



Universidad de las Ciencias Informáticas

“Facultad 1”

Título:

**Sistema de Gestión de la Trayectoria Productiva de
cada estudiante en la facultad 1**

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

Autores: Daniuska Laurencio Pérez
Yunieski Diéguez García

Tutores: Ing. Rolando Ávila Figueredo
Ing. Roberkis Terrero Galano

Ciudad de la Habana, junio de 2009.

Declaración de autoría

Por este medio declaramos ser los únicos autores del presente trabajo y autorizamos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmo el presente a los --- días del mes de Junio del 2009.

Firma del Autor

Daniuska Laurencio Pérez

Firma del Autor

Yunieski Diéguez García

Firma del Tutor

Ing. Rolando Ávila Figueredo

Firma del Tutor

Ing. Roberkis Terrero Galano

Datos de contacto

Tutores: Ing. Rolando Ávila Figueredo.

e-mail: ravilaf@uci.cu

Ing. Roberkis Terrero Galano.

e-mail: rterrero@uci.cu

Agradecimientos

A mis padres que los adoro por darme todo su amor y apoyo...

A mi hermana que la quiero tanto...

A mi familia, mi tía Maida y las otras tías y tíos que también quiero mucho...

A mis amigos de siempre Roxana, Islenis, Yaima, Yaremis, Orestes y Niuvis, que han estado en las buenas y malas, a los otros tantos que tampoco olvidaré...

A la Revolución por darme la oportunidad de hacer realidad mi sueño...

Daniuska

A mis padres: gracias por mi vida, son los mejores...

A mi familia que es la mejor del mundo: los quiero mucho...

A mis amistades, las nuevas y las de siempre...

Yunieski

Dedicatoria

A mis padres y a mi hermana querida por ser todo en mi vida...

Daniuska

A mis padres, nadie se la merece tanto...

Yunieski

Resumen

En la actualidad la facultad 1 no cuenta con un sistema para gestionar la información referente a los estudiantes vinculados a la producción durante su estancia en la universidad, que permita llevar el control y seguimiento de forma efectiva y ágil de la trayectoria productiva, todo este proceso se realiza de forma manual, existiendo pérdida o duplicado de la información y llega a ser bastante engorroso en algunos casos encontrar un estudiante con una característica (competencia o rol) determinada. Por lo que este trabajo tiene como objetivo principal realizar el análisis, diseño e implementación de un sistema para la gestión de la trayectoria productiva de los estudiantes, que integre varios procesos, facilitando así el control de la información productiva de los mismos. Se realiza un amplio estudio del negocio, para obtener una mayor comprensión del funcionamiento de los procesos identificados y quedan plasmados de esta forma los diagramas correspondientes desde la fase de análisis hasta la implementación. Para cumplir con el objetivo principal planteado se realizó un estudio de algunas tecnologías, herramientas y lenguajes de programación más utilizados en la actualidad para el desarrollo de aplicaciones Web. Se tiene como resultado un sistema de gestión de la trayectoria productiva, el cual integra diferentes procesos que hasta el momento se realizan de forma manual, por el vicedecano de producción de la facultad y los líderes de proyectos.

Palabras Claves: Trayectoria productiva, producción, gestión.

Índice

| | |
|---|-------|
| Introducción..... | - 1 - |
| Capítulo 1 | 6 |
| 1.1 Introducción | 6 |
| 1.2 Conceptos asociados al problema..... | 6 |
| 1.2.1 Gestión..... | 6 |
| 1.2.2 Funciones de la Gestión | 7 |
| 1.2.3 Gestión de la información | 7 |
| 1.2.4 Software de gestión | 8 |
| 1.2.5 Trayectoria Productiva | 8 |
| 1.3 Estudio de Sistemas de Gestión Similares | 8 |
| 1.3.1 Sistemas de Gestión en el mundo | 8 |
| 1.3.2 Sistemas de Gestión en Cuba | 9 |
| 1.3.3 Sistemas de Gestión en la UCI..... | 10 |
| 1.3.3.1 Akademos Gestión Académica..... | 11 |
| 1.4 Metodología, herramientas y tecnologías para el proceso de desarrollo de software | 12 |
| 1.4.1 Metodologías Ágiles | 12 |
| 1.4.1.1 Extreme Programming (XP)..... | 12 |
| 1.4.1.2 Microsoft Solution Framework (MSF) | 14 |
| 1.4.2 Metodologías Tradicionales | 15 |
| 1.4.2.1 Rational Unified Process (RUP)..... | 15 |
| 1.4.3 Lenguaje Unificado de Modelado UML | 17 |
| 1.4.4 Herramientas CASE | 17 |
| 1.4.4.1 Visual Paradigm..... | 18 |
| 1.4.4.2 Rational Rose Enterprise Edition..... | 19 |
| 1.4.5 Lenguajes de programación..... | 20 |
| 1.4.5.1 PHP | 20 |
| 1.4.5.2 Java Script..... | 21 |
| 1.4.6 Sistema Gestores de Contenidos..... | 22 |
| 1.4.6.1 Joomla..... | 22 |
| 1.4.6.2 Drupal..... | 23 |
| 1.4.6.3 Plone | 24 |
| 1.4.7 AJAX | 24 |
| 1.4.8 Sistema Gestor de Base de Datos (SGBD)..... | 25 |
| 1.4.8.1 Oracle..... | 26 |
| 1.4.8.2 PostgreSQL..... | 27 |
| 1.4.8.3 MySQL | 29 |
| 1.4.9 Servidor Web | 30 |
| 1.4.9.1 Apache | 30 |

| | | |
|------------|--|----|
| 1.4.10 | Justificación de las tecnologías a usar | 31 |
| Capítulo 2 | | 33 |
| 2.1 | Introducción | 33 |
| 2.2 | Modelo de negocio | 33 |
| 2.2.1 | Descripción de los procesos del negocio y mejoras propuestas | 33 |
| 2.2.2 | Actores del Negocio..... | 35 |
| 2.2.3 | Trabajadores del Negocio..... | 35 |
| 2.2.4 | Diagrama de casos de uso del negocio | 36 |
| 2.2.5 | Descripción de los Casos de Uso del Negocio | 36 |
| 2.3 | Modelo del Sistema..... | 41 |
| 2.3.1 | Requerimientos funcionales..... | 41 |
| 2.3.2 | Requerimientos no funcionales | 42 |
| 2.3.2.1 | Requerimientos de software..... | 42 |
| 2.3.2.2 | Requerimientos de hardware | 42 |
| 2.3.2.3 | Requerimientos de apariencia o interfaz externa | 43 |
| 2.3.2.4 | Requerimientos de seguridad | 43 |
| 2.3.2.5 | Requerimientos de usabilidad..... | 43 |
| 2.3.2.6 | Requerimientos de rendimiento..... | 44 |
| 2.3.2.7 | Requerimientos de disponibilidad..... | 44 |
| 2.4 | Definición de actores del sistema | 44 |
| 2.5 | Diagrama de Casos de Uso del sistema | 45 |
| 2.6 | Descripción de los Casos de Uso del sistema..... | 46 |
| | Conclusiones..... | 51 |
| Capítulo 3 | | 52 |
| 3.1 | Introducción | 52 |
| 3.2 | Modelo de Análisis | 52 |
| 3.3 | Patrones de diseño | 57 |
| 3.3.1 | Estructura de Drupal..... | 57 |
| 3.3.1.1 | Paquete modules..... | 59 |
| 3.3.1.2 | Módulos de soporte a usar | 62 |
| 3.3.2 | Paquetes del Diseño propuestos..... | 63 |
| 3.4 | Diagrama de clases persistentes..... | 63 |
| 3.5 | Modelo de Datos..... | 64 |
| 3.5.1 | Diagrama Entidad Relación de la BD | 64 |
| 3.5.1.1 | Descripción de las tablas del modelo de datos | 66 |
| 3.6 | Patrones de diseño que se aplican en el sistema..... | 72 |
| 3.7 | Seguridad..... | 72 |
| 3.8 | Tratamiento de errores | 73 |
| | Conclusiones..... | 74 |
| Capítulo 4 | | 75 |
| 4.1 | Introducción | 75 |

| | |
|--|-----|
| 4.2 Diagrama de despliegue..... | 75 |
| 4.3 Diagrama de Componentes | 76 |
| 4.4 Modelo de Prueba | 77 |
| 4.4.1 Descripción de los casos de prueba de integración..... | 77 |
| Conclusiones..... | 79 |
| Conclusiones..... | 80 |
| Recomendaciones..... | 81 |
| Referencias bibliográficas | 82 |
| Bibliografía consultada | 82 |
| Anexos | 84 |
| Anexo I | 84 |
| Diagramas de Actividades de los casos de uso del negocio. | 84 |
| Anexo II | 88 |
| Glosario de Términos | 114 |

Índice de tablas

| | |
|--|----|
| Tabla 2.0.1 Actores del Negocio | 35 |
| Tabla 2.0.2 Trabajadores del Negocio | 35 |
| Tabla 2.0.3 Descripción del caso de uso Asignar Estudiantes a Proyectos | 38 |
| Tabla 2.0.4 Descripción del caso de uso del Negocio Solicitar Competencias | 39 |
| Tabla 2.0.5 Descripción del caso de uso del Negocio Registrar Curso Optativo | 40 |
| Tabla 2.0.6 Descripción del caso de uso del Negocio Registrar evaluaciones de Practica Profesional | 41 |
| Tabla2.0.7 Actores del Sistema..... | 44 |
| Tabla2.0.8 Descripción del caso de uso del sistema Autenticar Usuario | 47 |
| Tabla2.0.9 Descripción del caso de uso del sistema Asignar Estudiantes a Proyecto | 47 |
| Tabla 2.0.10 Descripción del caso de uso del sistema Obtener estudiantes con una determinada competencia | 48 |
| Tabla2.0.11 Descripción del caso de uso del sistema Registrar Notas de Práctica Profesional..... | 48 |
| Tabla 2.0.12 Descripción del caso de uso del sistema Obtener Notas de Práctica Profesional..... | 48 |
| Tabla 2.0.13 Descripción del caso de uso del sistema Obtener estudiantes por proyecto | 49 |
| Tabla 2.0.14 Descripción del caso de uso del sistema Obtener estudiantes por rol | 49 |
| Tabla 2.0.15 Descripción del caso de uso del sistema Asignar Competencias | 49 |
| Tabla 2.0.16 Descripción del caso de uso del sistema Registrar evento | 50 |
| Tabla 2.0.17 Descripción del caso de uso del sistema Registrar incidencias..... | 50 |
| Tabla 2.0.18 Descripción del caso de uso del sistema Registrar Cursos Optativos..... | 50 |
| Tabla 2.0.19 Descripción del caso de uso del sistema Gestionar roles a usuarios | 50 |
| Tabla 3.0.1 node | 66 |

| | |
|---|----|
| Tabla 3.0.2 user | 67 |
| Tabla 3.0.3 user_roles | 67 |
| Tabla 3.0.4 role | 68 |
| Tabla 3.0.5 access..... | 68 |
| Tabla 3.0.6 node_access | 69 |
| Tabla 3.0.7 content_type_proyecto..... | 69 |
| Tabla 3.0.8 content_type_estudiante..... | 70 |
| Tabla 3.0.9 content_type_polo | 71 |
| Tabla 3.0.10 content_type_rol | 71 |
| Tabla 3.0.11 content_type_typoevento | 71 |
| Tabla 3.0.12 content_type_competencia | 72 |
| Tabla 4.0.1 Descripción del caso de prueba asignar estudiante a proyecto..... | 77 |
| Tabla 4.0.2 Descripción del caso de prueba Registrar Nota de Práctica Profesional | 77 |
| Tabla 4.3 Descripción del caso de prueba Gestionar rol a usuarios | 78 |

Índice de figuras

| | |
|--|----|
| Figura 1 Funciones de la Gestión | 7 |
| Figura 2 Metodología Programación Extrema | 13 |
| Figura 3 Metodología MSF | 14 |
| Figura 4 Diagrama de casos de uso del negocio | 36 |
| Figura 5 Diagrama de Casos De Uso del Sistema..... | 46 |
| Figura 6 DCA CUS Autenticar Usuario | 53 |
| Figura 7 DC CUS Autenticar Usuario..... | 53 |
| Figura 8 DCA CUS Asignar Estudiante..... | 54 |
| Figura 9 DC CUS Asignar Estudiante | 54 |
| Figura 10 DCA CUS Registrar Nota de Práctica Profesional | 55 |
| Figura 11 DC CUS Registrar Nota de Práctica Profesional..... | 55 |
| Figura 12 DC CUS Obtener Nota de Práctica Profesional..... | 56 |
| Figura 13 DC CUS Obtener Estudiantes x Rol | 56 |
| Figura 14 Diagrama de diseño Web de Drupal..... | 59 |
| Figura 15 Diagrama de clases sub-paquete Modules | 61 |
| Figura 16 Diagrama de Clases Persistentes..... | 64 |
| Figura 17 Diagrama Entidad Relación..... | 65 |
| Figura 18 Diagrama de Despliegue | 76 |
| Figura 19 Diagrama de Componentes | 76 |

Introducción

El desarrollo de la sociedad actual está estrechamente relacionado con el avance de las tecnologías y al mismo tiempo se hacen más dependientes de los medios informáticos, es por eso, que cada día el uso de las computadoras es más necesario en el mundo que nos rodea, y así mismo la informatización de numerosas actividades y procesos que antes se realizaban de forma manual. Todo esto en aras de lograr mayor calidad, agilidad y seguridad en los datos. Nuestro país no ha quedado exento de estos formidables cambios que tienen lugar tanto en los países desarrollados como en los que poseen menos adelanto. En Cuba la mayor institución creada con el objetivo de lograr avances en la rama de la informática es la Universidad de las Ciencias Informáticas (UCI), universidad estructurada en diez facultades cada una con su área de producción en las cuales existen Polos Productivos vinculados a variadas esferas de la sociedad, tales como: turismo, bioinformática, salud, comunicaciones, procesamiento de imágenes, FAR, INDER, entre otros.

