistema y otras restricciones del entorno de implementación que tienen impacto en el mismo, por tanto en él se definen las clases del diseño que conformarán el sistema que se va a implementar.

- **Páginas clientes:** Son las páginas encargadas de permitir a los usuarios interactuar con el sistema tanto para hacer solicitudes como para que sean mostradas las respuestas a las mismas.
- **Páginas servidoras:** Son las encargadas de la construcción de forma dinámica de las páginas clientes y sirven de enlace entre estas y el resto de las clases.
- **Páginas controladoras:** Son las responsables de realizar las operaciones que responden a los procesos de negocio y dar respuestas a las solicitudes hechas por el usuario.
- **Clases entidad:** Son las responsables de la persistencia de los datos físicamente.

Los diagramas de Clases del Diseño que componen el Modelo de Diseño se muestran en el anexo # 4.

3.3.2 Diagrama de Componentes

Un componente es una parte modular de un sistema, desplegable y reemplazable que típicamente contiene clases y puede ser implementado por uno o más artefactos (ficheros ejecutables, binarios, etc.). Es un elemento de implementación que representa algo físico, ya sea ficheros o archivos, y son creados

para poner el código, ya sea código fuente, código binario o código ejecutable. Y son creados, modificados o eliminados en el proceso de implementación.

Los diagramas de componentes son usados para estructurar el modelo de implementación en términos de subsistemas de implementación y mostrar las relaciones entre los elementos de implementación. Es un diagrama que muestra un conjunto de elementos del modelo tales como componentes, subsistemas de implementación y sus relaciones. En el Anexo # 5 se muestra el diagrama de componentes de la aplicación Web SGA, elemento fundamental del sistema.

3.3.3 Diagrama de Despliegue

El Diagrama de Despliegue es un grafo de nodos unidos por conexiones de comunicación que muestra las relaciones físicas entre los componentes de hardware que forman la topología sobre la que se ejecuta el sistema y la distribución de sus partes. Este refleja tanto la distribución del sistema con los nodos físicos (ordenadores) como la correspondencia que tienen los componentes con los nodos.

En este caso se pueden identificar como procesadores uno para la base de datos, otro para que funcione como servidor Web y uno que representa en general todos los clientes que puedan conectarse a la aplicación, así como el dispositivo necesario para obtener la versión impresa de los diferentes reportes. En el Anexo # 6 se muestra el diagrama de componentes de la aplicación Web SGA, elemento fundamental del sistema.

3.3.4 Diagrama de Clase Persistentes

Como parte del diseño se ha desarrollado el diagrama de clases persistentes que muestra las clases persistentes que se tienen dentro del sistema y la relación que existe dentro de ellas. Dicho diagrama se muestra en el Anexo # 7.

3.3.5 Diseño de Base de Datos

Los sistemas web que en la actualidad existen son desarrollados utilizando tecnologías cliente-servidor. Estos constan en la mayoría de los casos con bases de datos robustas que son las encargadas de almacenar todos los datos que en la misma se muestran.

SGA debido a la complejidad y el cúmulo de información que se maneja se ha desarrollado una base de datos para almacenar toda la información que se controla dentro del CPP. La misma viene incluida dentro de los anexos. (Ver Anexo # 8).

3.3.5.1 Descripción de las tablas

Las principales entidades de la base de datos son descritas a continuación.

TipoCursos

Nombre: tipocursos		
Descripción: En esta tabla se almacenan todos los atributos que son comunes para cada una de las tablas que heredan de la misma que son diplomados, especialidades, maestrías y postgrados.		
Atributo	Tipo	Descripción
Nombre	Varchar(100)	Se almacena el nombre que tiene el tipo de curso
Código (PK)	Varchar(100)	Almacena el código único de cada uno de los tipos de cursos. Este código es generado de forma dinámica.
Total_horas	Int	Almacena la cantidad de horas de las que consta el tipo de curso que se va a impartir.

Tabla 12

Curso

Nombre: Curso		
Descripción: En esta tabla se almacenan toda la información de los cursos que son impartidos dentro del CPP.		
Atributo	Tipo	Descripción
Código (PK)	Varchar(100)	Almacena el código único de cada uno de los cursos. Este código es generado de forma dinámica.
Nombre	Varchar(255)	Almacena el nombre que tiene el curso
Coordinador	Varchar(255)	Almacena la información de quien coordina las actividades que se realizan en el curso
Cantidad_horas	Int	Almacena la cantidad de horas de las que consta el curso.

Tabla 13

Persona

Nombre: Persona		
Descripción: En esta tabla se almacenan toda la información común de las tablas hijas Profesores y cursistas.		
Atributo	Tipo	Descripción
CI (PK)	Varchar(11)	Almacena el CI de las diferentes personas que se guardan.
Nombre	Varchar(255)	Almacena el nombre de la persona
Apellidos	Varchar(255)	Almacena los apellidos de las personas
Municipio (FK)	Varchar(255)	Almacena el nombre del municipio en el cual reside la persona.
Provincia(FK)	Int	Almacena el id de la provincia en la cual vive la persona.

Tabla 14

Cursista

Nombre: Cursista		
Descripción: En esta tabla se almacenan toda la información referente al cursista como tal.		
Atributo	Tipo	Descripción
PersonaCI (PK)	Varchar(11)	Almacena el CI de las diferentes personas que se guardan.
EmpresalID(FK)	Int	Almacena el id de la empresa a la cual pertenece la persona.
CategoriaOcupacionalid(FK)	Int	Almacena el id de la categoría ocupacional a la cual pertenece el cursista.

Tabla 15

Profesor

Nombre: Profesor		
Descripción: En esta tabla se almacenan toda la información referente al profesor.		
Atributo	Tipo	Descripción

CI (PK)	Varchar(11)	Almacena el CI de las diferentes personas que se guardan.
Especialidad	Varchar(255)	Almacena la especialidad a la cual pertenece el profesor.
Categoria	Varchar(255)	Almacena la categoría a la cual pertenece el profesor.
Curriculum	Varchar(255)	Almacena el currículum del profesor.
Telefono	Int	Almacena el teléfono del profesor.
Email	Varchar(100)	Almacena el correo electrónico del profesor.
CursoCodigo(FK)	Varchar(100)	Almacena el código del curso que imparte el profesor.

Tabla 16

Conclusiones

En el capítulo que recién concluye se trataron temas relacionados con la arquitectura seleccionada el análisis y el diseño del sistema que se presenta, así como elementos específicos dentro del mismo que quedan plasmados en los anexos del documento debido a su extensión. Entre los cuales se encuentran el Modelo de Clases Persistentes y el Modelo Entidad Relación que contiene todo el modelo de datos que se ha elaborado para el soporte de los datos que se mantendrán almacenados en la aplicación.

Capítulo #4: Estudio de Factibilidad

Introducción.

Cuando se comienza con el desarrollo de un proyecto se debe tener en cuenta la estimación de tiempo y esfuerzo del que requiere el mismo. En la mayoría de los casos la estimación se hace basada en la experiencia obtenida como única guía. Aunque en algunos casos la experiencia no se suficiente.

4.1 Técnica de Estimación.

Las técnicas de estimación son una forma de resolución de problemas. En la actualidad existen diversos métodos para realizar la estimación de un proyecto entre las más usadas se encuentran COCOMO II y el Análisis de Puntos de Casos de Uso.

4.1.1 Análisis de Puntos de Función y COCOMO II.

Este método está basado en ecuaciones matemáticas que permiten calcular el esfuerzo a partir de ciertas métricas de tamaño estimado, como: el Análisis de Puntos de Función y las Líneas de Código Fuente (en inglés SLOC, Source Line Of Code).

4.1.2 Análisis de Puntos de Casos de Uso.

Existe una posibilidad de predecir el tamaño de un sistema a partir de las características de sus requisitos, expresados en los casos de uso.

4.2 Estimación de Tiempo y esfuerzo utilizando Análisis de Punto de Casos de Uso.

Para llevar a cabo la estimación de tiempo y esfuerzo que se requiere para la implementación se utilizó la técnica de estimación Análisis por Punto de Casos de USO que nos permitió teniendo bien definido todos los actores y los casos de Uso, determinar el tiempo que se demoraría la creación del sistema.

Para ello tenemos relacionado los diferentes actores que existen en la Tabla 17, para calcular UAW⁶.

Tipo de Actor	Descripción	Cant.* Peso
Simple	Otro sistema que interactúa con el sistema a desarrollar mediante una interfaz de programación (API, Application Programming Interface)	0*1

⁶ Factor de Peso de los Actores sin ajustar.

Medio	Otro sistema que interactúa con el sistema a desarrollar mediante un protocolo o una interfaz basada en texto.	0*2
Complejo	Una persona que interactúa con el sistema mediante una interfaz gráfica.	5*3
Total		15

Tabla 17

Para calcular UUCW⁷ teniendo en cuenta cada uno de los casos de uso que se tienen en la aplicación y la cantidad de transacciones que cada uno contiene se muestra en la Tabla 18.

Tipo	Descripción	Peso	Cant * peso
Simple	El Caso de Uso contiene de 1 a 3 transacciones	5	4*5
Medio	El Caso de Uso contiene de 4 a 7 transacciones	10	10*10
Complejo	El Caso de Uso contiene más de 8 transacciones	15	3*15
Total			165

Tabla 18

Posteriormente a realizar las relaciones que anteriormente se detallan se procedió a calcular el UUCP⁸ utilizando la fórmula $UUCP = UAW + UUCW$.

$$UUCP = 15 + 165 = 180$$

4.3 Ajustando los Puntos de Casos de Uso.

Una vez que tenemos los puntos de casos de uso sin ajustar procedemos a ajustarlo utilizando la fórmula $UCP^9 = UUCP^{10} \times TCF^{11} \times EF^{12}$.

Para ello comenzamos calculando el TCF. Este coeficiente se calcula mediante la cuantificación de un conjunto de factores que determinan la complejidad técnica del sistema. Cada uno de los factores se

⁷ Factor de Peso de los Casos de Uso sin ajustar.

⁸ Puntos de Casos de Uso sin ajustar

⁹ Puntos de Casos de Uso ajustados.

¹⁰ Puntos de Casos de Uso sin ajustar.

¹¹ Factor de complejidad técnica.

¹² Factor de ambiente.

Sistema para la Secretaria Docente del Centro Politécnico del Petróleo

cuantifica con un valor de 0 a 5, donde 0 significa un aporte irrelevante y 5 un aporte muy importante. En la Tabla 19 se muestra el significado y peso de cada uno de estos factores.