En la facultad 1 se desarrollan varios proyectos en los cuales están implicados un gran número de estudiantes, de los cuales se hace necesario gestionar información relacionada a toda la trayectoria productiva, dígame roles que ha ocupado en el proyecto del cual es miembro, cursos optativos que se le hayan acreditado, participación en eventos científicos y si se han dado de baja o alta en un proyecto determinado. En la actualidad no existe una aplicación que maneje la información referente a los datos productivos de los estudiantes, todo este proceso se realiza en la Facultad de forma manual, almacenando los datos en documentos físicos, resultando en algunos casos engorrosa la manipulación de los datos, ya sea una búsqueda, modificación o cualquier otra actividad; afectándose así la agilidad y calidad del proceso, esto trae consigo que:

- Exista duplicados o ausencia en algunos casos de la información.
- Dificultad a la hora de asignar un estudiante a un proyecto productivo.
- La búsqueda y recuperación de la información a la hora de hacer reportes es lenta.

- Ausencia de un sistema integrado que permita gestionar toda la información referente a un estudiante en el área productiva.

Condición por la cual se hace necesaria elaborar un sistema informático capaz de perfeccionar y hacer más eficiente el proceso antes mencionado.

Por tal situación se plantea como **problema a resolver**: ¿Cómo gestionar la información referente a la trayectoria productiva de los estudiantes en la facultad 1?

Se define como **objeto de estudio**: los Procesos de gestión de trayectoria estudiantil en la producción de software, donde el **campo de acción** estaría enmarcado en los procesos de acreditación de cursos optativos, participación en eventos y gestión de roles desde la producción de Software.

Para darle solución al problema existente surge como **objetivo general**: Implementar un sistema que permita la gestión de la información de la trayectoria productiva de cada estudiante en la facultad 1. Del objetivo general se derivan los siguientes **objetivos específicos**:

- Documentar la situación actual del proceso de gestión de la trayectoria productiva de los estudiantes en la facultad 1.
- Definir los requerimientos funcionales del cliente con respecto a la gestión de la información de los estudiantes vinculados a proyectos productivos.
- Evaluar y proponer las posibles herramientas a utilizar en el análisis, diseño e implementación del sistema.
- Realizar el Análisis, Diseño e Implementación del Sistema de gestión de la trayectoria productiva de los estudiantes.

La investigación se sustenta en la siguiente **idea a defender**: El desarrollo de una aplicación web que gestione la información de la trayectoria productiva de los estudiantes de la facultad 1, facilitará el trabajo de almacenamiento y gestión de datos referentes a la trayectoria productiva llevada a cabo por la dirección de producción de la facultad.

Para dar cumplimiento al trabajo se hace necesario las siguientes **tareas**:

- Estudio de la documentación existente sobre la trayectoria productiva de los estudiantes en la facultad 1.
- Entrevista a los líderes de proyectos y directivos involucrados en el área productiva de la facultad 1.
- Análisis y selección de las herramientas que permitan obtener una solución óptima.
- Caracterización de la metodología de desarrollo de software para el modelado del sistema.
- Análisis y diseño de la aplicación a desarrollar.
- Implementación de la aplicación a desarrollar.
- Estudio de las pruebas que se le deben realizar al sistema para comprobar su seguridad.

Como **posibles resultados** del presente trabajo está:

- Se logra obtener un sistema de Gestión de la trayectoria productiva que permite:
 - Registrar las evaluaciones de los cursos optativos vencidos.
 - Almacenamiento de los resultados obtenidos en los trabajos científicos presentados en eventos.
 - La asignación de un estudiante a un proyecto y su rol dentro del mismo.
 - Registrar y obtener las competencias alcanzadas por los estudiantes vinculados a proyectos.
 - Presentación de una interfaz amigable para el usuario que haga uso del mismo.
- Confiabilidad y seguridad en los datos que se gestionen.
- Reducción de costos económicos en materiales de oficina (hojas, uso de impresoras, etc.).
- Un método novedoso para la gestión de la información, ya que hasta el momento este proceso no está informatizado.

En el proceso de desarrollo de la investigación científica se emplearon **Métodos** Teóricos y Empíricos. Como métodos teóricos se usaron: Análisis Histórico-Lógico donde se estudian las formas de solución dadas de años anteriores a problemas similares al planteado sobre el proceso de gestión de la trayectoria productiva de los estudiantes vinculados a proyectos. Además, se usó

el método Hipotético-Deductivo en el cual se analizaron otras hipótesis parecidas en las que se proponían como procedimiento una solución informática para mejorar la gestión de procesos similares y se llegó a la conclusión de que esto podría solucionar el problema planteado.

Dentro de los métodos empíricos usados están: la observación donde se realizó una serie de observaciones de cómo se realiza la gestión de la trayectoria productiva de los estudiantes de la facultad 1 de la UCI y todos los problemas relacionados con este proceso. Y la entrevista donde se efectuaron entrevistas a los directivos de la facultad y líderes de proyectos con el objetivo de profundizar en el problema y entender sus especificidades para obtener la solución óptima.

El contenido esta estructurado de la siguiente forma:

Capítulo I: Capítulo en el cual se realiza un estudio de otros sistemas para darle solución a problemas similares al planteado. Se mencionan cada una de las tecnologías, metodologías y herramientas usadas para el desarrollo de la aplicación.

Capítulo II: Identificación de los procesos del negocio, así como las reglas que lo rigen; también se describen los casos de uso, actores y trabajadores del mismo. Se especifican las funcionalidades que el sistema una vez realizado debe ser capaz de efectuar. Además se definen y detallan los casos de uso y actores del sistema.

Capítulo III: Se realiza el modelado de los diagramas de clases y de colaboración pertenecientes al análisis; así como el diagrama del diseño Web con la pertinente descripción de los paquetes que lo conforman. También en dicho capítulo se encuentran presentes el diagrama de clases persistentes y el modelo de datos relacional de la base datos en el cual se almacena toda la información que se procesa y finalmente se describen las tablas que componen este último.

Capítulo IV: Recoge aspectos que definen la implementación a partir del diagrama de despliegue y el diagrama de componentes y se determinan las pruebas a realizar en el

sistema para verificar su integridad y si se ajusta a los requerimientos planteados por el cliente y se plantean posibles mejoras al sistema.

Capítulo 1

Fundamentación Teórica.

1.1 Introducción

La utilización de las tecnologías de la informática y las comunicaciones (TIC) se ha generalizado debido al gran avance que ha tenido la sociedad de la información, la creación, modificación y distribución de la información forman un eslabón de suma importancia para el avance y bienestar de la humanidad. Debido al gran impacto que ha tenido en todos los ámbitos, cada vez es más difícil lograr eficiencia al prescindir de las tecnologías por lo que es indispensable para el progreso de toda la sociedad. La utilización de las TIC hace posible que la gestión de la información se genere de forma más eficaz y rápida las peticiones de reportes que satisfagan las necesidades de los usuarios en el área que se desarrollen. En el presente capítulo explican los principales conceptos asociados al problema, se realiza una investigación de algunos sistemas similares en la universidad. Luego de realizar un amplio estudio de las herramientas y tecnologías más usadas se define cuales serán las más adecuadas para obtener un óptimo resultado.

1.2 Conceptos asociados al problema

1.2.1 Gestión

Gestión: proceso mediante el cual se obtiene, despliega o utiliza una variedad de recursos básicos para apoyar los objetivos de la organización.

1.2.2 Funciones de la Gestión



Figura 1 Funciones de la Gestión

Planificar: Proceso de establecer objetivos con el fin de alcanzar determinados resultados. (Establecimiento de objetivos, elaboración de planes, etc.)

Organizar: Proceso de dividir el trabajo y de coordinar el logro de resultados que tienen un propósito común.

Dirigir: Proceso de conducir y coordinar esfuerzos laborales de las personas que integran una organización. Función mediante la cual se ponen en marcha las tareas programadas.

Controlar: Proceso de supervisar las actividades y resultados, comparándolos con los objetivos y tomando las acciones correctivas, si son necesarias.

1.2.3 Gestión de la información

Gestión de la información se puede definir como un conjunto de actividades realizadas con el fin de controlar, almacenar y posteriormente recuperar adecuadamente la información producida, recibida o retenida por cualquier organización en el desarrollo de sus actividades, en el centro de la gestión de la información se encuentra la gestión de la documentación, así como la repartida en bases de datos corporativas y aplicaciones informáticas.

1.2.4 Software de gestión

Las aplicaciones o software de gestión son aquellas diseñadas para sustituir uno o varios procedimientos, tanto comerciales como administrativos, que habitualmente realiza una persona en una empresa o institución de forma presencial, por un software, que permita realizar al cliente los mismos procedimientos de forma no presencial o disminuir el esfuerzo empleado para los mismos.

1.2.5 Trayectoria Productiva

La trayectoria productiva de un estudiante es el comportamiento del mismo a lo largo de su estancia en la universidad vinculado al área de producción, aquí se incluye el proyecto al cual pertenece, el rol o los roles que ocupa, así como las competencias alcanzadas por el mismo dentro del proyecto. Además, se tiene en cuenta los eventos referentes a la producción y la investigación en que participa, los cursos optativos acreditados dentro del proyecto y como característica más importante que define la evaluación del estudiante en la producción se tiene la nota de Práctica Profesional. Todo esto se tiene controlado desde el momento que es estudiante se vincula al área de producción hasta el último momento que esté ubicado en un proyecto productivo.

1.3 Estudio de Sistemas de Gestión Similares

1.3.1 Sistemas de Gestión en el mundo

RRHH (Recursos Humanos)

RRHH es una línea de productos de software argentino que brindan una solución integral y definitiva para las necesidades de las empresas que deseen optimizar los procesos de selección y búsqueda de personal, para lograr rapidez, mayor efectividad, menores costos y mejorar la imagen empresarial. ¿Qué se logra con RRHH? Capturar curriculums vía internet, email, o ingresarlos manualmente en la base de datos. Archivar ordenadamente información de sus empleados y verlos en un formato estándar. Realizar búsquedas por cualquier combinación de

datos que hayan sido ingresados (formación, sexo, edad, etc.). Puede realizar el seguimiento de las entrevistas de los candidatos. RRHH está conformado por dos productos que satisfacen distintas necesidades y que juntos conforman la herramienta más completa del mercado de Recursos Humanos [1].

Los productos que conforman el paquete son:

- Recursos Humanos para Búsquedas: Un programa único y exclusivo de búsqueda inteligente y seguimiento de información muy importante en el mercado.
- Recursos Humanos para Web: Incluye todo lo necesario para que las personas interesadas en participar de búsquedas de empleo en su empresa ingresen los currículos a través del sitio en internet de su Organización.

Ventajas Concretas: Brinda información procesada en forma totalmente sistematizada y estandarizada. Ofrece infinitos parámetros o criterios de selección, color de ojos, estatura, con auto, con moto, sin experiencia. En segundos el programa brinda los resultados de una búsqueda compleja. La respuesta al cliente externo o interno es rápida en tiempo y minuciosa en cuanto a contenido. Cualquier persona dentro de la empresa puede acceder a los datos, con previa clave de autorización. Permite el seguimiento de las personas entrevistadas, con comentarios y evaluaciones, inclusive dentro de las distintas áreas de la empresa. Se evita la acumulación de papeles.

1.3.2 Sistemas de Gestión en Cuba

Gestor de Mantenimiento Integral (GMI)

GMI es un software para la organización, planificación y control de la Gestión del Mantenimiento, creado con el objetivo de brindar una herramienta que permita realizar estas funciones de manera más rápida y efectiva, soportado en ambiente Windows, amigable y sencillo en su operación, da la posibilidad de generar una variedad de reportes que pueden ser utilizados en la toma de decisiones y para la evaluación del desempeño productivo y económico en el área de

Mantenimiento. Este software permite establecer un flujo de información coordinado entre el área de mantenimiento y el resto de las áreas de la empresa. Dotado de una alta flexibilidad permite una relación armoniosa entre la planificación productiva, el área de aseguramiento y otras áreas de la empresa con la actividad de mantenimiento. Diseñado específicamente para un uso sencillo, para minimizar el proceso de entrada de datos inicial y facilitar su operación diaria [2].

Algunas de las ventajas que proporciona son:

- Almacenar grandes volúmenes de información escrita y gráfica de todos los medios y las instalaciones productivas de la empresa, así como de los trabajos realizados en estos, disminuyendo los niveles de archivos.
- Realizar toda la programación de mantenimiento, de los recursos materiales y humanos.
- Facilidad y rapidez en la emisión de las Ordenes de Trabajo
- Disminuir los tiempos y el personal dedicado a la actividad de planificación y control de mantenimiento.
- Emitir reportes e informaciones periódicas para su análisis técnico y definición de estrategias para mejorar el desempeño de la actividad de mantenimiento.

1.3.3 Sistemas de Gestión en la UCI

Actualmente se vive en la etapa donde la humanidad ha alcanzado un desarrollo imprevisible; cada día son mayores las diferencias sociales, políticas y económicas. La información es un elemento fundamental para el desarrollo. Con el de cursar de los años, la gestión de la información ocupa, cada vez más, un espacio mayor en la economía de los países a escala mundial. Con el surgimiento de la teoría de la organización, se acentuó la importancia de la información. El impacto de los cambios económicos, políticos, culturales, tecnológicos y otros ha originado una revolución en materia de gestión de información en las organizaciones. Los servicios de información, como parte esencial de la infraestructura para la gestión del conocimiento, suministran información, impulsan la generación del conocimiento para la búsqueda de soluciones a los problemas que enfrentan las organizaciones, analizan su impacto

sobre los resultados de las empresas e influyen en el comportamiento de los individuos ante la información.

Al calor de la Batalla de Ideas ha surgido en nuestro país un nuevo programa revolucionario, la Universidad de las Ciencias Informáticas, con el objetivo de formar y preparar a jóvenes estudiantes en Ingenieros Informáticos que más tarde serán la clave esencial para informatizar al país. El desarrollo tecnológico está sujeto a cambios constantemente y nuevas son las herramientas que día a día son creadas para perfeccionar el manejo de la información. La universidad es una organización que presta servicios de información a los profesores, estudiantes y trabajadores, diseñados de acuerdo a las necesidades de cada uno. La cual comprende el proceso de enseñanza-aprendizaje a partir de una actitud abierta al cambio. Como parte esencial de la gestión del conocimiento brinda servicios de información para la búsqueda de soluciones a los problemas que enfrenta la comunidad informática. La universidad busca asegurar que se disponga de la información y las capacidades necesarias para su adaptación a continuos cambios internos y externos del centro. La Dirección de Información abre las puertas del conocimiento cada día para brindar servicios que apoyarán las actividades investigativas, académicas y de entretenimiento.

Muchos son los sistemas de gestión que se ponen en práctica en la universidad, los cuales han dado excelentes servicios, entre ellos se encuentra:

1.3.3.1 Akademos Gestión Académica

Sistema de Gestión Académica es una herramienta que mantiene informado a los estudiantes sobre su desempeño académico y lleva el control del desarrollo del proceso docente, notas y asistencia. Además de la definición de los planes de estudio. Asignaturas, disciplinas y perfiles. Esta herramienta se ha convertido en una referencia para resolver los problemas de esta índole. Pero tiene la desventaja que se basa totalmente en la plataforma .Net y su sistema gestor de base de datos es SQLServer, ambas tecnologías propietarias descartadas para el desarrollo en la UCI.

1.4 Metodología, herramientas y tecnologías para el proceso de desarrollo de software

Debido a que la facultad 1 no cuenta con un sistema que le permita gestionar la trayectoria productiva de los estudiantes vinculados a proyectos productivos, se ha decidido implementar una aplicación que disponga de las funcionalidades fundamentales para garantizar una respuesta adecuada de los procesos que se llevan a cabo. En aras de desarrollar un sistema que tenga la calidad y eficiencia requeridas se utilizaron las siguientes herramientas, tecnologías y metodología para su implementación.

1.4.1 Metodologías Ágiles

Durante los últimos años han surgido las llamadas metodologías ágiles. Las cuales aportan nuevas técnicas y métodos de trabajo para el desarrollo de cada etapa de un software. En general estas metodologías hacen un balance entre los procesos y el esfuerzo, ya que tratan de centrarse en las cuestiones necesarias sin perderse en las burocráticas. Además tienen como prioridad satisfacer al cliente mediante tempranas y continuas entregas de software que le aporte un valor, capturan los cambios para que el cliente tenga una ventaja competitiva, construir el proyecto en torno a individuos motivados y lograr que el personal del negocio y los desarrolladores trabajen juntos a lo largo del proyecto, entre otras. Dentro de las más mencionadas se encuentran *Extreme Programming (XP)* y *Microsoft Solution Framework (MSF)*.

1.4.1.1 Extreme Programming (XP)

XP es una de las metodologías de desarrollo de software más exitosa en la actualidad, utilizadas para proyectos de corto plazo y corto equipo, donde cuyo plazo de entrega es muy corto. La metodología consiste en una programación rápida o extrema, cuya particularidad es tener como parte del equipo, al usuario final, pues es uno de los requisitos para llegar al éxito del proyecto.

En la figura 2 se puede observar la metodología XP.

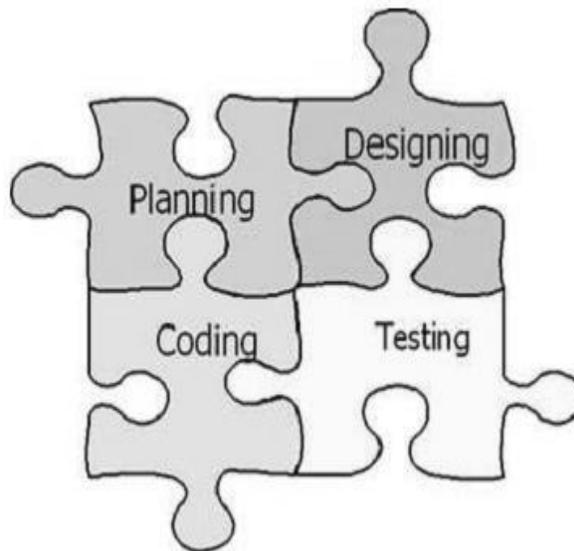


Figura 2 Metodología Programación Extrema

La metodología *Extreme Programming* (XP) se basa en las Pruebas Unitarias, la Refabricación y la Programación en pares como características fundamentales. Además XP convierte a los clientes o usuarios en miembros del equipo. Propone que los clientes tienen el derecho de decidir que se implementa, de saber el estado real del proyecto, su progreso, de añadir, cambiar o quitar requerimientos en cualquier momento, como también obtener lo máximo de cada semana de trabajo y de un sistema funcionando cada 3 o 4 meses. XP plantea derechos que posee el desarrollador como son: el decidir como se implementan los procesos, crear el sistema con la mejor calidad posible, pedir al cliente en cualquier momento aclaraciones de los requerimiento, estimar el esfuerzo para implementar el sistema y cambiar los requerimientos en base a nuevos descubrimientos. Lo fundamental de este tipo de metodología es la comunicación, entre los usuarios y los desarrolladores, así como la simplicidad, al desarrollar y codificar los módulos del sistema, además de la retroalimentación concreta y frecuente del equipo de desarrollo, el cliente y los usuarios finales. Mientras que RUP intenta reducir la complejidad del software por medio de estructura y la preparación de las tareas pendientes en función de los objetivos de la fase y actividad actual, XP, como toda metodología ágil, lo intenta por medio de un trabajo orientado directamente al objetivo, basado en las relaciones interpersonales y la velocidad de reacción.

1.4.1.2 Microsoft Solution Framework (MSF)

MSF es una metodología flexible e interrelacionada con una serie de conceptos, modelos y prácticas de uso, que controlan la planificación, el desarrollo y la gestión de proyectos tecnológicos. Se centra en los modelos de proceso y de equipo dejando en un segundo plano las elecciones tecnológicas.

En la figura 3 se puede observar la metodología MSF.

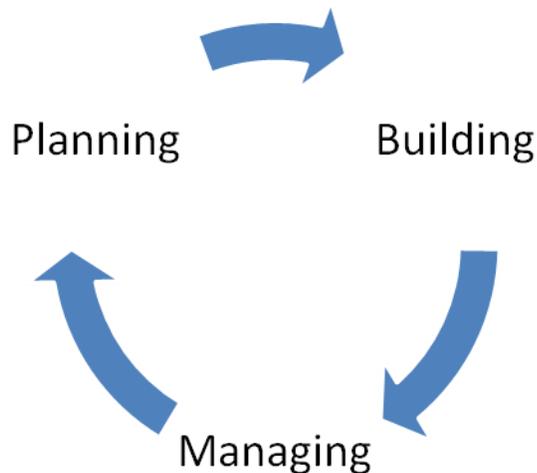


Figura 3 Metodología MSF

Esta metodología presenta características como: el ser una tecnología agnóstica, adaptable, escalable y flexible. A continuación se explican brevemente.

1. Adaptable: es parecido a un compás, usado en cualquier parte como un mapa, del cual su uso es limitado a un específico lugar.
2. Escalable: puede organizar equipos tan pequeños entre 3 o 4 personas, así como también, proyectos que requieren 50 personas o más.
3. Flexible: es utilizada en el ambiente de desarrollo de cualquier cliente.
4. Tecnología Agnóstica: porque puede ser usada para desarrollar soluciones basadas sobre cualquier tecnología.

1.4.2 Metodologías Tradicionales

Las Metodologías Tradicionales imponen una disciplina de trabajo sobre el proceso de desarrollo del software, haciendo énfasis en la previa y total planificación del proyecto a desarrollar. Una vez detallado todo el trabajo a realizar es que comienza el ciclo de desarrollo del software. Todo lo anteriormente planteado se realiza con el objetivo de conseguir un software más eficiente y predecible. RUP es la metodología más conocida dentro de esta clasificación. Mundialmente es muy utilizada, al igual que en la Universidad (UCI).

1.4.2.1 Rational Unified Process (RUP)

La metodología RUP es el resultado de varios años de desarrollo y uso práctico en el que se han unificado técnicas de desarrollo, a través del UML, y trabajo de muchas metodologías utilizadas por los clientes. La versión que se ha estandarizado vio la luz en 1998 y se conoció en sus inicios como Proceso Unificado de Desarrollo 5.0; de ahí las siglas con las que se identifica a este proceso de desarrollo. Como proceso en sí, define como sus elementos más importantes: (a) los trabajadores, que indican el “quién”. Define el comportamiento y responsabilidades (rol) de un individuo, grupo de individuos, sistema automatizado o máquina, que trabajan en conjunto como un equipo. (b) Actividades, que definen el “cómo”, una tarea que tiene un propósito claro, es realizada por un trabajador y manipula elementos. (c) Los artefactos, que no son más que el “qué”, productos tangibles del proyecto que son producidos, modificados y usados por las actividades (modelos, elementos dentro del modelo, código fuente y ejecutables). Por último el Flujo de actividades, que identifica el “Cuándo”, secuencia de actividades realizadas por trabajadores y que produce un resultado de valor observable. RUP, tiene como característica fundamental en su ciclo de vida ser:

- Dirigido por casos de uso.
- Centrado en la arquitectura.
- Iterativo e Incremental.