Para calcular el TCF utilizamos la ecuación que a continuación se relaciona.

$$TCF = 0.6 + 0.01 * \sum (\text{Peso}_i * \text{Valor}_i) \text{ (Donde valor es un número entre 0 y 5).}$$

Significado de los valores

0: No presente o sin influencia,

1: Influencia incidental o presencia incidental

2: Influencia moderada o presencia moderada

3: Influencia media o presencia media

4: Influencia significativa o presencia significativa

5: Fuerte influencia o fuerte presencia

Factor	Descripción	Peso	Valor	Comentario	$\sum (\text{Peso}_i * \text{Valor}_i)$
T1	Sistema distribuído	2	0	El sistema es Centralizado	0
T2	Objetivos de performance o tiempo de respuesta	1	3	La velocidad de respuesta del sistema debe ser la de menor tiempo posible.	3
T3	Eficiencia del usuario final	1	1	Escasas restricciones de eficiencia	1
T4	Procesamiento interno complejo	1	0	No hay cálculos Complejos	0
T5	El código debe ser reutilizable	1	2	Se pretende reutilizar el modulo de administración para aplicaciones que se implementen en un futuro	2
T6	Facilidad de instalación	0.5	1	Escasos requerimientos de facilidad de instalación	0.5
T7	Facilidad de uso	0.5	3	Normal	1.5
T8	Portabilidad	2	0	No se requiere que el	0

				sistema sea portable	
T9	Facilidad de cambio	1	3	Se requiere un costo moderado de mantenimiento	3
T10	Concurrencia	1	0	No hay concurrencia	0
T11	Incluye objetivos especiales de seguridad	1	4	Se requiere de gran seguridad debido a los fines del software.	4
T12	Provee acceso directo a terceras partes	1	5	Los usuarios web tienen acceso directo	5
T13	Se requieren facilidades especiales de entrenamiento a los usuarios	1	1	Pocos usuarios internos, sistema fácil de usar.	1
Total					21

Tabla 19

$$TCF = 0.6 + 0.01 * 21 = 0.81.$$

A continuación se procede a calcular el EF utilizando la fórmula

$$EF = 1.4 - 0.03 * \sum (\text{Peso}_i * \text{Valor}_i) \text{ (Donde Valor es un número del 0 al 5)}$$

Factor	Descripción	Peso	Valor	Comentario	$\sum (\text{Peso}_i * \text{Valor}_i)$
E1	Familiaridad con el modelo de proyecto utilizado	1.5	4	El grupo está bastante familiarizado con el modelo	6
E2	Experiencia en la aplicación	0.5	3	Parte del grupo ha trabajado algún tiempo en ésta aplicación	2
E3	Experiencia en	1	4	La mayoría del grupo	4

orientación a objetos				programa en Objetos	
E4	Capacidad del analista líder	0.5	4	Es bastante bueno en el trabajo que realiza.	2
E5	Motivación	1	5	El grupo está altamente motivado	5
E6	Estabilidad de los requerimientos	2	2	Se esperan cambios	4
E7	Personal part-time	-1	3	Algunos dentro del grupo se encuentran vinculados en otras tareas que hacen que no dedique el 100% del tiempo al trabajo.	-3
E8	Dificultad del lenguaje de programación	-1	2	Se usará lenguaje php	-2
Total					18

Tabla 20

$$EF = 1.4 - 0.03 * 18 = 0.86.$$

Teniendo ya UUCP, TCF y EF procedemos a calcular UCP.

$$UCP = 180 * 0.81 * 0.86 = 125.388$$

Calculando el esfuerzo de trabajo del flujo de Implementación utilizando la fórmula $E^{13} = UCP * CF^{14}$.

Para calcular CF según las Horas Hombre.

$$CF = 20 \text{ horas-hombre (si } Total_{ef} \leq 2)$$

$$CF = 28 \text{ horas-hombre (si } Total_{ef} = 3 \text{ o } Total_{ef} = 4)$$

$$CF = \text{abandonar o cambiar proyecto (si } Total_{ef} \geq 5)$$

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

¹³ Esfuerzo estimado en horas-hombre.

¹⁴ Factor de conversión.

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

CF = 20 horas-hombre.

Posteriormente procedemos a calcular el Esfuerzo.

$$E = 125.388 * 20 = 2507.76 \text{ horas-hombre}$$

El esfuerzo que se ha calculado hasta el momento es solo para el flujo de trabajo de implementación y a continuación se procede a calcular el esfuerzo para todo el ciclo de desarrollo del proyecto.

Actividad	% esfuerzo	Valor esfuerzo
Análisis	20%	1253.5 horas-hombre
Diseño	20%	1253.5 horas-hombre
Implementación	40%	2507.76 horas-hombre
Prueba	10%	627.75 horas-hombre
Sobrecarga	10%	627.75 horas-hombre
Total	100%	6267.5 horas-hombre

Tabla 21

Suponiendo que una persona trabaje 8 horas por día, y un mes tiene como promedio 30 días; la cantidad de horas que puede trabajar una persona en 1 mes es 240 horas

Si $E_T = 6267.5$ horas-hombre y por cada 240 horas se tiene 1 mes eso daría un $E_T = 26.1$ mes-hombre

Teniendo en cuenta que el proyecto fue realizado por 3 personas que emplearon el mismo esfuerzo y tiempo, se puede dar por concluido que el sistema se puede desarrollar en un tiempo de 9 meses, sin tener en cuenta los inconvenientes que afectan de forma directa el desarrollo del proyecto.

Conclusiones.

Se concluye diciendo que el sistema que se desarrolló ,para un grupo de desarrollo de 3 integrantes tomó un tiempo de 9 meses aproximadamente utilizando el método de estimación Puntos de Casos de Uso. Teniendo un UUCP = 180, un TCF = 0.81, un EF = 0.86, un E = 2507.5 para el FT Implementación y un E = 6267.5 horas-hombres para el desarrollo del proyecto completo.

Conclusiones Generales.

Durante el desarrollo del presente trabajo se llevaron a cabo los diferentes flujos de trabajo que propone la metodología utilizada, se investigó el estado del arte de las aplicaciones de Gestión Académica y se estudiaron y seleccionaron las tendencias y tecnologías más factibles para su desarrollo. Dándole cumplimiento a las tareas de investigación propuestas. Al término de la investigación se logró la implementación de la versión 1.0 del Sistema de Gestión Académica (SGA), capaz de gestionar la información resultante de los procesos de negocio de la Secretaría Docente del Centro Politécnico de Petróleo. Este permite la gestión de los estudiantes y los diferentes tipos de cursos que se imparten, así como la generación de documentación a través de diversos reportes en formato PDF. Además de dar respuesta en menor tiempo a reportes solicitados por la dirección de politécnico. También se corroboró la compatibilidad del sistema con la plataforma libre, garantizando que su despliegue sea en un servidor instalado con la distribución Debian 4.0 o superior del sistema operativo Linux aunque para las PC clientes se asegura la completa compatibilidad con sistemas de tipo UNIX y Windows.

RECOMENDACIONES

Sobre la presente investigación los autores recomiendan:

- Identificar nuevas funcionalidades a incorporar al sistema.
- Establecer las condiciones que permitan automatizar los procesos del CPP con otras entidades de CUPET.
- implantar el mismo sistema en las secretarías de los de demás CPPs del país.

Bibliografía Consultada

- **Booch, Rumbaugh, Jabcoson.** 1999. El Lenguaje Unificado de Modelado. S.I.: Addison-Wesley, 1999.
- **Booch, Rumbaugh, Jacobson.** 1999. The Unified Software Development Process. S.I.: Addison-Wesley, 1999.
- **Brewer, J.** 2001. Extreme Programing FQA. [En línea] 2001. <http://www.jera.com/techinfo/xpfaq.html>.
- **Ceria, Santiago.** 2006. Ingeniería de Software I. Casos de Uso. Un Método Práctico para explorar requerimientos. 2006.
- **Cesares, Claudio.** 2007. Curso de SQL. <http://www.programacion.com/bbdd/tutorial/sql/1/>.
- **Conallen, Jim.** 1999. Building Web Applications with UML. S.I.: Addison-Wesley, 1999. —. 1999. Modeling Web Applications with UML. ModelingWebApplications. [En línea] Marzo de 1999. <http://www.connallen.org/whitepapers/webapps/ModelingWebApplications>.
- **Darío.** 2006. ¿Qué es Joomla? [En línea] 2006. <http://www.maestrosdelweb.com/editorial/que-es-joomla/>.
- **Giraldo, Zapata.** Herramientas de desarrollo de Ingeniería de Software para Linux. [En línea] <http://hugolopez.phi.com.co/docs.74>.
- **Henst S, Christian Van Der.** 2001. Qué es PHP. <http://www.maestrosdelweb.com/editorial/phpintro>.
- **Jabcoson, I., Booch, G., Rumbaugh, J.** 2000. El proceso unificado de desarrollo de software. [Catálogo en línea de la Biblioteca de la Universidad de las Ciencias Informáticas] s.l.: Addison-Wesley, 2000.
- **L, Néstor a. Díaz.** 2003. Ventajas de PostgreSQL. http://soporte.tiendalinux.com/portal/Portfolio/postgresql_ventajas_html.
- **Larman, Craig.** 2002. UML y Patrones, Introducción al análisis y diseño orientado a objetos. S.I.: Prentice-Hall, 2002.
- **Lévenez, É.** 2007. Computer Lenguajes History. [En línea] 2007. <http://www.levenez.com.lang/>.
- **Mato García, Rosa María.** 2005. Sistema de Base de Datos. S.I.: Editorial Pueblo y Educación, 2005. págs. 3-4.

- **Molpeceres, A.** 2002. Procesos de desarrollo: Rup, XP y FDD. S.I.: Tomado de la Ayuda del Rational Unified Process, 2002.
- **Mononeurona.** 2007. ¿PHP, Python, ASP, Perl o JSP? <http://www.mononeurona.org/pages/display/150>.
- **MySQL.** 2008. Panorámica del sistema de gestión de base de datos MySQL. [En línea] 2008. <http://dev.mysql.com/doc/refman/5.0/es/what-is.html>.
- **Paradimg, Visual.** 2007. Why Visual Paradimg for UML. [En línea] 2007 <http://www.visual-paradimg.com/>.
- **Pressman, Roger S.** 2002. Ingeniería del Software, un enfoque práctico. S.I.: Mc Graw-Hill, 2002. pág. 14.
- **S, Christian Van Der Henst.** 2001. Qué es PHP. <http://www.maestrosdelweb.com/editorial/phpintro>.
- **Schmulerr, Joseph.** 2004. UML. [En línea] 2004. <http://usuarios.lycos.es/oopere/uml.htm>.
- **Sintes Marco, Bartolomé.** Catálogo de Software Libre. [En línea] [http://www.cdlibre.org/consultar/catalogo/Desarrollo-Web_Sistemas-de-gestion-de-contenidos-\(CMS\).html](http://www.cdlibre.org/consultar/catalogo/Desarrollo-Web_Sistemas-de-gestion-de-contenidos-(CMS).html).
- **Teleformación.** 2007. Introducción al proceso de desarrollo de software. [En línea] 2007. <http://teleformacion.uci.cu/mod/resource/view.php?id=11402>.
- **Tomás Javier Robles Prado, Raúl Turienzo Fernández.** 2007. Introducción a PostgreSQL. [En línea] 2007. <http://www.programacion.com/bbdd/tutorial/intropostgresql/2/>.
- **Villariño, Luis.** 2002. UML para Web.
- **Wells, D.** 2006. Extreme Programing: A gentle introduction. <http://www.extremeprograming.org/>.