Divide en 4 fases el desarrollo del software:

- Inicio: El Objetivo en esta etapa es determinar la visión del proyecto.

- Elaboración: En esta etapa el objetivo es determinar la arquitectura óptima.
- Construcción: En esta etapa el objetivo es llevar a obtener la capacidad operacional inicial.
- Transición: El objetivo es llegar a obtener el realce del proyecto. Cada una de estas etapas es desarrollada mediante el ciclo de iteraciones, la cual consiste en reproducir el ciclo de vida en cascada a menor escala. Los objetivos de una iteración se establecen en función de la evaluación de las iteraciones precedentes.

El ciclo de vida que se desarrolla por cada iteración, es llevado bajo dos disciplinas:

Disciplina de Desarrollo:

- Ingeniería de Negocios: Entendiendo las necesidades del negocio.
- Requerimientos: Traslado de las necesidades del negocio a un sistema automatizado.
- Análisis y Diseño: Traslado de los requerimientos dentro de la arquitectura de software.
- Implementación: Creando software que se ajuste a la arquitectura y que tenga el comportamiento deseado.
- Pruebas: Asegurándose que el comportamiento requerido es el correcto y que todo lo solicitado está presente.

Disciplina de Soporte:

- Configuración y administración del cambio: Guardando todas las versiones del proyecto.
- Administrando el proyecto: Administrando horarios y recursos.
- Ambiente: Administrando el ambiente de desarrollo.
- Distribución: Hacer todo lo necesario para la salida del proyecto.

Los elementos del RUP son:

- Actividades: Son los procesos que se llegan a determinar en cada iteración.
- Trabajadores: Vienen hacer las personas o entes involucrados en cada proceso.
- Artefactos: Un artefacto puede ser un documento, un modelo, o un elemento de modelo.

RUP es una de las metodologías más importantes para alcanzar un grado de certificación en el desarrollo del software dado por la particularidad de que en cada ciclo de iteración, se hace exigente el uso de artefactos.

1.4.3 Lenguaje Unificado de Modelado UML

Lenguaje Unificado de Modelado. Es uno de los más conocidos y utilizados en la actualidad; aún cuando todavía no es un estándar oficial, está respaldado por el OMG (*Object Management Group*). Ofrece un estándar para describir un plano del sistema (modelo), incluyendo aspectos conceptuales tales como procesos de negocios y funciones del sistema y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y componentes de software reutilizables. Es importante resaltar que UML es un lenguaje para especificar, construir, visualizar y documentar los artefactos de un sistema de software Orientado a Objetos (OO) y no para describir métodos o procesos. Se puede aplicar en una gran variedad de formas para dar soporte a una metodología de desarrollo de software (tal como el RUP), pero no especifica en sí mismo qué metodología o proceso usar. Es una técnica de modelado de objetos y como tal supone una abstracción de un sistema para llegar a construirlo en términos concretos.

1.4.4 Herramientas CASE

Las Herramientas CASE (*Computer Aided Software Engineering*, Ingeniería de Software Asistida por Ordenador) son diversas aplicaciones informáticas destinadas a aumentar la productividad en el desarrollo de software reduciendo el costo de las mismas en términos de tiempo y de dinero. Estas herramientas pueden ayudar en todos los aspectos del ciclo de vida de desarrollo del software en tareas como el proceso de realizar un diseño del proyecto, cálculo de costes, implementación de parte del código automáticamente con el diseño dado, compilación automática, documentación o detección de errores, entre otras. Estas herramientas tienen como objetivos:

- Mejorar la productividad en el desarrollo y mantenimiento del software.
- Aumentar la calidad del software.
- Mejorar el tiempo y coste de desarrollo y mantenimiento de los sistemas informáticos.
- Mejorar la planificación de un proyecto
- Aumentar la biblioteca de conocimiento informático de una empresa ayudando a la búsqueda de soluciones para los requisitos.

- Garantizar una correcta documentación, generación de código, pruebas de errores y gestión del proyecto.
- Ayuda a la reutilización del software, portabilidad y estandarización de la documentación.
- Gestión global en todas las fases de desarrollo de software con una misma herramienta.
- Facilitar el uso de las distintas metodologías propias de la ingeniería del software.

Dentro de la lista de aplicaciones CASE cabe destacar una muy conocida, ella es el Visual Paradigm para UML, que cuenta con una versión gratuita denominada *Community Edition*.

1.4.4.1 Visual Paradigm

Los procesos de análisis y diseño del presente trabajo se desarrollaron en Visual Paradigm. Visual Paradigm para UML es una de las herramientas CASE, considerada muy completa y fácil de usar, con soporte multiplataforma y que proporciona excelentes facilidades de interoperabilidad con otras aplicaciones. Fue creada para el ciclo vital completo del desarrollo del software que lo automatiza y acelera, permitiendo la captura de requisitos, análisis, diseño e implementación. Es muy sencilla de usar, fácil de instalar y actualizar. Genera código para varios lenguajes.

También proporciona características tales como generación del código, ingeniería inversa y generación de informes. Tiene la capacidad de crear el esquema de clases a partir de una base de datos y crear la definición de base de datos a partir del esquema de clases. Permite invertir código fuente de programas, archivos ejecutables y binarios en modelos UML al instante, creando de manera simple toda la documentación. Cabe destacar igualmente su robustez, usabilidad y portabilidad. Está diseñada para usuarios interesados en sistemas de software de gran escala con el uso del acercamiento orientado a objeto. Incorpora el soporte para trabajo en equipo, esto permite que varios desarrolladores trabajen a la vez en el mismo diagrama y vean en tiempo real los cambios realizados por sus compañeros.

1.4.4.2 Rational Rose Enterprise Edition

Rational Rose Enterprise Edition es una herramienta CASE que permite el diseño detallado del software y la generación de código fuente de programas y bases de datos e ingeniería inversa (obtención del diseño a partir del código fuente), basado en modelos con soporte UML. Fue desarrollada por los creadores de UML Booch, Rumbaugh y Jacobson. Cubre todo el ciclo de vida de un proyecto desde la concepción y formalización del modelo, pasando por la construcción de los componentes, transición a los usuarios, hasta la certificación de las distintas fases y entregables.

El navegador UML de Rational Rose permite establecer una trazabilidad real entre el modelo (análisis y diseño) y el código ejecutable. Facilita el desarrollo de un proceso cooperativo en el que todos los agentes tienen sus propias vistas de información (vista de Casos de Uso, vista Lógica, vista de Componentes y vista de Despliegue), pero utilizan un lenguaje común para comprender y comunicar la estructura y la funcionalidad del sistema en construcción. Esta herramienta propone la utilización de cuatro tipos de modelos para realizar el diseño del sistema, utilizando una vista estática y otra dinámica de los modelos del sistema, uno lógico y otro físico. Permite crear y refinar estas vistas, creando de esta forma un modelo completo que representa el dominio del problema y el sistema de software. Rational Rose utiliza un proceso de desarrollo iterativo controlado, donde el desarrollo se lleva a cabo en una secuencia de iteraciones. Cada iteración comienza con una primera aproximación del análisis, diseño e implementación para identificar los riesgos del diseño, los cuales se utilizan para conducir la iteración, primero se identifican los riesgos y después se prueba la aplicación para que estos se hagan mínimos. Cuando la implementación pasa todas las pruebas que se determinan en el proceso, esta se revisa y se añaden los elementos modificados al modelo de análisis y diseño. Una vez que la actualización del modelo se ha modificado, se realiza la siguiente iteración. Rational Rose permite que hayan varias personas trabajando a la vez en el proceso iterativo controlado, para ello posibilita que cada desarrollador opere en un espacio de trabajo privado que contiene el modelo completo y tenga un control exclusivo sobre la propagación de los cambios en ese espacio de trabajo. También es posible descomponer el modelo en unidades controladas e

integrarlas con un sistema para realizar el control de proyectos que permite mantener la integridad de dichas unidades.

1.4.5 Lenguajes de programación

1.4.5.1 PHP

PHP es un lenguaje script que corre del lado del servidor. Es un lenguaje muy utilizado por los desarrolladores de páginas Web dinámicas, aún más cuando su versión reciente, la PHP 5 (v5.05), tiene incorporado un grupo de mejoras en cuanto a Programación Orientada a Objetos (POO), funciones y extensiones. Puede ser utilizado para la creación de otro tipo de programas incluyendo aplicaciones con interfaz gráfica usando las librerías. PHP permite embeber sus pequeños fragmentos de código dentro de la página HTML y realizar determinadas acciones de una forma fácil y eficaz sin tener que generar programas íntegramente en un lenguaje distinto al HTML. Presenta ventajas como:

- Muy sencillo de aprender.
- Similar en sintaxis a C y a PERL.
- Permite las técnicas de Programación Orientada a Objetos (POO).
- Se puede hacer de todo lo que se pueda transmitir por vía HTTP.
- Es software libre. Se puede obtener en la Web y su código esta disponible bajo la licencia GPL.
- Viene acompañado por una excelente biblioteca de funciones que permite realizar cualquier labor (acceso a base de datos, encriptación, envío de correo, XML y creación de PDF).
- Es un lenguaje multiplataforma.
- Capacidad de conexión con la mayoría de los manejadores de base de datos que se utilizan en la actualidad.
- Leer y manipular datos desde diversas fuentes, incluyendo datos que pueden ingresar los usuarios desde formularios HTML.

- Capacidad de expandir su potencial utilizando la enorme cantidad de módulos (llamados extensiones).
- Puede interactuar con muchos motores de bases de datos tales como MySQL, MS SQL, Oracle, Informix, PostgreSQL, y otros muchos.

Posee gran rapidez. PHP generalmente es utilizado como módulo de Apache, lo que lo hace extremadamente veloz. Esta completamente escrito en C, así que se ejecuta rápidamente utilizando poca memoria.

1.4.5.2 Java Script

Lenguaje que permite dar dinamismo a las páginas Web. Los scripts escritos con Java Script amplían las páginas Web dotándolas de una interactividad con el usuario mucho mayor que los simples hipervínculos y formularios. Además permiten cambiar de forma dinámica, es decir, se adaptan al entorno de la máquina cliente mientras son recibidas por el navegante y permite crear efectos especiales en las páginas. El encargado de interpretar las instrucciones JavaScript es el navegador cliente siendo el que las ejecuta para realizar las interactividades. Después de HTML, Java Script es el siguiente paso que puede dar un programador Web que decida mejorar sus páginas y la resistencia de sus proyectos. Es un lenguaje bastante sencillo y fácil de aprender, pensado para hacer las cosas con rapidez.

Con Java Script se crean aplicaciones específicamente orientadas a su funcionamiento en la red. Este lenguaje permite crear páginas HTML dinámicas que procesen la entrada del usuario y además que sean capaces de gestionar datos usando archivos y bases de datos relacionales. También facilita validar información sin necesidad de realizar transmisión de datos por la red. Posee una posición ventajosa al ser utilizado para la creación de grandes soluciones para la Web. Para ejecutar el código sólo hay que crearlo y cargarlo. Muchas páginas Web incorporan elementos que lo usan y por lo general los exploradores Web trabajan con él. Posibilita la reducción de carga del servidor, se hace cargo de la mayoría de las funcionalidades del cliente de las cuales se encargaba el servidor.

1.4.6 Sistema Gestores de Contenidos

Un Sistema de Gestión de Contenidos (Content Management System en inglés, abreviado CMS) es un programa que permite crear una estructura de soporte (framework) para la creación y administración de contenidos por parte de los participantes principalmente en páginas Web.

Consiste en una interfaz que controla una o varias bases de datos donde se aloja el contenido del sitio. El sistema permite manejar de manera independiente el contenido y el diseño. Así, es posible manejar el contenido y darle en cualquier momento un diseño distinto al sitio, sin tener que darle formato al contenido de nuevo. Además de permitir la fácil y controlada publicación en el sitio a varios editores. Un ejemplo clásico son los editores que cargan el contenido al sistema y otro de nivel superior que permite que estos contenidos sean visibles a todo el público.

Existe un número importante de CMS que hacen posible todas estas funcionalidades. Gran cantidad de estas aplicaciones, al menos las más importantes, como lo pueden ser Joomla, Drupal, etc.

Estos sistemas son adaptables, permiten una gran flexibilidad y escalabilidad, justifican su utilización en prácticamente cualquier tipo de Web y tienen una gran comunidad.

1.4.6.1 Joomla

Joomla es uno de los más potentes CMS del planeta. Es usado en todo el mundo para todo, desde los sitios web más simples hasta las más complejas aplicaciones corporativas. Es muy sencillo de instalar, simple de administrar y además es confiable. Incluye características como: hacer caché de páginas para mejorar el rendimiento, indexamiento web, versiones imprimibles de páginas, flash con noticias, etc. La primera versión de Joomla fue publicada en septiembre de 2005. Se trataba de una versión mejorada de Mambo 4.5.2.3 combinada con otras modificaciones de seguridad y anti-*bugs*. Actualmente los programadores han publicado Joomla 1.5 bajo un código completamente reescrito y construido bajo PHP 5.

1.4.6.2 Drupal

Drupal es un sistema de administración de contenido para sitios Web. Permite publicar artículos, imágenes u otros archivos y servicios añadidos como foros, encuestas, votaciones, blogs, administración de usuarios y permisos. Drupal es un sistema dinámico: en lugar de almacenar sus contenidos en archivos estáticos en el sistema de ficheros del servidor de forma fija, el contenido textual de las páginas y otras configuraciones son almacenados en una base de datos y se editan utilizando un entorno Web incluido en el producto. Contiene las principales características de un CMS en el núcleo del sistema, gran comunidad de usuarios, muy flexible (se adapta a cualquier cosa), ampliable (cientos de módulos), buena arquitectura de programación ya que es seguro y posee un buen rendimiento. Además es multilinguaje, posee un sistema de búsqueda integrada, genera un informe de errores y estadística e incorpora un sistema de caché que agiliza la velocidad de las páginas.

Características generales de Drupal

- Ayuda on-line: Un robusto sistema de ayuda online y páginas de ayuda para los módulos del núcleo, tanto para usuarios como para administradores.
- Búsqueda: Todo el contenido en Drupal es totalmente indexado en tiempo real y se puede consultar en cualquier momento.
- Código abierto: El código fuente de Drupal está libremente disponible bajo los términos de la licencia GNU/GPL. Al contrario que otros sistemas de 'blogs' o de gestión de contenido propietarios, es posible extender o adaptar Drupal según las necesidades.
- Módulos: La comunidad de Drupal ha desarrollado muchos módulos que proporcionan funcionalidades como página de categorías, autenticación mediante jabber, mensajes privados, *bookmarks*, etc.
- Personalización: Un robusto entorno de personalización está implementado en el núcleo de Drupal. Tanto el contenido como la presentación pueden ser individualizados de acuerdo las preferencias definidas por el usuario.
- URLs amigables: Drupal usa el *mod_rewrite* de Apache para crear URLs que son manejables por los usuarios y los motores de búsqueda.

Uno de los tipos de CMS más reconocidos ofrece una plataforma necesaria para desarrollar e implementar aplicaciones que den solución a necesidades específicas y para construir soluciones de gestión de contenidos. Drupal se destaca entre los CMS de tipo genérico, este fue seleccionado como mejor CMS del 2007 y 2008. Se optó por el uso de este CMS para el sistema de gestión ya que es uno de los más completos actualmente, acorde a los requerimientos de nuestra aplicación y flexible a las modificaciones.

1.4.6.3 Plone

Es un CMS muy flexible y poderoso, que presenta una excelente interfaz de usuario, una instalación muy limpia, un impresionante grado de personalización, y una integración con LDAP u otros sistemas de *login*. Posibilita además construir un sitio cuyo contenido se puede enriquecer constantemente, debido a sus facilidades de actualización e interacción, y con ello, obtener ventajas competitivas. Pero, probablemente su característica más importante es que es un sistema de código abierto y libre.

Plone es un sistema de gestión de contenidos basado en Zope45 y programado en Python, que está basado en código abierto. Puede utilizarse como servidor intranet o extranet, un sistema de publicación de documentos y una herramienta de trabajo en grupo para colaborar entre entidades distantes. Es un CMS muy reciente lo que no lo hace menos portable y poderoso. Presenta características similares al resto de los CMS como completamente modularizado donde los módulos se pueden gestionar, personalización extensible para los principales elementos de seguridad, gestión de usuarios y además tiene un sistema de permisos de grupo de gran alcance y de uso fácil.

1.4.7 AJAX

AJAX, acrónimo de *Asynchronous Java Script And XML* es una técnica de desarrollo Web para crear aplicaciones interactivas o RIA (*Rich Internet Applications*). Éstas se ejecutan en el cliente, es decir, en el navegador de los usuarios y mantiene comunicación asíncrona con el servidor en segundo plano. De esta forma es posible realizar cambios sobre la misma página sin necesidad

de recargarla; esto significa aumentar la interactividad, velocidad y usabilidad en la misma. Con esta técnica se abre todo un mundo de posibilidades a la hora de construir aplicaciones Web altamente interactivas. Tiene numerosas características que hacen que cada día sea más potente, por ejemplo se basa en estándares abiertos y esto es debido a que de todas las tecnologías que lo conforman XML HTTP *Request Object* es el único que no es estándar, pero si es soportado por los navegadores que son más utilizados en Internet por ejemplo el Internet Explorer, Mozilla y Opera. Dentro de sus características se pueden resaltar: Gran usabilidad: Permite a las páginas hacer una petición muy pequeña al servidor y recibirla sin necesidad de recargar la página completa. Válido en cualquier plataforma y navegador: Es muy fácil programar aplicaciones AJAX en los navegadores que tienen el número uno en el mercado de Internet, los cuales son Internet Explorer, Mozilla y Firefox; pero también se pueden construir aplicaciones Web basadas en AJAX que funcionen en otros navegadores. Independiente del tipo de tecnología de servidor que se utilice: AJAX funciona con cualquier navegador, es compatible con cualquier tipo de servidor estándar y lenguaje de programación Web, ejemplo: PHP, ASP. ASP.Net, Perl, y gracias a que es completamente compatible el desarrollo de estas tecnologías hace que AJAX sea cada día más popular y mejor. Mejora la estética de la Web: Con AJAX se puede combinar toda la imaginación del desarrollador con la usabilidad de una aplicación Web de forma tal que si una aplicación no estuviera dentro de un navegador, pudiera pasar por una aplicación normal.

1.4.8 Sistema Gestor de Base de Datos (SGBD)

Los sistemas gestores de Base de Datos son un tipo de software muy específico, dedicado a servir de interfaz entre la base de datos, el usuario y las aplicaciones que la utilizan. Se compone de un lenguaje de definición de datos, un lenguaje de manipulación de datos y un lenguaje de consulta.

Estos permiten manejar con facilidad grandes volúmenes de información en muy poco tiempo y ofrecen independencia y seguridad en el tratamiento de información.

Existen varios objetivos que deben cumplir los SGBD, dentro de los cuales se consideran más importantes los siguientes:

- Seguridad: Los SGBD deben disponer de un sistema de permisos a usuarios y grupos de usuarios, que permitan restringir los accesos no autorizados.
- Integridad: Se debe proteger la información almacenada ante fallos de hardware, datos introducidos por descuido o cualquier situación que pueda alterar la integridad de los datos almacenados.
- Tiempo de respuesta: Se debe minimizar el tiempo que el SGBD tarda en proporcionar la información solicitada y en guardar los cambios efectuados.
- Abstracción de la información: Los usuarios deben ser ajenos a detalles sobre la localización física donde han sido almacenados los datos.
- Redundancia mínima: Un diseño correcto de una BD debe evitar la duplicación o redundancia de la información.
- Independencia: Debe ser posible modificar la BD sin necesidad de realizar cambios en las aplicaciones que la accedan.

El propósito general de los sistemas de gestión de Base de Datos es el de manejar de manera clara, sencilla y ordenada un conjunto de datos que posteriormente se convertirán en información relevante, para un buen manejo de datos. Dentro de los SGBD libres más utilizados se encuentra MySQL.

1.4.8.1 Oracle

Oracle es sin duda una de las mejores bases de datos que existen en el mercado, es un sistema gestor de base de datos robusto, tiene muchas características que garantizan la seguridad e integridad de los datos; que las transacciones se ejecuten de forma correcta, sin causar inconsistencias; ayuda a administrar y almacenar grandes volúmenes de datos; estabilidad, escalabilidad.

Oracle está disponible en múltiples plataformas como Windows, Linux, todas las versiones de Unix ofrecidas por diversas empresas como IBM, Sun, Digital, HP, Sequent, etc. y también en VAX-VMS, así como en MVS. La naturaleza multiplataforma de Oracle, lo convierte en una verdadera solución empresarial.

Aunque su dominio en el mercado de servidores empresariales ha sido casi total, recientemente sufre la competencia de gestores de bases de datos comerciales y de la oferta de otros con licencia Software Libre como PostgreSQL, MySQL o FireBird. Las últimas versiones de Oracle han sido certificadas para poder trabajar bajo Linux.

1.4.8.2 PostgreSQL

PostgreSQL es un Sistema de Gestión de Bases de Datos Objeto-Relacionales (ORDBMS) de código abierto más avanzada del mundo. Comenzó como un proyecto denominado Ingres en la Universidad Berkeley de California. Ingres fue más tarde desarrollado comercialmente por la *Relational Technologies/Ingres Corporation*. En 1986 otro equipo dirigido por Michael Stonebraker de Berkeley continuó el desarrollo del código de Ingres para crear un sistema de bases de datos objeto-relacionales llamado Postgres. En 1996, debido a un nuevo esfuerzo de código abierto y a la incrementada funcionalidad del software, Postgres fue renombrado a PostgreSQL, tras un breve periplo como Postgres95. El proyecto PostgreSQL sigue actualmente un activo proceso de desarrollo a nivel mundial gracias a un equipo de desarrolladores y contribuidores de código abierto.

PostgreSQL posee muchas características que normalmente sólo se encontraban en las bases de datos comerciales tales como DB2 u Oracle. La siguiente es una breve lista de algunas de esas características:

- **DBMS Objeto-Relacional:** PostgreSQL aproxima los datos a un modelo objeto-relacional, y es capaz de manejar complejas rutinas y reglas. Ejemplos de su avanzada funcionalidad son consultas SQL declarativas, control de concurrencia multi-versión, soporte multi-usuario, optimización de consultas, herencia, y *arrays*.

- Altamente Extensible: PostgreSQL soporta operadores, funcionales métodos de acceso y tipos de datos definidos por el usuario.
- Soporte_SQL_Completo: PostgreSQL soporta la especificación SQL99 e incluye características avanzadas tales como las uniones (*joins*) SQL92.
- Integridad Referencial: PostgreSQL soporta integridad referencial, la cual es utilizada para garantizar la validez de los datos de la base de datos.
- API Flexible: La flexibilidad del API de PostgreSQL ha permitido a los vendedores proporcionar soporte al desarrollo fácilmente para el RDBMS PostgreSQL. Estas interfaces incluyen Object Pascal, Python, Perl, PHP, ODBC, Java/JDBC, Ruby, TCL, C/C++, y Pike.
- Lenguajes Procedurales: PostgreSQL tiene soporte para lenguajes procedurales internos, incluyendo un lenguaje nativo denominado PL/pgSQL. Este lenguaje es comparable al lenguaje procedural de Oracle, PL/SQL. Otra ventaja de PostgreSQL es su habilidad para usar Perl, Python, o TCL como lenguaje procedural embebido.
- MVCC: MVCC, o Control de Concurrencia *Multi-Versión (Multi-Version Concurrency Control)*, es la tecnología que PostgreSQL usa para evitar bloqueos innecesarios. MVCC está considerado mejor que el bloqueo a nivel de fila porque un lector nunca es bloqueado por un escritor. En su lugar, PostgreSQL mantiene una ruta a todas las transacciones realizadas por los usuarios de la base de datos. PostgreSQL es capaz entonces de manejar los registros sin necesidad de que los usuarios tengan que esperar a que los registros estén disponibles.
- Cliente/Servidor: PostgreSQL usa una arquitectura proceso-por-usuario cliente/servidor. Esta es similar al método del Apache 1.3.x para manejar procesos. Hay un proceso maestro que se ramifica para proporcionar conexiones adicionales para cada cliente que intente conectar a PostgreSQL.