Referencias Bibliográfica

- Baeza-Yayes, R. (1993) "¿Cuál ha sido el primer navegador de Internet?" **Volumen**, DOI:
- Zapata, F. C. (2008) "¿Que es UNISIGA2?" **Volumen**, DOI:
- ESENTIA. (2004). "El Sistema MentorVox." Retrieved 3, 2009, from <http://www.esentia.com/mentorvox.htm>.

- Yisel Alfaro Abreus, I. M. A. L., Ing. Orelvis Vázquez Ruiz, Dr. Julio Martínez Prieto, Ing. Carmen R. Martínez San Emeterio, Ing. Yunev Marín Silva, Alexis Cedeño Trujillo, Alejandro Martínez Grau, Yenisleidy Vallejo Ramos, Yosvanys Aponte Báez (2004) "Sistema de Gestión Académica." **Volume**, DOI:
- González, D. J. A. T., L. Y. C. Delgado, et al. (2001) "GESTACAD: UN SISTEMA PARA LA GESTIÓN ACADÉMICA UNIVERSITARIA." **Volumen**, 12 DOI:
- DALMENDRAY, F. B. (2006) **Volumen**, DOI:
- Schwaber, K. (2004). Agile Project Management with Scrum.
- Fernández, C. A. F. Y. (2000) "El Proceso Unificado de Rational para el Desarrollo de Software." **Volumen**, 22 DOI:
- Pablo, J. (2007).
- JHONATAN MINA, F. M., ANDRES FDO PEREZ (2008) "Resumen ejecutivo RUP." **Volumen**, DOI:
- Torres, P. L. (2005) "Desarrollo de Software Orientado a Objeto usando UML." **Volumen**, DOI:
- Hinostroza, R. R. (2007) "Características de PHP." **Volumen**, DOI:
- Kendall, K. (2005). "Análisis y Diseño de Sistemas."
- Pecos, D. (2008) "PostgreSQL vs. MySQL." **Volume**, DOI:
- Quiñones, E. (2006) "INTRODUCCION A POSTGRESQL." **Volumen**, DOI:
- Larman, C. (2004). Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development.
- Jacobson, I. H. (1999). The Unified Software Development Process.

Anexos.

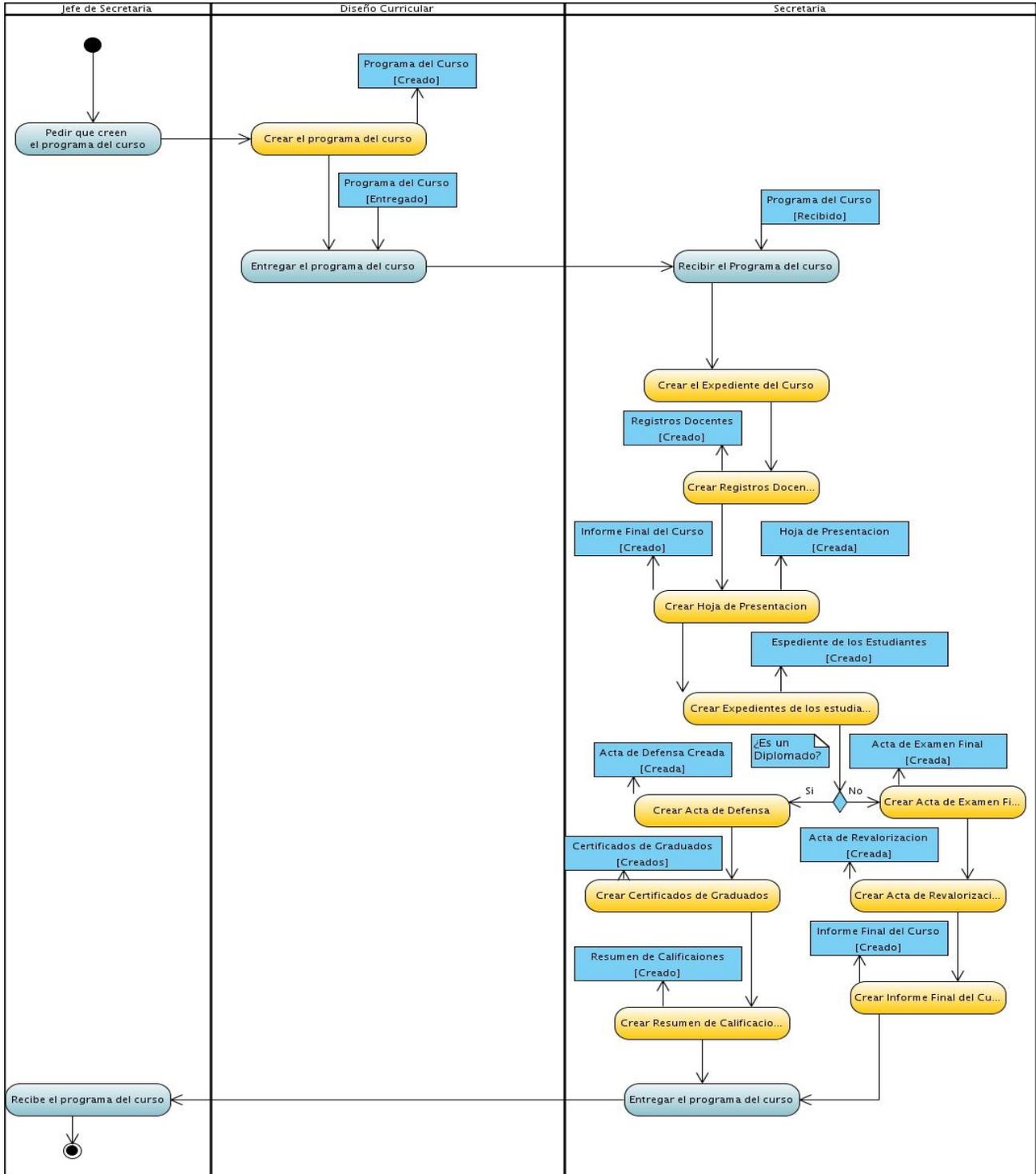
Glosario de Términos.

A	
B	
Base de Datos Relacional	Una base de datos relacional es una base de datos en donde todos los datos visibles al usuario están organizados estrictamente como tablas de valores, y en donde todas las operaciones de la base de datos operan sobre estas tablas.
C	
Cliente Ligero	Un Cliente Liviano (Thin client) es una computadora (cliente) en una arquitectura de red cliente-servidor que tiene muy poca o ninguna lógica del programa
D	
E	
F	
Feeds	Es un archivo generado por algunos sitios web (y por muchos weblogs) que contiene una versión específica de la información publicada en esa web.
G	
H	
I	
Intranet	Una Intranet es una red de ordenadores privados que utiliza tecnología Internet para compartir de forma segura cualquier información o programa del sistema operativo
Internet	Internet es un conjunto descentralizado de redes de comunicación interconectadas, que utilizan la familia de protocolos TCP/IP.
J	
K	
L	

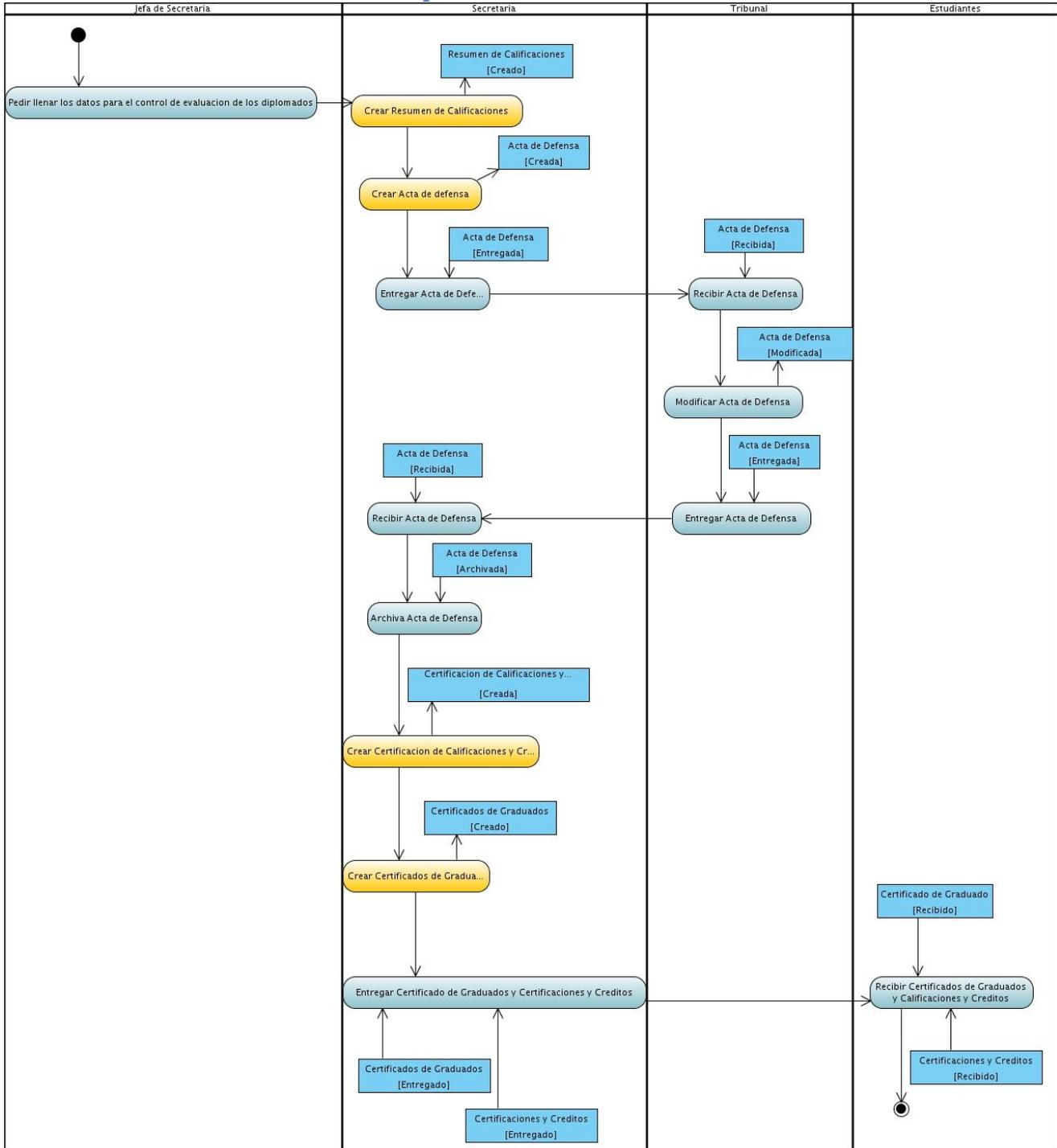
Licencia BSD	La licencia BSD es la licencia original de una distribución de Software: <i>Berkeley Software Distribution</i> , que acabó convirtiéndose en un derivativo de UNIX realizado por la conocida Universidad de California, Berkeley. Hablamos de BSD UNIX, del que derivan los diferentes *BSD Libres de los que a veces hablo por aquí. (2004)
M	
N	
Navegador	Un navegador web (del inglés, web browser) es una aplicación software que permite al usuario recuperar y visualizar documentos de hipertexto, comúnmente descritos en HTML, desde servidores web
Ñ	
O	
P	
Q	
R	
S	
Script CGI	Es una tecnología del lado del cliente que se combina con un formulario para llenar con miras a recolectar y tratar datos introducidos por el usuario.
T	
U	
V	
W	
Web	Un servidor web es un programa que implementa el protocolo HTTP (hypertext transfer protocol)
X	