- *Write Ahead Logging (WAL)*: La característica de PostgreSQL conocida como “*Write Ahead Logging*” incrementa la dependencia de la base de datos al registro de cambios antes de que estos sean escritos en la base de datos. Esto garantiza que en el hipotético caso de que la base de datos se caiga, existirá un registro de las transacciones a partir del cual se puede restaurar la base de datos.

1.4.8.3 MySQL

MySQL es un Sistema de Gestión de Base de Datos relacional, multihilo y multiusuario con más de seis millones de instalaciones. Es el gestor de Bases de Datos de código fuente abierto más usado en la actualidad. Es propiedad y está patrocinado por una empresa privada, que posee el *copyright* de la mayor parte del código. Sin lugar a duda, lo mejor de MySQL es su velocidad a la hora de realizar las operaciones, lo que le hace uno de los gestores que ofrecen mayor rendimiento. Su bajo consumo lo hacen apto para ser ejecutado en una máquina con escasos recursos sin ningún problema.

Las utilidades de administración de este gestor son envidiables para muchos de los gestores comerciales existentes, debido a su facilidad de configuración e instalación. Las características funcionales de este gestor han dado lugar a un sistema de administración de las bases de datos, incomparable en velocidad, compactación, estabilidad y facilidad de despliegue, envidiables para muchos de los gestores comerciales existentes, debido a su gran facilidad de configuración e instalación. Además provee altas características de seguridad, contiene un sistema de autenticación capaz de aceptar o denegar las conexiones, también una herramienta para administración de los privilegios y métodos de encriptación. Tiene una probabilidad muy reducida de corromper los datos, incluso en los casos en los que los errores no se produzcan en el propio gestor, sino en el sistema en el que está. El conjunto de aplicaciones Apache-PHP-MySQL es uno de los más utilizados en Internet en servicios de foro y de buscadores de aplicaciones.

1.4.9 Servidor Web

Un servidor Web es un programa que se encarga de mantenerse a la espera de peticiones HTTP llevadas a cabo por un cliente HTTP que suele conocer como navegador. El navegador realiza una petición al servidor y éste le responde con el contenido que el cliente solicita. El servidor responde al cliente enviando el código HTML de la página; el cliente, una vez recibido el código, lo interpreta y lo muestra en pantalla. El cliente es el encargado de interpretar el código HTML, es decir, de mostrar las fuentes, los colores y la disposición de los textos y objetos de la página; el servidor tan sólo se limita a transferir el código de la página sin llevar a cabo ninguna interpretación de la misma. Algunos servidores web son:

- Apache
- Internet Information Server
- Sun Java System
- Zeus
- Light http

1.4.9.1 Apache

Apache es un servidor Web y de aplicaciones de código abierto para plataformas Unix (BSD, GNU/Linux, etc.), Windows, entre otras. Es el más extendido, por versátil y potente, entre los servidores Web en la Internet. Apache presenta entre otras características mensajes de error altamente configurables. La arquitectura del servidor Apache es modular. El servidor consta de una sección y diversos módulos que aportan mucha de la funcionalidad que podría considerarse básica para un servidor Web. La configuración de cada módulo se hace mediante la configuración de las directivas que están contenidas dentro del módulo. Los módulos del Apache se pueden clasificar en tres categorías:

1. Módulos Base: Contiene las funciones básicas del Apache.

2. Módulos Multiproceso: Son los responsables de la unión con los puertos de la máquina, aceptando las peticiones y enviando a los hijos a atender a las peticiones.

Módulos Adicionales: Cualquier otro módulo que le añada una funcionalidad al servidor.

Las funcionalidades más elementales se encuentran en el módulo base, siendo necesario un módulo multiproceso para manejar las peticiones. De este último se han diseñado varios para cada uno de los sistemas operativos sobre los que se ejecuta el Apache, optimizando el rendimiento y rapidez del código. El resto de funcionalidades del servidor se consiguen por medio de módulos adicionales que se pueden cargar. Para añadir un conjunto de utilidades al servidor, simplemente hay que añadirle uno nuevo, de forma que no es necesario volver a instalar el software. Apache puede ser adaptado a diferentes entornos y necesidades, con los diferentes módulos de apoyo y con la API de programación de estos. Además posee gran fiabilidad y extensibilidad que le convierten en una herramienta potente.

1.4.10 Justificación de las tecnologías a usar

Luego de un detallado análisis de las tendencias y tecnologías actuales más utilizadas para la construcción de aplicaciones web se propone hacer uso del Sistema Gestor de Contenido (CMS) Drupal para la creación del sistema. Debido a que es altamente configurable y con una flexibilidad que lo distingue de los demás al igual que la gestión de usuarios y roles. También considerado uno de los más seguros, con alta velocidad de respuesta a los usuarios y posee independencia entre la persistencia de datos y la presentación del contenido. Como lenguaje de programación se usa PHP debido a que es el lenguaje en el cual fue desarrollado Drupal, además de que es fácil de usar y a la vez seguro, es multiplataforma, cuenta con una amplia comunidad de desarrolladores tanto a nivel internacional como en la propia universidad permitiendo así el fácil acceso y consulta a la documentación referente al mismo. PostgreSQL es el Sistema Gestor de Base de Datos seleccionado, es un sistema gestor de bases de datos, entre sus características permiten modelar bases de datos relacionales el cual es un sistema objeto-relacional. Como servidor web Apache es el seleccionado ya que al igual que PostgreSQL y

Drupal es multiplataforma, es rápido, eficiente, seguro y altamente configurable en dependencia de las necesidades del usuario y la aplicación, puede adaptarse a diferentes entornos de trabajo. Posee varios módulos de apoyo que permiten un mejor uso del mismo y extienden sus funcionalidades. Por versátil y potente es el más usado entre los servidores Web en Internet. Tanto el CMS como el Sistema Gestor de Base de Datos y el servidor web son libres y cuentan con licencia GPL siendo esta una de las mayores ventajas para hacer uso de ellos.

Rational Unified Process (Proceso Unificado de Desarrollo), o RUP, es una metodología flexible de procesos de desarrollo de software que ayuda proveyendo guías consistentes y personalizadas de procesos para todo el equipo de proyecto [3]. Entre las ventajas de por qué usarlo se tiene:

- Es un proceso hecho práctico - Diferente que otras metodologías comerciales, la plataforma **RUP** hace que el proceso sea práctico con bases de conocimiento y guías para ayudar en el despegue de la planificación del proyecto, integrar rápidamente a los miembros del equipo y poner en acción el proceso personalizado.
- Se adapta a las necesidades de los proyectos - Solo la plataforma **RUP** proporciona un framework de proceso configurable que permite seleccionar e implantar los componentes específicos de proceso necesarios para proporcionar un proceso consistente y customizado para cada equipo y proyecto.

Capítulo 2

Descripción de la solución propuesta.

2.1 Introducción

En el presente capítulo se hace una descripción de las mejoras propuestas de acuerdo a la situación polémica planteada en el capítulo 1. A su vez, quedan identificados los actores y trabajadores del negocio y la justificación de los mismos, además de que se detallan los procesos con los que interactúan a través de la representación de los casos de uso del negocio y su formato ampliado. También se incluyen los diagramas de actividades correspondientes a cada uno de los casos de uso, para una mejor comprensión de los procesos que tienen lugar en el ámbito del negocio y mediante un diagrama de clases queda plasmada la interacción de los trabajadores con las distintas entidades.

A continuación de lo antes mencionado se detallan los requisitos funcionales a considerar para la construcción del sistema y los requisitos no funcionales que posibilitarán la correcta puesta en funcionamiento del Software una vez desarrollado. Por último, se presenta el diagrama de casos de uso del sistema y luego cada uno por separado acompañado de su descripción.

2.2 Modelo de negocio

2.2.1 Descripción de los procesos del negocio y mejoras propuestas

El proceso del seguimiento de la trayectoria productiva de los estudiantes vinculados a proyectos productivos en la facultad 1 tiene entre sus objetivos principales registrar un expediente del estudiante en el área de producción. Esta información incluye cursos optativos acreditados dentro del proyecto, roles que ocupa el estudiante en el mismo, eventos en que participa y sus resultados y el conjunto de todos estos datos da lugar a la nota de Práctica profesional del alumno al terminar su año académico.

Actualmente esto se realiza a través de procesos como: Asignar estudiante(s) a un proyecto, el cual comienza cuando se decide por parte de la dirección de la facultad incorporar uno o varios estudiantes a un proyecto en dependencia del estado del mismo. Primeramente al asignar un estudiante a un proyecto se registran todos los datos personales del mismo y son almacenados en un documento Excel; donde se beneficia el Vicedecano de producción, este trabajo lo lleva a cabo la Secretaria del Vicedecano de Producción, registrando estos datos en el Excel de Capital Humano. Luego se registran datos secundarios donde se incluyen datos necesarios como, línea de investigación, intereses y habilidades del estudiante (pueden ser modificados a lo largo de su trayectoria), beneficiándose igualmente el Vicedecano de producción y realizando trabajo la Secretaria de Vicedecano de producción, lo cual se guarda en el mismo Excel de Capital Humano.

Registrar evaluación de práctica profesional también es uno de los procesos, este se inicia cuando el Jefe de la asignatura de PP les pide a cada líder de proyecto las evaluaciones de práctica profesional a los estudiantes miembros del proyecto. Al registrar las evaluaciones de Práctica Profesional se beneficia el Jefe asignatura de PP, encargándose de guardar esas evaluaciones el líder de proyecto en el Excel de la evaluación de la producción.

Solicitar competencias de los estudiantes donde el vicedecano de producción solicita a los líderes que informen sobre las competencias alcanzadas por los estudiantes en un tema específico que sea de su interés.

Como último proceso está gestionar cursos optativos en el cual el Líder de proyecto se encarga de registrar en un documento elaborado por el los estudiantes que reciben el curso y el nombre del mismo para entregarlo a la secretaria docente, esta elabora un acta en la cual el líder registra las evaluaciones y se las entrega para que posteriormente sean registradas en Akademos.

2.2.2 Actores del Negocio

| Actor | Justificación |
|--------------------------|---|
| Vicedecano de producción | Se beneficia ya que a medida que tenga la mayor cantidad de estudiantes vinculados a la producción, los resultados de la producción de la facultad son mejores. |
| Jefe asignatura de PP | Recibe las evaluaciones de las notas de la asignatura de práctica profesional de cada estudiante vinculado a la producción. |
| Secretaria docente | Recibe de parte de los líderes de proyectos las notas de los cursos optativos impartidos a los estudiantes y los registra en Akademos. |

Tabla 2.0.1 Actores del Negocio

2.2.3 Trabajadores del Negocio

| Trabajador | Justificación |
|---------------------------------------|--|
| Secretaria del Vicedecano producción. | Es la persona encargada de insertar a los estudiantes en los proyectos productivos registrando todos sus datos. |
| Líder de Proyecto. | Es la persona encargada de asignar los roles de los estudiantes y emitir evaluación de la producción de acuerdo al desempeño de cada uno en el proyecto. |