Y	
Z	

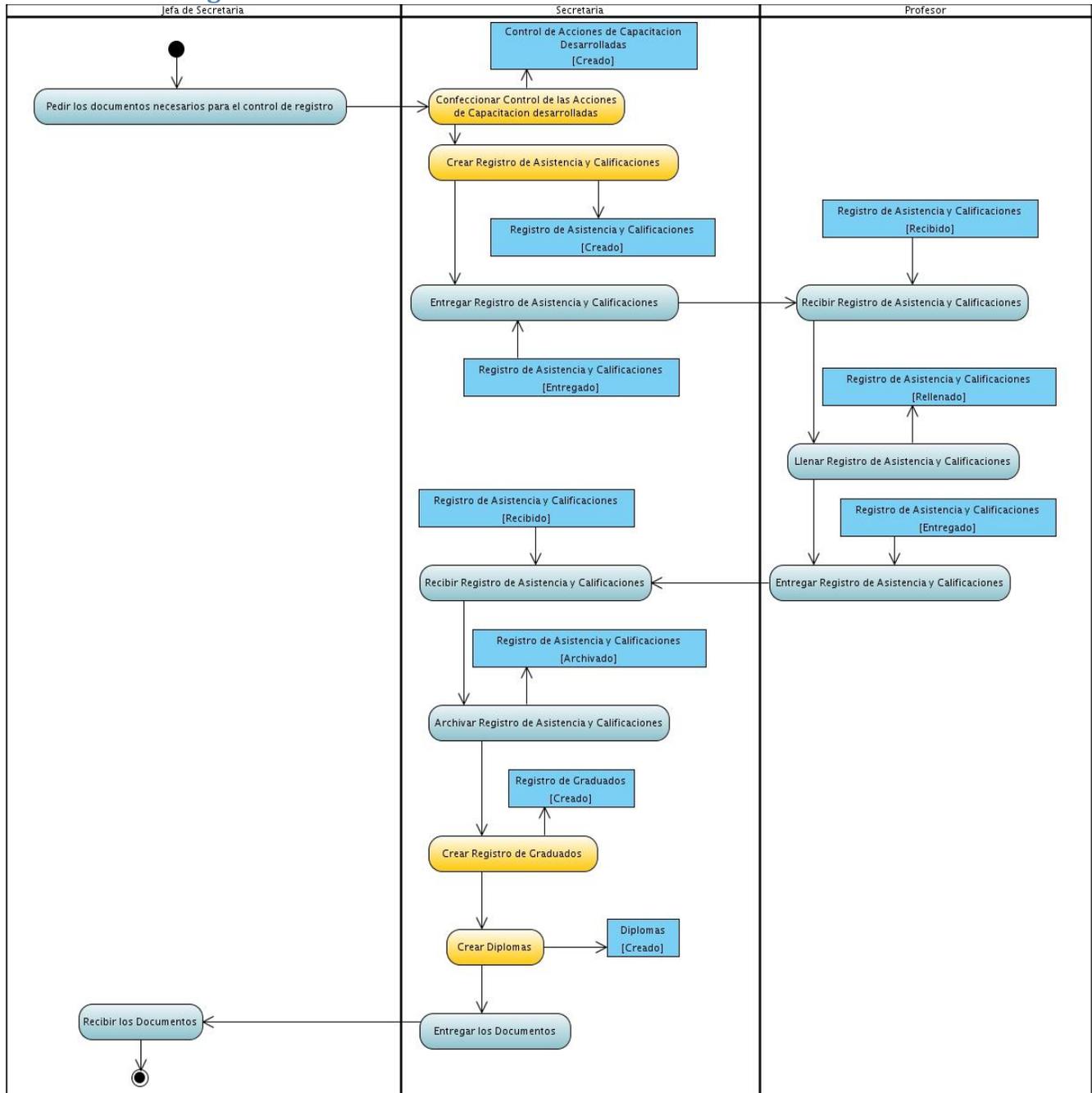
DA Gestionar Curso



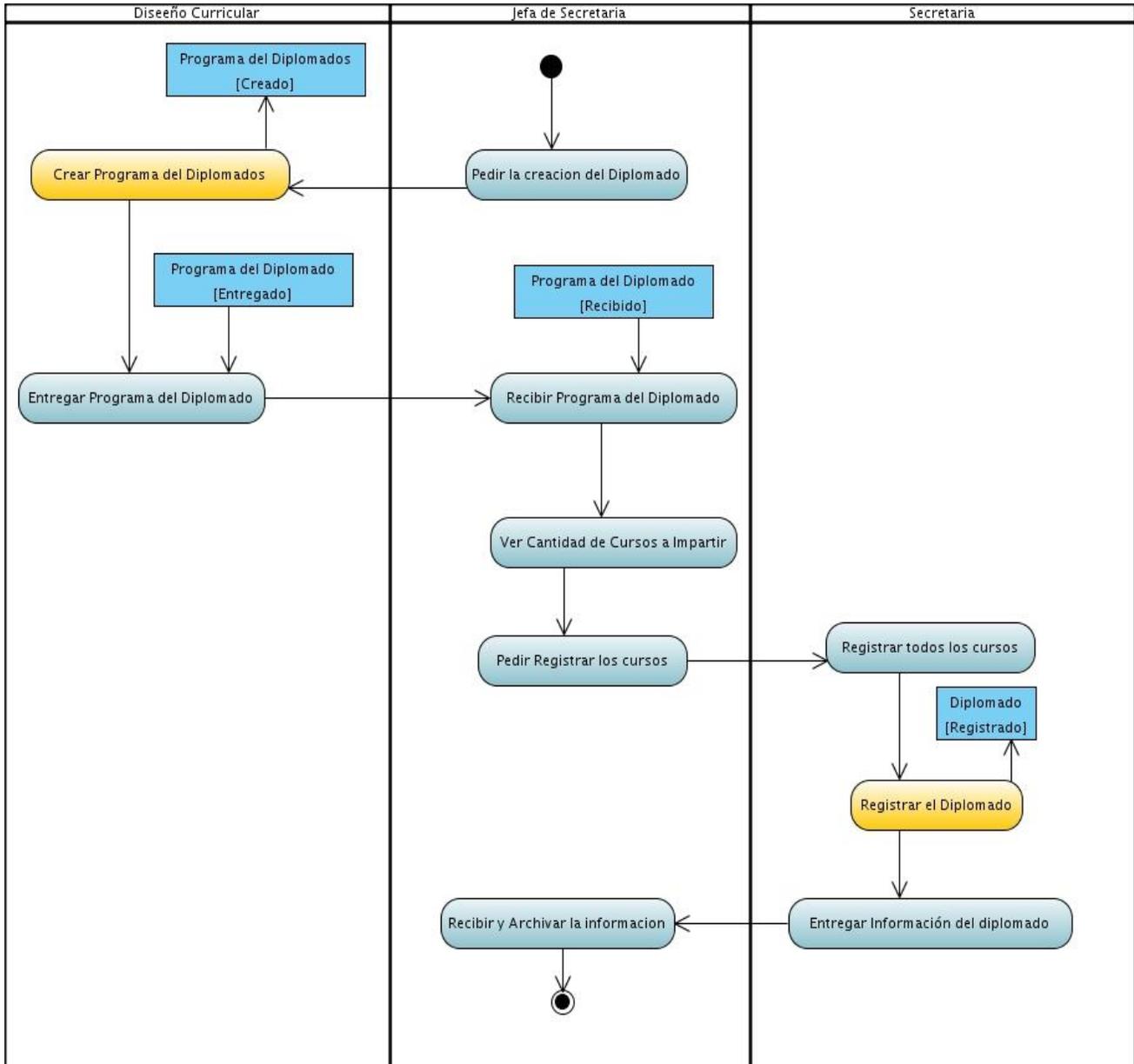
DA Control de Evaluaciones de Diplomados



DA Control de Registro



DA Gestionar Diplomado



Anexo #2

Descripción del CUS Gestionar Curso

Caso de Uso:	CRUD Gestionar Curso	
Actores:	Secretaria	
Resumen:	El caso de uso se inicia cuando la secretaria selecciona la opción Gestionar Cursos	
Precondiciones:	1. Ser la secretaria	
Referencias		
Prioridad	Crítico	
Flujo Normal de Eventos		
Sección "Gestionar Cursos"		
Acción del Actor	Respuesta del Sistema	
1. La secretaria selecciona la opción de Gestionar Cursos	2. El sistema muestra las opciones: Insertar Curso, Modificar Curso y Eliminar Curso.	
Sección "Insertar Curso"		
Acción del Actor	Respuesta del Sistema	
1. La secretaria selecciona la opción Insertar Curso. 3. La secretaria llena el formulario con los datos del nuevo curso 4. La secretaria envía la solicitud al servidor.	2. El sistema muestra un formulario a llenar con los datos del curso a insertar. 5. El sistema verifica que no haya redundancia o errores en los datos. 6. En caso de no existir problemas, se registra el nuevo curso en el sistema. 7. Fin del flujo	
<i>Prototipo de Interfaz</i>		

Flujos Alternos	
Acción del Actor	Respuesta del Sistema
<p>5.2. La secretaria corrige los problemas señalados.</p> <p>5.3. La secretaria envía la solicitud al servidor</p>	<p>5.1. En caso de existir redundancia o errores en los datos, se le solicita a la secretaria que corrija los problemas encontrados en los datos del formulario.</p> <p>5.4. El sistema ejecuta la acción 5.</p> <p>5.5. Fin del flujo alternativo</p>
<i>Prototipo de Interfaz</i>	
Sección "Modificar Curso"	
Acción del Actor	Respuesta del Sistema
<p>1. La secretaria selecciona la opción Modificar Curso.</p> <p>3. La secretaria modifica los datos del curso.</p> <p>4. La secretaria envía al servidor el formulario con los datos modificados.</p>	<p>2. El sistema muestra un formulario con los datos del curso solicitado.</p> <p>5. El sistema verifica que no haya redundancia o errores en los datos.</p> <p>6. En caso de no existir problemas, se actualiza la información del curso en el sistema.</p> <p>7. Fin del flujo</p>
<i>Prototipo de Interfaz</i>	
Flujos Alternos	
Acción del Actor	Respuesta del Sistema

<p>5.2. La secretaria corrige los problemas señalados.</p> <p>5.3. La secretaria envía la solicitud al servidor</p>	<p>5.1. En caso de existir redundancia o errores en los datos, se le solicita a la secretaria que corrija los problemas encontrados en los datos del formulario.</p> <p>5.4. El sistema ejecuta la acción 5.</p> <p>5.5. Fin del flujo alterno</p>
<i>Prototipo de Interfaz</i>	
Sección "Eliminar Curso"	
Acción del Actor	Respuesta del Sistema
<p>1. La secretaria selecciona un curso para ser eliminado.</p> <p>3. La secretaria selecciona la opción deseada</p>	<p>2. El sistema muestra un mensaje de confirmación para eliminar el curso seleccionado.</p> <p>4. En caso de haber confirmado la eliminación, el sistema elimina los datos del curso registrados en el sistema.</p> <p>5. Fin del flujo</p>
<i>Prototipo de Interfaz</i>	
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
	<p>4.1 En caso haber cancelado la eliminación, se termina el caso de uso.</p> <p>4.2 Fin del flujo alterno.</p>
<i>Prototipo de Interfaz</i>	
Pos condiciones	<p>1. Curso creado en la base de datos</p> <p>2. Curso modificado en la base de datos</p> <p>3. Curso eliminado de la base de datos</p>

Descripción del CUS Gestionar Diploma

Caso de Uso:	CRUD Gestionar Diploma	
Actores:	Secretaria	
Resumen:	El caso de uso se inicia cuando la secretaria selecciona la opción Gestionar Diplomas	
Precondiciones:	1. Ser la secretaria.	
Referencias		
Prioridad	Crítico	
Flujo Normal de Eventos		
Sección "Gestionar Diplomas"		
Acción del Actor	Respuesta del Sistema	
1. La secretaria selecciona la opción de Gestionar Diplomas	2. El sistema muestra las opciones: Insertar Diploma, Modificar Diploma y Eliminar Diploma.	
Sección "Insertar Diploma"		
Acción del Actor	Respuesta del Sistema	
1. La secretaria selecciona la opción Insertar Diploma. 3. La secretaria llena el formulario con los datos del nuevo diploma 4. La secretaria envía la solicitud al servidor.	2. El sistema muestra un formulario a llenar con los datos del diploma a insertar. 5. El sistema verifica que no haya redundancia o errores en los datos. 6. En caso de no existir problemas, se registra el nuevo diploma en el sistema. 7. Fin del flujo	

<i>Prototipo de Interfaz</i>	
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
5.2. La secretaria corrige los problemas señalados. 5.3. La secretaria envía la solicitud al servidor	5.1. En caso de existir redundancia o errores en los datos, se le solicita a la secretaria que corrija los problemas encontrados en los datos del formulario. 5.4. El sistema ejecuta la acción 5. 5.5. Fin del flujo alternativo
<i>Prototipo de Interfaz</i>	
Sección "Modificar Diploma"	
Acción del Actor	Respuesta del Sistema
1. La secretaria selecciona la opción Modificar Diploma. 3. La secretaria modifica los datos del diploma. 4. La secretaria envía al servidor el formulario con los datos modificados.	2. El sistema muestra un formulario con los datos del diploma solicitado. 5. El sistema verifica que no haya redundancia o errores en los datos. 6. En caso de no existir problemas, se actualiza la información del diploma en el sistema. 7. Fin del flujo
<i>Prototipo de Interfaz</i>	
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
	5.1. En caso de existir redundancia o errores en los datos, se le solicita a la secretaria

<p>5.2. La secretaria corrige los problemas señalados.</p> <p>5.3. La secretaria envía la solicitud al servidor</p>	<p>que corrija los problemas encontrados en los datos del formulario.</p> <p>5.4. El sistema ejecuta la acción 5.</p> <p>5.5. Fin del flujo alterno</p>
<i>Prototipo de Interfaz</i>	
Sección “Eliminar Diploma”	
Acción del Actor	Respuesta del Sistema
<p>1. La secretaria selecciona la opción Eliminar Diploma.</p> <p>3. La secretaria selecciona la opción deseada</p>	<p>2. El sistema muestra un mensaje de confirmación para eliminar el diploma seleccionado.</p> <p>4. En caso de haber confirmado la eliminación, el sistema elimina los datos del diploma registrado en el sistema.</p> <p>5. Fin del flujo</p>
<i>Prototipo de Interfaz</i>	
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
	<p>4.1 En caso de haber cancelado la eliminación, se termina el caso de uso.</p> <p>4.2 Fin del flujo alterno.</p>
<i>Prototipo de Interfaz</i>	
Pos condiciones	<p>1. Diploma creado en la base de datos</p> <p>2. Diploma modificado en la base de datos</p> <p>3. Diploma eliminado de la base de datos</p>

Descripción del CUS Gestionar Diplomado

Caso de Uso:	CRUD Gestionar Diplomado	
Actores:	Secretaria	
Resumen:	El caso de uso se inicia cuando la secretaria selecciona la opción Gestionar Diplomados	
Precondiciones:	1. Ser la secretaria	
Referencias		
Prioridad	Crítico	
Flujo Normal de Eventos		
Sección "Gestionar Diplomados"		
Acción del Actor	Respuesta del Sistema	
1. La secretaria selecciona la opción de Gestionar Diplomados	2. El sistema muestra las opciones: Insertar Diplomado, Modificar Diplomado y Eliminar Diplomado.	
Sección "Insertar Diplomado"		
Acción del Actor	Respuesta del Sistema	
1. La secretaria selecciona la opción de Insertar Diplomado 3. La secretaria llena el formulario con los datos del nuevo diplomado 4. La secretaria envía la solicitud al servidor.	2. El sistema muestra un formulario a llenar con los datos del diplomado a insertar. 5. El sistema verifica que no haya redundancia o errores en los datos. 6. En caso de no existir problemas, se registra el nuevo diplomado en el sistema. 7. Fin del flujo	
<i>Prototipo de Interfaz</i>		
Flujos Alternos		

Acción del Actor	Respuesta del Sistema
<p>5.2. La secretaria corrige los problemas señalados.</p> <p>5.3. La secretaria envía la solicitud al servidor</p>	<p>5.1. En caso de existir redundancia o errores en los datos, se le solicita a la secretaria que corrija los problemas encontrados en los datos del formulario.</p> <p>5.4. El sistema ejecuta la acción 5</p> <p>5.5. Fin del flujo alterno</p>
<i>Prototipo de Interfaz</i>	
Sección "Modificar Diplomado"	
Acción del Actor	Respuesta del Sistema
<p>1. La secretaria selecciona la opción Modificar Diplomado</p> <p>3. La secretaria modifica los datos del diplomado.</p> <p>4. La secretaria envía al servidor el formulario con los datos modificados.</p>	<p>2. El sistema muestra un formulario con los datos del diplomado solicitado.</p> <p>5. El sistema verifica que no haya redundancia o errores en los datos.</p> <p>6. En caso de no existir problemas, se actualiza la información del diplomado en el sistema.</p> <p>7. Fin del flujo</p>
<i>Prototipo de Interfaz</i>	
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
	<p>5.1. En caso de existir redundancia o errores en los datos, se le solicita a la secretaria</p>

5.2. La secretaria corrige los problemas señalados. 5.3. La secretaria envía la solicitud al servidor	que corrija los problemas encontrados en los datos del formulario. 5.4. El sistema ejecuta la acción 5. 5.5. Fin del flujo alterno
<i>Prototipo de Interfaz</i>	
Sección “Eliminar Diplomado”	
Acción del Actor	Respuesta del Sistema
1. La secretaria selecciona la opción Eliminar Diplomado. 3. La secretaria selecciona la opción deseada.	2. El sistema muestra un mensaje de confirmación para eliminar el diplomado seleccionado. 4. En caso de haber confirmado la eliminación, el sistema elimina los datos del diplomado registrados en el sistema. 5. Fin del flujo
<i>Prototipo de Interfaz</i>	
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
	4.1 En caso de haber cancelado la eliminación, se termina el caso de uso. 4.2 Fin del flujo alterno.
<i>Prototipo de Interfaz</i>	
Pos condiciones	1. Diplomado creado en la base de datos 2. Diplomado modificado en la base de datos 3. Diplomado eliminado de la base de datos

Descripción del CUS Gestionar Especialidad

Caso de Uso:	CRUD Gestionar Especialidad
--------------	-----------------------------

Sistema para la Secretaria Docente del Centro Politécnico del Petróleo

Actores:	Secretaria	
Resumen:	El caso de uso se inicia cuando la secretaria selecciona la opción Gestionar Especialidades.	
Precondiciones:	2. Ser la secretaria	
Referencias		
Prioridad	Crítico	
Flujo Normal de Eventos		
Sección "Gestionar Especialidades"		
Acción del Actor	Respuesta del Sistema	
1. La secretaria selecciona la opción de Gestionar Especialidades	2. El sistema muestra las opciones: Insertar Especialidad, Modificar Especialidad y Eliminar Especialidad.	
Sección "Insertar Especialidad"		
Acción del Actor	Respuesta del Sistema	
1. La secretaria selecciona la opción Insertar Especialidad.	2. El sistema muestra un formulario a llenar con los datos de la especialidad a insertar.	
3. La secretaria llena el formulario con los datos de la nueva especialidad		
4. La secretaria envía la solicitud al servidor.		
	6. En caso de no existir problemas, se registra la nueva especialidad en el sistema.	
	7. Fin del flujo	
<i>Prototipo de Interfaz</i>		
Flujos Alternos		
Acción del Actor	Respuesta del Sistema	
	5.1. En caso de existir redundancia o	