Tabla 2.0.2 Trabajadores del Negocio

2.2.4 Diagrama de casos de uso del negocio

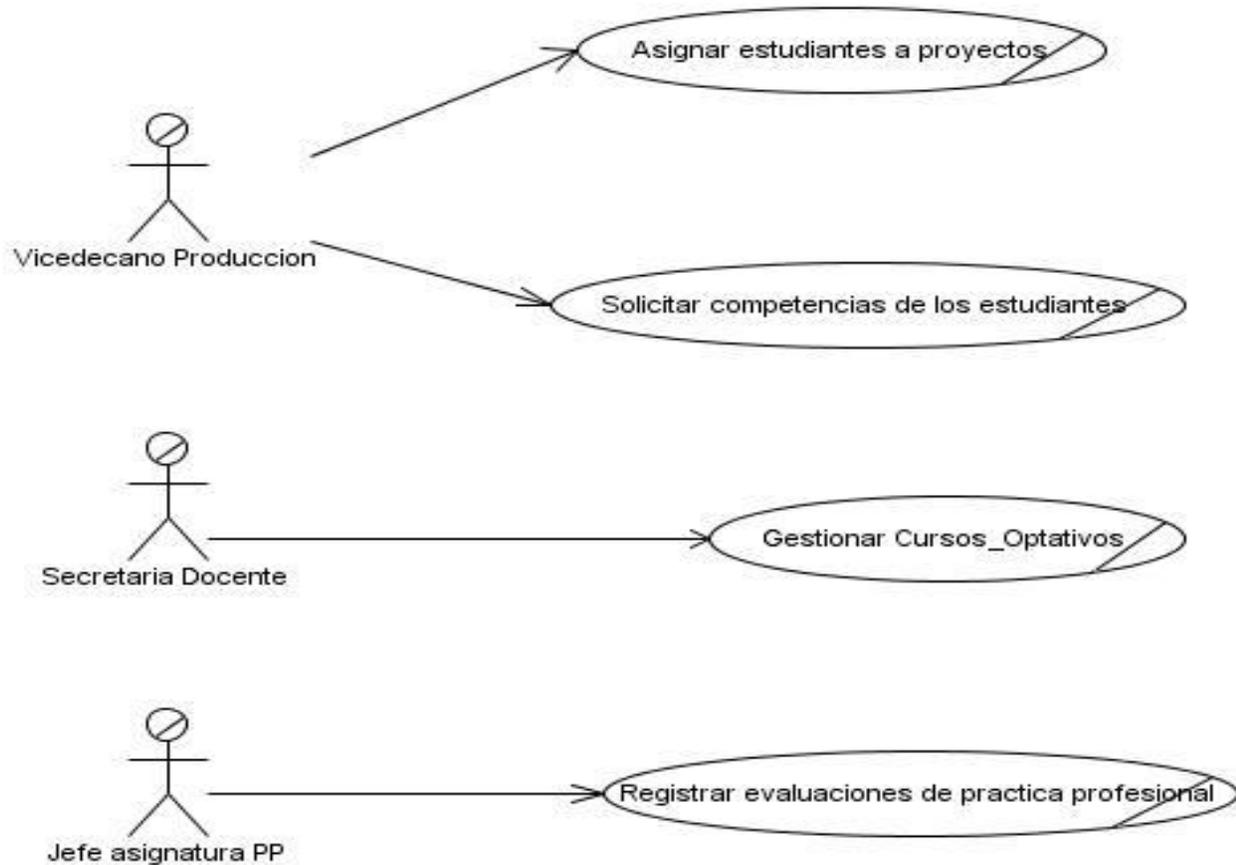


Figura 4 Diagrama de casos de uso del negocio

2.2.5 Descripción de los Casos de Uso del Negocio

| | |
|---------------|---|
| Caso de Uso: | Asignar estudiantes a proyectos |
| Actor: | Vicedecano de producción |
| Trabajadores: | Secretaria del vicedecano de producción, lideres de proyectos |

| | |
|------------------|--|
| Flujos Alternos | |
| Acción del Actor | Respuesta del Negocio |
| | En caso que el líder no haya asignado los roles a los estudiantes, debe hacerlo e informar al Vicedecano de producción. |
| Pos condiciones | El o los estudiantes quedan incorporados en un proyecto productivo y los datos registrados en la oficina del vicedecano de producción. |

Tabla 2.0.3 Descripción del caso de uso Asignar Estudiantes a Proyectos

| | |
|--|---|
| Caso de Uso: | Solicitar Competencias |
| Actores: | Vicedecano de producción |
| Trabajadores: | Líder de proyecto. |
| Resumen: | Este caso de uso se inicializa cuando el vicedecano de producción solicita a los líderes que informen sobre las competencias alcanzadas por los estudiantes en un tema específico que sea de su interés, luego el líder de proyecto busca en sus estudiantes los que tengan competencias alcanzadas en ese tema y se los entrega al vicedecano de producción. |
| Precondiciones: | Que el estudiante sea ya miembro del proyecto |
| Flujo Normal de Eventos | |
| Acción del Actor | Respuesta del Negocio |
| Solicita al líder de proyecto estudiantes que tengan las competencias alcanzadas que sean de su interés. | El líder del proyecto busca los estudiantes que hayan |

| | |
|-----------------|--|
| | <p>alcanzado las competencias necesarias para el vicedecano de producción.</p> <p>El líder de proyecto informa al vicedecano de producción la información deseada.</p> |
| Pos condiciones | |

Tabla 2.0.4 Descripción del caso de uso del Negocio Solicitar Competencias

| | |
|---|---|
| Caso de Uso: | Registrar Cursos Optativos |
| Actores: | Secretaria Docente |
| Trabajadores: | Líder de proyecto. |
| Resumen: | Se inicializa cuando el Líder de proyecto registra en un documento elaborado por él, los estudiantes que reciben el curso y el nombre del mismo, para entregarlo a la secretaria docente, esta elabora un acta en la cual el líder registra las evaluaciones y se las entrega para que posteriormente sean registradas en Akademos. |
| Precondiciones: | Que se haya impartido algún curso optativo en el proyecto. |
| Flujo Normal de Eventos | |
| Acción del Actor | Respuesta del Negocio |
| 2. La secretaria docente hace un acta con los datos y se la envía al líder de proyecto. | 1. Entrega a la secretaria el nombre del curso optativo y los estudiantes que lo integran |

| | |
|--|--|
| <p>4. Registra los datos para luego actualizarlos en Akademos.</p> | <p>3. Luego de que se haya impartido el curso registra las notas y se las envía a la secretaría docente.</p> |
| <p>Pos condiciones</p> | |

Tabla 2.0.5 Descripción del caso de uso del Negocio Registrar Curso Optativo

| | | |
|--|---|--|
| <p>Caso de Uso:</p> | <p>Registrar evaluaciones de Práctica Profesional</p> | |
| <p>Actores:</p> | <p>Jefe de la asignatura de Práctica Profesional</p> | |
| <p>Trabajadores:</p> | <p>Líder de proyecto</p> | |
| <p>Resumen:</p> | <p>Este caso de uso lo inicializa el jefe de la asignatura Practica Profesional cuando solicita a cada líder de proyecto por la nota de PP de los estudiantes vinculados a ese proyecto, el líder informa sobre las notas teniendo en cuenta la trayectoria durante el curso de esos estudiantes y emite una nota que posteriormente es registrada en Akademos por el jefe de la asignatura PP.</p> | |
| <p>Precondiciones:</p> | | |
| <p>Flujo Normal de Eventos</p> | | |
| <p>Acción del Actor</p> | <p>Respuesta del Negocio</p> | |
| <p>Solicita a los líderes de proyecto la nota de PP de los estudiantes del proyecto.</p> | <p>El líder de proyecto chequea la trayectoria de los estudiantes, teniendo en cuenta, cumplimiento de las tareas, desempeño en el proyecto de acuerdo a su rol, participación en eventos y resultados en los mismos y cursos optativos vencidos.</p> <p>El líder de proyecto emite una nota para cada estudiante.</p> | |

| | |
|---|--|
| <p>5. El jefe de la asignatura PP registra las notas de todos los estudiantes de la facultad en Akademos terminando así el caso de uso.</p> | <p>El líder de proyecto informa al jefe de la asignatura PP las notas de los estudiantes del proyecto.</p> |
| <p>Flujos Alternos</p> | |
| <p>Acción del Actor</p> | <p>Respuesta del Negocio</p> |
| | <p>En caso que un estudiante tenga participación en eventos o cursos optativos acreditados, el líder decide la nota por su propio criterio de acuerdo al comportamiento del estudiante en el proyecto.</p> |
| <p>Pos condiciones</p> | <p>Quedan registradas en Akademos todas las notas de PP de los estudiantes.</p> |

Tabla 2.0.6 Descripción del caso de uso del Negocio Registrar evaluaciones de Practica Profesional

2.3 Modelo del Sistema

2.3.1 Requerimientos funcionales

RF1- Autenticar usuario.

RF2- Gestionar roles a los usuarios del sistema.

RF3- Asignar competencias a un estudiante.

RF4-Obtener estudiantes con una determinada competencia.

RF5- Asignar un estudiante a un proyecto productivo.

RF6- Gestionar información de un estudiante dentro de un proyecto productivo.

6.1 Asignar rol a un estudiante dentro de un proyecto productivo.

6.2 Registrar cursos optativos de un estudiante dentro de un proyecto productivo.

6.3 Registrar resultados de la participación de un estudiante en algún evento referente a la producción y la investigación.

RF7- Registrar nota de Práctica Profesional a un estudiante dentro de un proyecto productivo

RF8- Obtener nota de Práctica Profesional de un estudiante dentro de un proyecto productivo.

RF9- Obtener estudiantes por rol.

RF10- Obtener estudiantes por proyecto.

RF11- Registrar incidencias.

2.3.2 Requerimientos no funcionales

2.3.2.1 Requerimientos de software

Servidor web Apache versión 1.3.

Servidor de Bases de Datos PostgreSQL versión 8.2 o 8.3.

Sistema Operativo Windows XP, Linux (cualquier distribución).

Navegador web Mozilla Firefox o Internet Explorer.

PHP versión 5.x.

2.3.2.2 Requerimientos de hardware

Requerimientos mínimos para la máquina servidora:

Procesador de tipo Pentium III con velocidad de 1 GHz o superior.

Memoria RAM de 256 MB o superior.

Disco duro de 20 GB o superior.

Tarjeta de red o módem.

2.3.2.3 Requerimientos de apariencia o interfaz externa

Interfaz amigable con facilidades para su uso por usuarios de cualquier nivel, reduciendo el tiempo de entrenamiento a los mismos.

Funcionalidades visibles en todo momento que faciliten la navegación.

Elementos organizativos para las funcionalidades como iconos, pestañas, menús y árboles jerárquicos.

2.3.2.4 Requerimientos de seguridad

Sistema de permisos por roles y usuarios para el acceso a la información, para protegerla de accesos no autorizados y divulgación.

Control de reglas de acceso permitiendo que solo accedan usuarios autorizados y la selección de direcciones IP o rangos desde los cuales se pueda acceder.

Los usuarios del sistema serán aquellos registrados en el Dominio UCI.

Validación de datos en el servidor, para evitar estados inconsistentes en la información.

2.3.2.5 Requerimientos de usabilidad

Familiarización con el entorno web.

Diseño adaptable a diferentes resoluciones de pantalla.

Diseño orientado a la simplicidad, con mínimas distracciones y una clara arquitectura de la información.

2.3.2.6 Requerimientos de rendimiento

Reducción de los tiempos de respuestas y alta velocidad de procesamiento de la información a través de peticiones asincrónicas al servidor.

2.3.2.7 Requerimientos de disponibilidad

Disponibilidad del sistema las 24 horas del día haciendo posible el acceso al mismo en cualquier momento.

2.4 Definición de actores del sistema

Los actores del sistema son terceros que colaboran con el mismo, pueden ser usuarios u otros sistemas con los que pueda interactuar. A continuación se muestra la tabla en la cual se incluyen los actores mencionados.

| Actor | Justificación |
|----------------------------------|--|
| Vicedecano de producción | Es el actor con más privilegios dentro del sistema ya que es el máximo responsable del mismo y se beneficia de el cuando obtiene reportes para comprobar algún dato de su interés u realizar otra actividad para la que esté facultado, a la vez que es quien da los permisos a otros usuarios para su interacción con el sistema. |
| Profesor de Práctica Profesional | Chequea las evaluaciones de Práctica Profesional de los estudiantes, así como las de los cursos optativos sin necesidad de interactuar personalmente con los líderes de proyecto que son quienes los registran primero en el sistema. |
| Líder de proyecto | Registra los datos de los miembros del proyecto referentes a la producción como roles que ocupan, competencias alcanzadas, eventos en que participan con sus resultados, cursos optativos e incidencias. |

Tabla 2.0.7 Actores del Sistema

2.5 Diagrama de Casos de Uso del sistema

Los casos de uso del sistema son artefactos que describen, a través de acciones y respuestas, el comportamiento desde el punto de vista del usuario. Son un conjunto de secuencia de actividades que un sistema ejecuta y que produce un resultado observable para un actor en dependencia de sus acciones. Dicho en otras palabras, no son más que fragmentos de funcionalidad que el sistema ofrece a los actores que interactúan con el mismo, reportándoles beneficios. El modelo de casos de uso del sistema describe lo que hace el sistema para cada tipo de usuario. Permite que se establezca un acuerdo entre clientes y desarrolladores sobre las condiciones y posibilidades (requisitos) que debe cumplir el sistema y proporciona la entrada fundamental para el análisis y diseño constituyendo así la base descriptiva para el desarrollo de una aplicación. A continuación se muestra una figura con el diagrama de casos de uso del sistema.