<p>5.2. La secretaria corrige los problemas señalados.</p> <p>5.3. La secretaria envía la solicitud al servidor</p>	<p>errores en los datos, se le solicita a la secretaria que corrija los problemas encontrados en los datos del formulario.</p> <p>5.4. El sistema ejecuta la acción 5</p> <p>5.5. Fin del flujo alterno</p>
<i>Prototipo de Interfaz</i>	
Sección "Modificar Especialidad"	
Acción del Actor	Respuesta del Sistema
<p>1. La secretaria selecciona la opción Modificar Especialidad</p> <p>3. La secretaria modifica los datos de la especialidad.</p> <p>4. La secretaria envía al servidor el formulario con los datos modificados.</p>	<p>2. El sistema muestra un formulario con los datos de la especialidad solicitada.</p> <p>5. El sistema verifica que no haya redundancia o errores en los datos.</p> <p>6. En caso de no existir problemas, se actualiza la información de la especialidad en el sistema.</p> <p>7. Fin del flujo</p>
<i>Prototipo de Interfaz</i>	
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
<p>5.2. La secretaria corrige los problemas</p>	<p>5.1. En caso de existir redundancia o errores en los datos, se le solicita a la secretaria que corrija los problemas encontrados en los datos del formulario.</p>

señalados. 5.3. La secretaria envía la solicitud al servidor	5.4. El sistema ejecuta la acción 5. 5.5. Fin del flujo alterno
<i>Prototipo de Interfaz</i>	
Sección "Eliminar Especialidad"	
Acción del Actor	Respuesta del Sistema
1. La secretaria selecciona una especialidad para ser eliminada. 3. La secretaria selecciona la opción deseada.	2. El sistema muestra un mensaje de confirmación para eliminar la especialidad seleccionada. 4. En caso de haber confirmado la eliminación, el sistema elimina los datos de la especialidad registrados en el sistema. 5. Fin del flujo
<i>Prototipo de Interfaz</i>	
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
	4.1 En caso de haber cancelado la eliminación, se termina el caso de uso. 4.2 Fin del flujo alterno.
<i>Prototipo de Interfaz</i>	
Pos condiciones	1. Especialidad creada en la base de datos 2. Especialidad modificada en la base de datos. 3. Especialidad eliminada en la base de datos.

Descripción del CUS Gestionar Postgrado

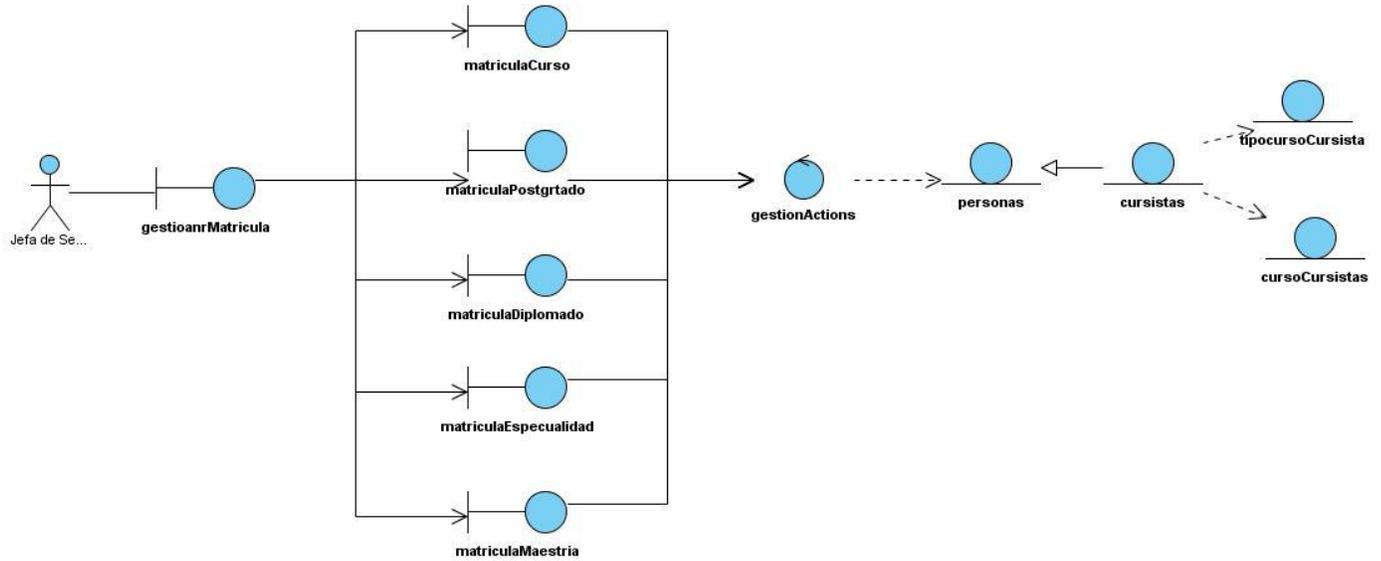
Caso de Uso:	CRUD Gestionar Postgrado	
Actores:	Secretaria	
Resumen:	El caso de uso se inicia cuando la secretaria selecciona la opción Gestionar Postgrados	
Precondiciones:	3. Ser la secretaria	
Referencias		
Prioridad	Crítico	
Flujo Normal de Eventos		
Sección "Gestionar Postgrados"		
Acción del Actor	Respuesta del Sistema	
1. La secretaria selecciona la opción de Gestionar Postgrados	2. El sistema muestra las opciones: Insertar Postgrado, Modificar Postgrado y Eliminar Postgrado.	
Sección "Insertar Postgrado"		
Acción del Actor	Respuesta del Sistema	
1. La secretaria selecciona la opción Insertar Postgrado.	2. El sistema muestra un formulario a llenar con los datos del postgrado a insertar.	
3. La secretaria llena el formulario con los datos del nuevo postgrado		
4. La secretaria envía la solicitud al servidor.		
	5. El sistema verifica que no haya redundancia o errores en los datos.	
	6. En caso de no existir problemas, se registra el nuevo postgrado en el sistema.	
	7. Fin del flujo	
<i>Prototipo de Interfaz</i>		

Flujos Alternos	
Acción del Actor	Respuesta del Sistema
<p>5.2. La secretaria corrige los problemas señalados.</p> <p>5.3. La secretaria envía la solicitud al servidor</p>	<p>5.1. En caso de existir redundancia o errores en los datos, se le solicita a la secretaria que corrija los problemas encontrados en los datos del formulario.</p> <p>5.4. El sistema ejecuta la acción 5</p> <p>5.5. Fin del flujo alternativo</p>
<i>Prototipo de Interfaz</i>	
Sección "Modificar Postgrado"	
Acción del Actor	Respuesta del Sistema
<p>1. La secretaria selecciona la opción Modificar Postgrado</p> <p>3. La secretaria modifica los datos del postgrado.</p> <p>4. La secretaria envía al servidor el formulario con los datos modificados.</p>	<p>2. El sistema muestra un formulario con los datos del postgrado solicitado.</p> <p>5. El sistema verifica que no haya redundancia o errores en los datos.</p> <p>6. En caso de no existir problemas, se actualiza la información de la postgrado en el sistema.</p> <p>7. Fin del flujo</p>
<i>Prototipo de Interfaz</i>	
Flujos Alternos	
Acción del Actor	Respuesta del Sistema

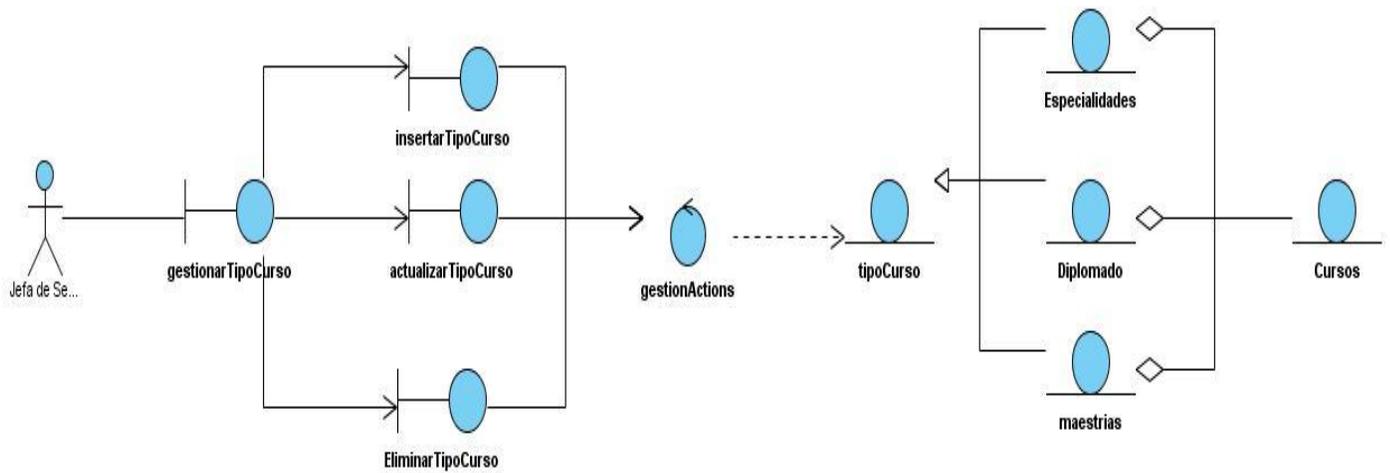
<p>5.2. La secretaria corrige los problemas señalados.</p> <p>5.3. La secretaria envía la solicitud al servidor</p>	<p>5.1. En caso de existir redundancia o errores en los datos, se le solicita a la secretaria que corrija los problemas encontrados en los datos del formulario.</p> <p>5.4. El sistema ejecuta la acción 5.</p> <p>5.5. Fin del flujo alterno</p>
<i>Prototipo de Interfaz</i>	
Sección "Eliminar Postgrado"	
Acción del Actor	Respuesta del Sistema
<p>1. La secretaria selecciona un postgrado para ser eliminada.</p> <p>3. La secretaria selecciona la opción deseada.</p>	<p>2. El sistema muestra un mensaje de confirmación para eliminar el postgrado seleccionada.</p> <p>4. En caso de haber confirmado la eliminación, el sistema elimina los datos de la postgrado registrados en el sistema.</p> <p>5. Fin del flujo</p>
<i>Prototipo de Interfaz</i>	
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
	<p>4.1 En caso de haber cancelado la eliminación, se termina el caso de uso</p> <p>4.2 Fin del flujo alterno.</p>
<i>Prototipo de Interfaz</i>	
Pos condiciones	<p>4. Postgrado creado en la base de datos</p> <p>5. Postgrado modificado en la base de datos.</p> <p>6. Postgrado eliminado en la base de datos.</p>

Anexo #3

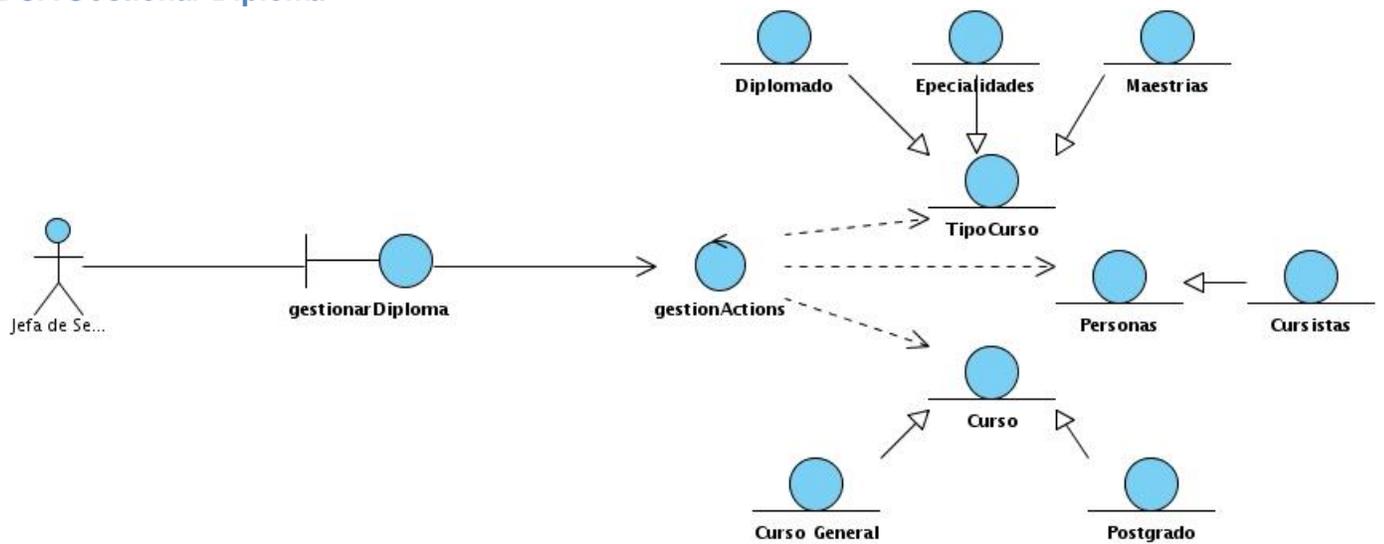
DCA Gestionar Matrícula



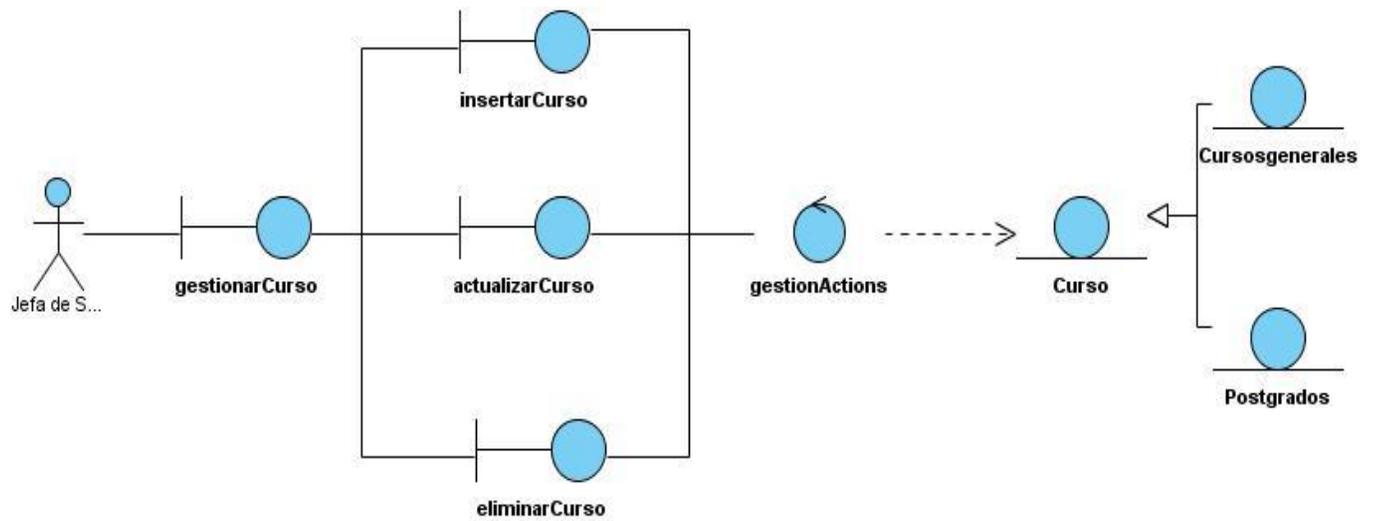
DCA Gestionar Tipo de Curso



DCA Gestionar Diploma

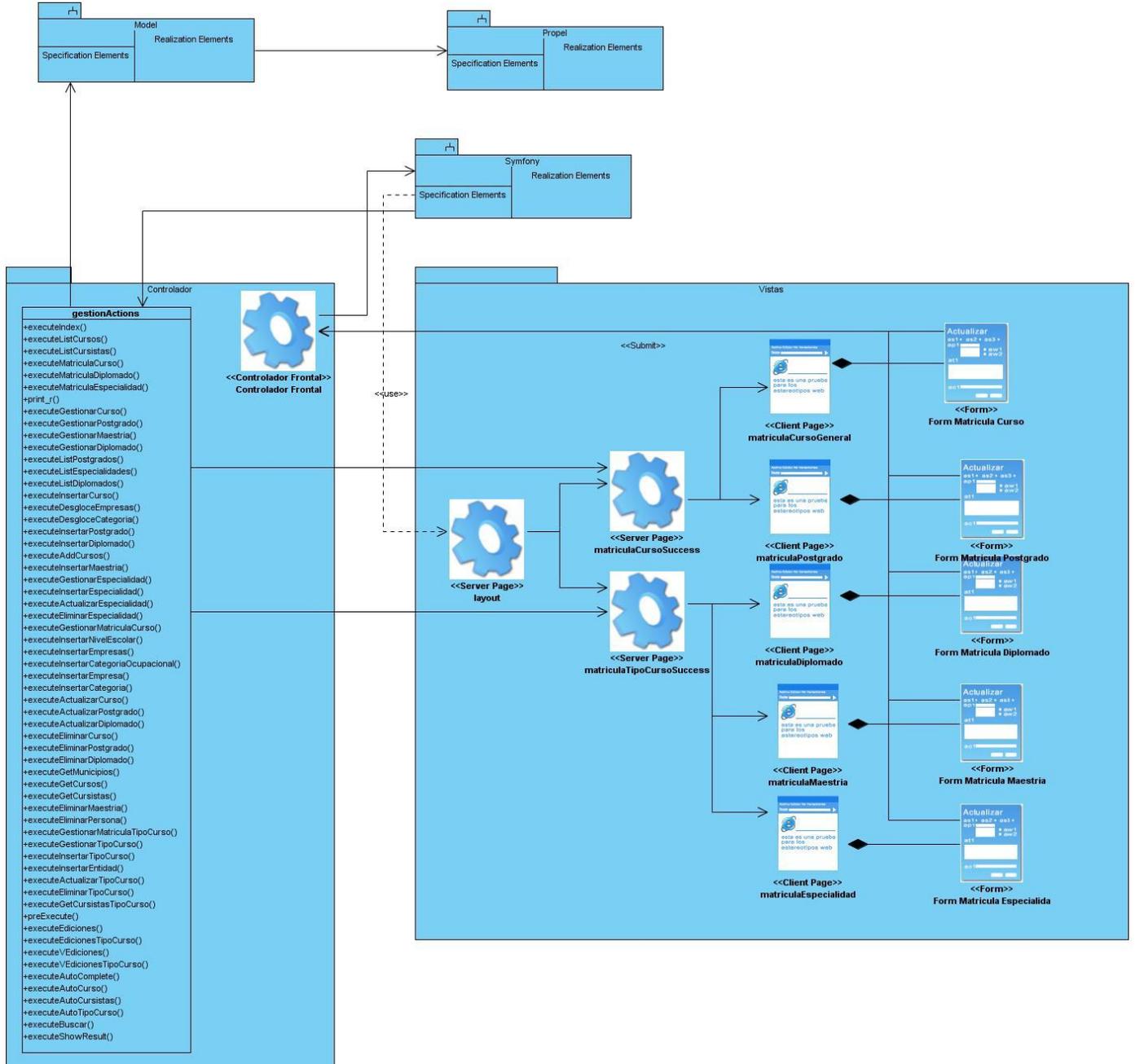


DCA Gestionar Curso

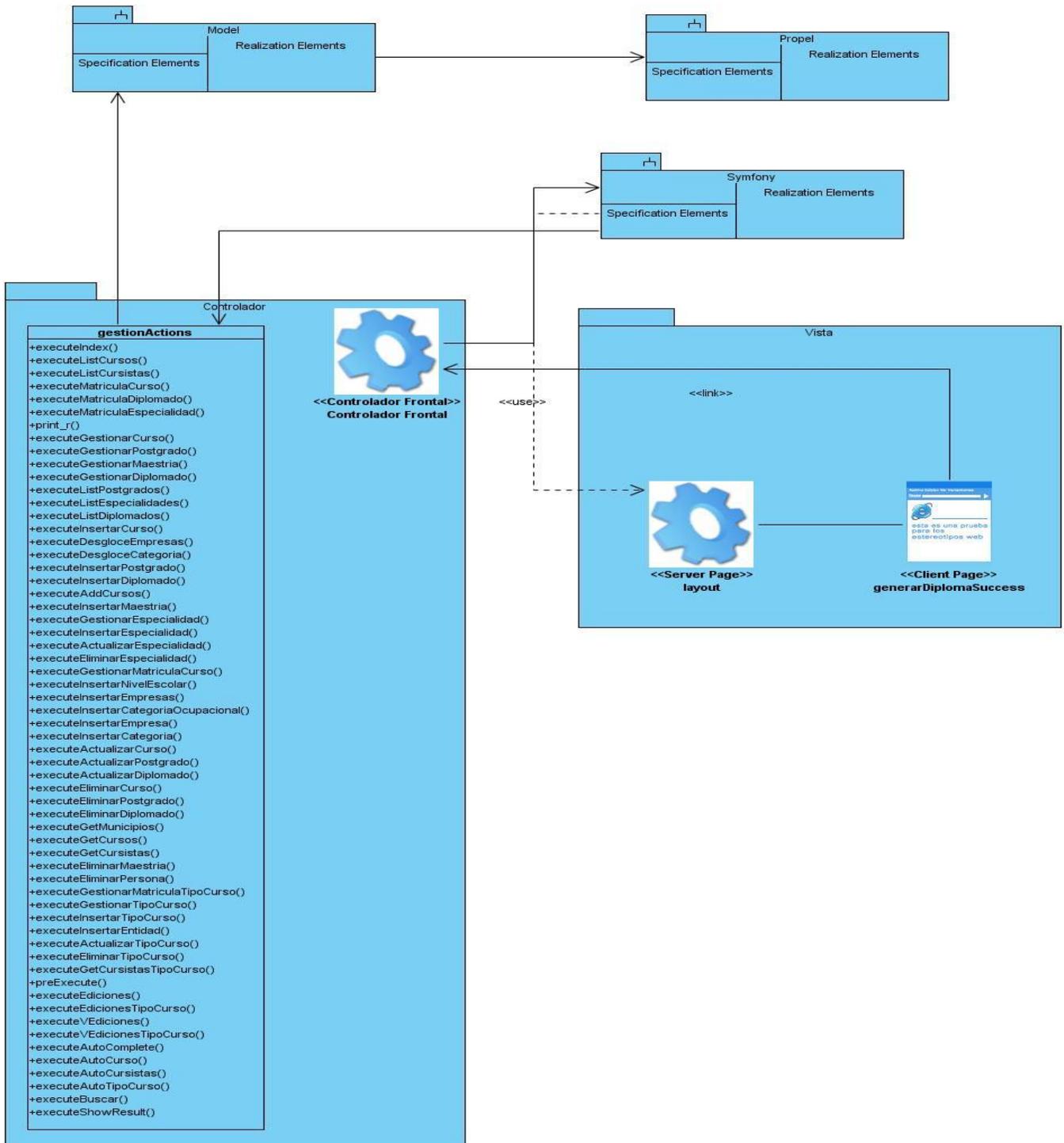


Anexo # 4

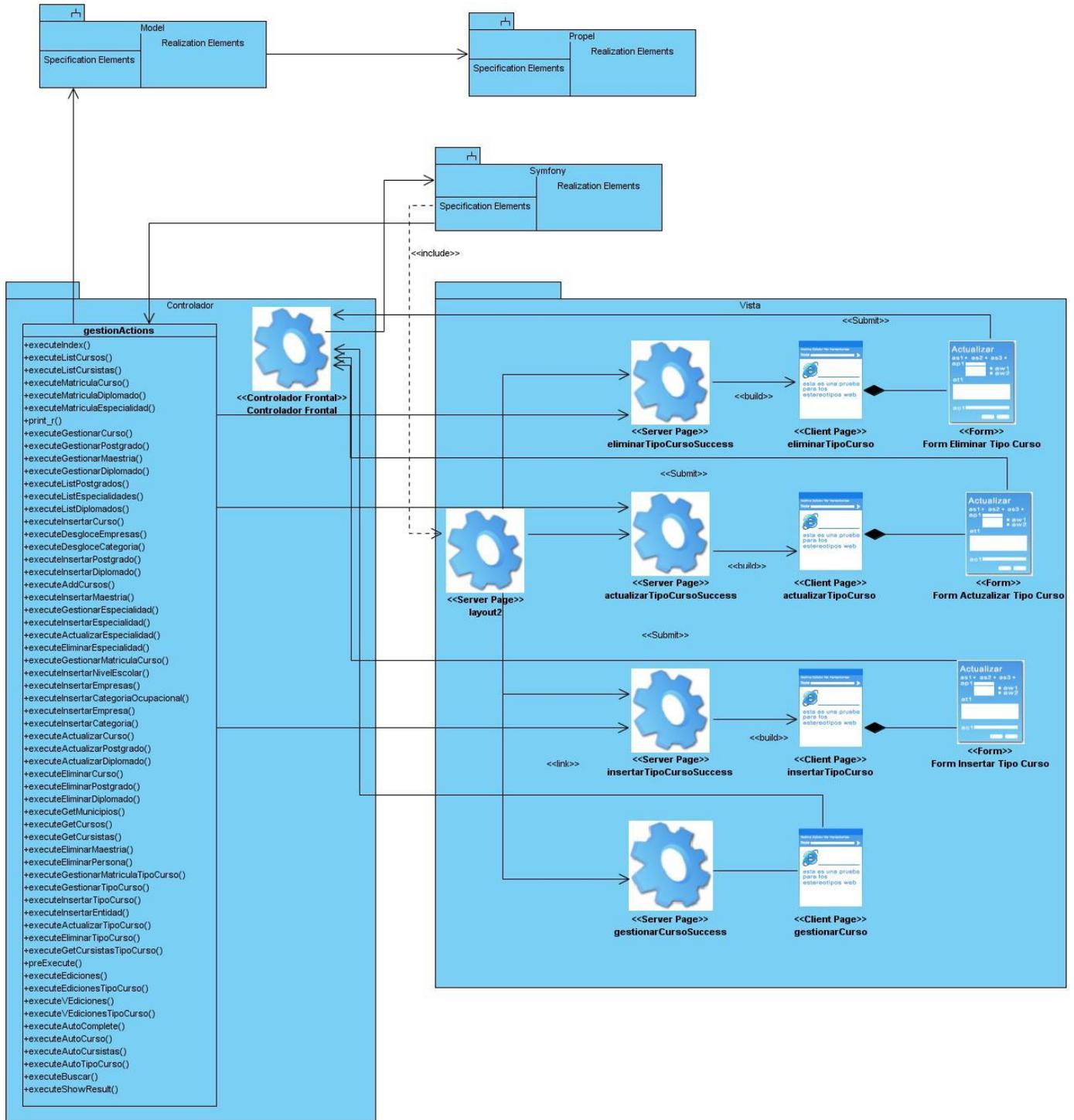
DCD Gestionar Matrícula



DCD Gestionar Diploma

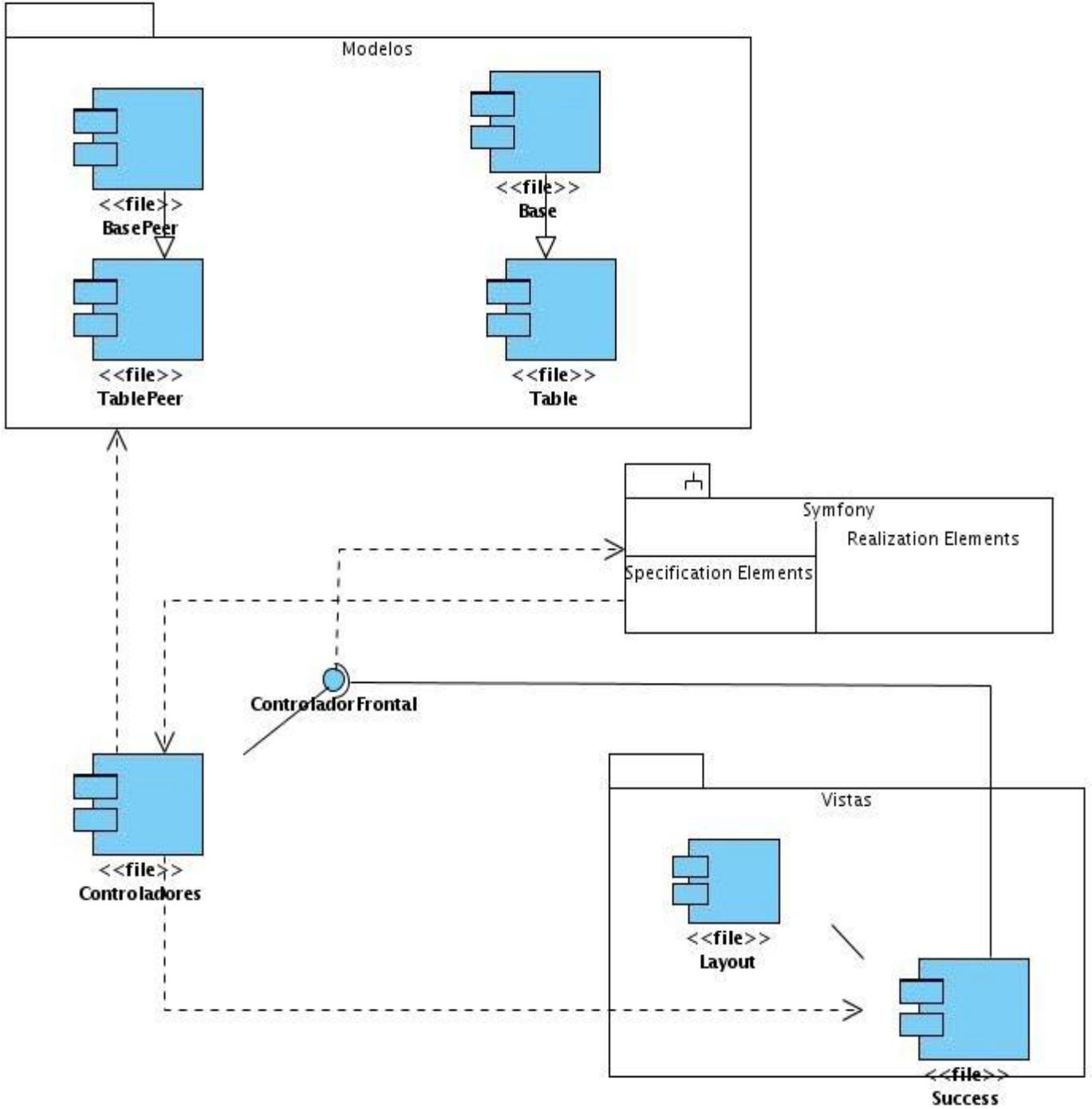


DCD Gestionar Curso



Anexo # 5

Diagrama de Componentes



Anexo # 6

Diagrama de Despliegue