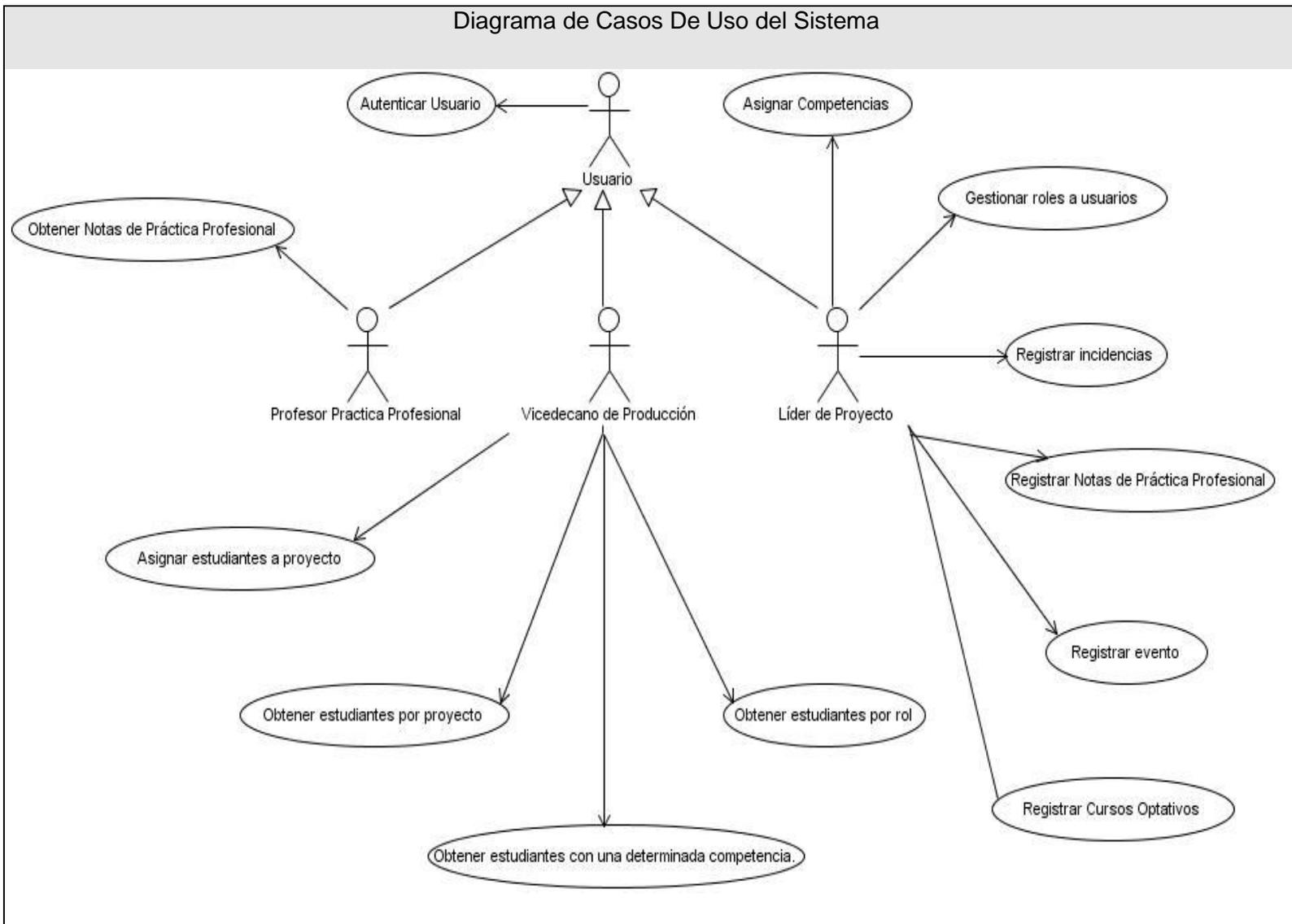


Figura 5 Diagrama de Casos De Uso del Sistema

2.6 Descripción de los Casos de Uso del sistema

La descripción de casos de uso del sistema es una descripción textual de las actividades que se realizan entre el usuario y el sistema, el flujo de lo que realiza el usuario sobre el sistema y la respuesta del mismo en dependencia de sus acciones.

A continuación se muestran las descripciones de los Casos de Uso del sistema, en el Anexo I se muestran los restantes.

| | |
|-------------|--|
| CUS_1 | Autenticar Usuario |
| Actor | Usuario |
| Descripción | El caso de uso se inicia cuando el usuario (que puede ser vicedecano, el líder de proyecto o el profesor de práctica profesional) introduce los datos para autenticarse, se verifica que los datos introducidos son correctos. En caso contrario no se permite la autenticación del usuario. |
| Referencia | RF1 |

Tabla2.0.8 Descripción del caso de uso del sistema Autenticar Usuario

| | |
|-------------|---|
| CUS_2 | Asignar estudiantes a proyecto |
| Actor | Vicedecano de producción |
| Descripción | El vicedecano de producción cuando decide asignar uno a varios estudiantes va a la opción insertar estudiantes y el sistema le muestra un formulario donde el vicedecano entra los datos referentes a este estudiante comenzando así la confección del expediente productivo de ese estudiante. |
| Referencia | RF5 |

Tabla2.0.9 Descripción del caso de uso del sistema Asignar Estudiantes a Proyecto

| | |
|-------|--|
| CUS_3 | Obtener estudiantes con una determinada competencia. |
| Actor | Vicedecano de producción |

| | |
|-------------|---|
| Descripción | Este caso de uso se inicializa cuando el vicedecano de producción solicita a los líderes que informen sobre las competencias alcanzadas por los estudiantes en un tema específico que sea de su interés, luego el líder de proyecto busca en sus estudiantes los que tengan competencias alcanzadas en ese tema y se los entrega al vicedecano de producción. |
| Referencia | RF4 |

Tabla 2.0.10 Descripción del caso de uso del sistema Obtener estudiantes con una determinada competencia

| | |
|-------------|--|
| CUS_4 | Registrar Notas de Práctica Profesional |
| Actor | Líder de Proyecto |
| Descripción | El líder de proyecto se introduce en la aplicación, busca la trayectoria de los estudiantes, para luego da una evaluación y guardar los datos. |
| Referencia | RF7 |

Tabla2.0.11 Descripción del caso de uso del sistema Registrar Notas de Práctica Profesional

| | |
|-------------|---|
| CUS_5 | Obtener Notas de Práctica Profesional |
| Actor | Profesor de Práctica Profesional |
| Descripción | El profesor de práctica profesional ingresa al sistema para obtener las notas de los estudiantes. |
| Referencia | RF8 |

Tabla 2.0.12 Descripción del caso de uso del sistema Obtener Notas de Práctica Profesional

| | |
|-------|----------------------------------|
| CUS_6 | Obtener estudiantes por proyecto |
|-------|----------------------------------|

| | |
|-------------|---|
| Actor | Vicedecano de Producción |
| Descripción | Este caso de uso se inicia cuando el vicedecano de producción desea obtener los estudiantes que pertenecen a un proyecto determinado. |
| Referencia | RF11 |

Tabla 2.0.13 Descripción del caso de uso del sistema Obtener estudiantes por proyecto

| | |
|-------------|--|
| CUS_7 | Obtener estudiantes por rol |
| Actor | Vicedecano de Producción |
| Descripción | Este caso de uso se inicia cuando el vicedecano desea obtener todos los estudiantes que desempeñen un rol determinado. |
| Referencia | RF10 |

Tabla 2.0.14 Descripción del caso de uso del sistema Obtener estudiantes por rol

| | |
|-------------|---|
| CUS_8 | Asignar Competencias |
| Actor | Líder de Proyecto |
| Descripción | Este caso de uso se inicia cuando el líder de proyecto decide insertar una o varias competencias a un estudiante. |
| Referencia | RF3 |

Tabla 2.0.15 Descripción del caso de uso del sistema Asignar Competencias

| | |
|-------------|--|
| CUS_9 | Registrar evento |
| Actor | Líder de proyecto |
| Descripción | Este caso de uso se inicia cuando el líder de proyecto desea registrar la participación de un estudiante en un evento determinado. |

| | |
|------------|------------|
| Referencia | RF6, RF6.3 |
|------------|------------|

Tabla 2.0.16 Descripción del caso de uso del sistema Registrar evento

| | |
|-------------|---|
| CUS_10 | Registrar incidencias |
| Actor | Líder de proyecto |
| Descripción | Este caso de uso se inicia cuando el líder de proyecto desea registrar alguna incidencia cometida por un estudiante en su proyecto. |
| Referencia | RF12 |

Tabla 2.0.17 Descripción del caso de uso del sistema Registrar incidencias

| | |
|-------------|--|
| CUS_11 | Registrar Cursos Optativos |
| Actor | Líder de proyecto |
| Descripción | Este caso de uso se inicia cuando el líder de proyecto desea registrar un nuevo curso optativo impartido a estudiantes del proyecto. |
| Referencia | RF6, RF6.2 |

Tabla 2.0.18 Descripción del caso de uso del sistema Registrar Cursos Optativos

| | |
|-------------|---|
| CUS_12 | Gestionar roles a usuarios |
| Actor | Vicedecano de producción |
| Descripción | Este caso de uso se inicia cuando el Vicedecano de producción decide modificar o delegar un rol a un usuario determinado. |
| Referencia | RF2 |

Tabla 2.0.19 Descripción del caso de uso del sistema Gestionar roles a usuarios

Conclusiones

En este capítulo se realizó la descripción de la solución propuesta, en la cual se definieron los requisitos funcionales y no funcionales. Además de modelar el diagrama de caso de uso del sistema y describir los casos de uso, para obtener así un mayor entendimiento del problema a resolver.

Capítulo 3

Análisis y Diseño del Sistema

3.1 Introducción

En el análisis y diseño de la solución propuesta se pretende modelar todos los artefactos que forman las clases de cada paquete de la aplicación y la relación entre ellos. Además, tiene como fin mostrar el modelo de datos, diagramas de clases del análisis, diagramas de colaboración y el diagrama de clases persistentes, permitiendo así una mejor comprensión de toda la información manipulada para el proceso de datos.

3.2 Modelo de Análisis

Al tener estructurados los requisitos en casos de usos del sistema, se analizan nuevamente los mismos pero esta vez para agruparlos en clases de análisis y paquetes.

El artefacto de mayor importancia a la hora de realizar el análisis es el modelo de análisis, el mismo está formado por paquetes que colaboran entre sí, conteniendo las clases y sus objetos. Entre las clases de análisis están las clases interfaz, entidad y control, las cuales describen las realizaciones de los casos de uso. Las clases interfaz modelan la interacción entre el sistema y sus actores, las clases entidad modelan información que posee larga vida y que es a menudo persistente y las clases control coordinan la realización de uno o unos pocos casos de uso coordinando las actividades de los objetos que implementan la funcionalidad del caso de uso

Se construye un diagrama de clases del análisis por cada caso de uso, además por cada escenario del caso de uso un diagrama de interacción; en el análisis RUP propone que se realice el diagrama de colaboración el cual destaca la organización de los objetos que participan en la interacción.

A continuación se muestran los diagramas de clases del análisis de los casos de usos descritos anteriormente y los diagramas de colaboración por cada escenario de estos casos de uso. En el anexo III se representan el resto de los diagramas.

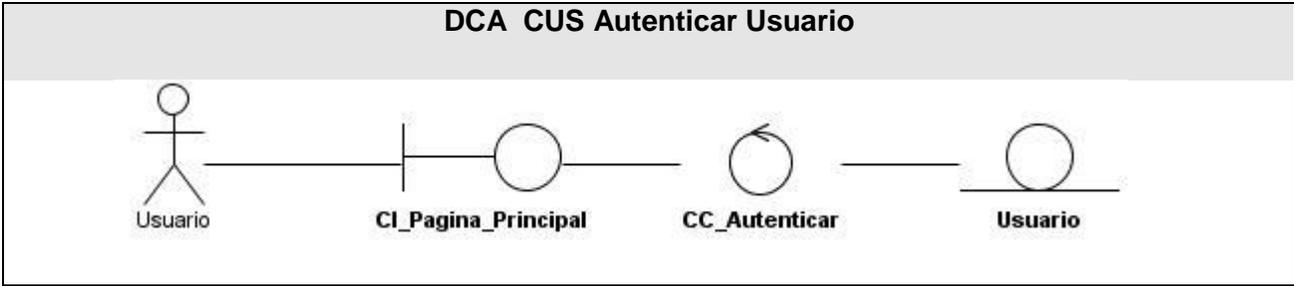


Figura 6 DCA CUS Autenticar Usuario

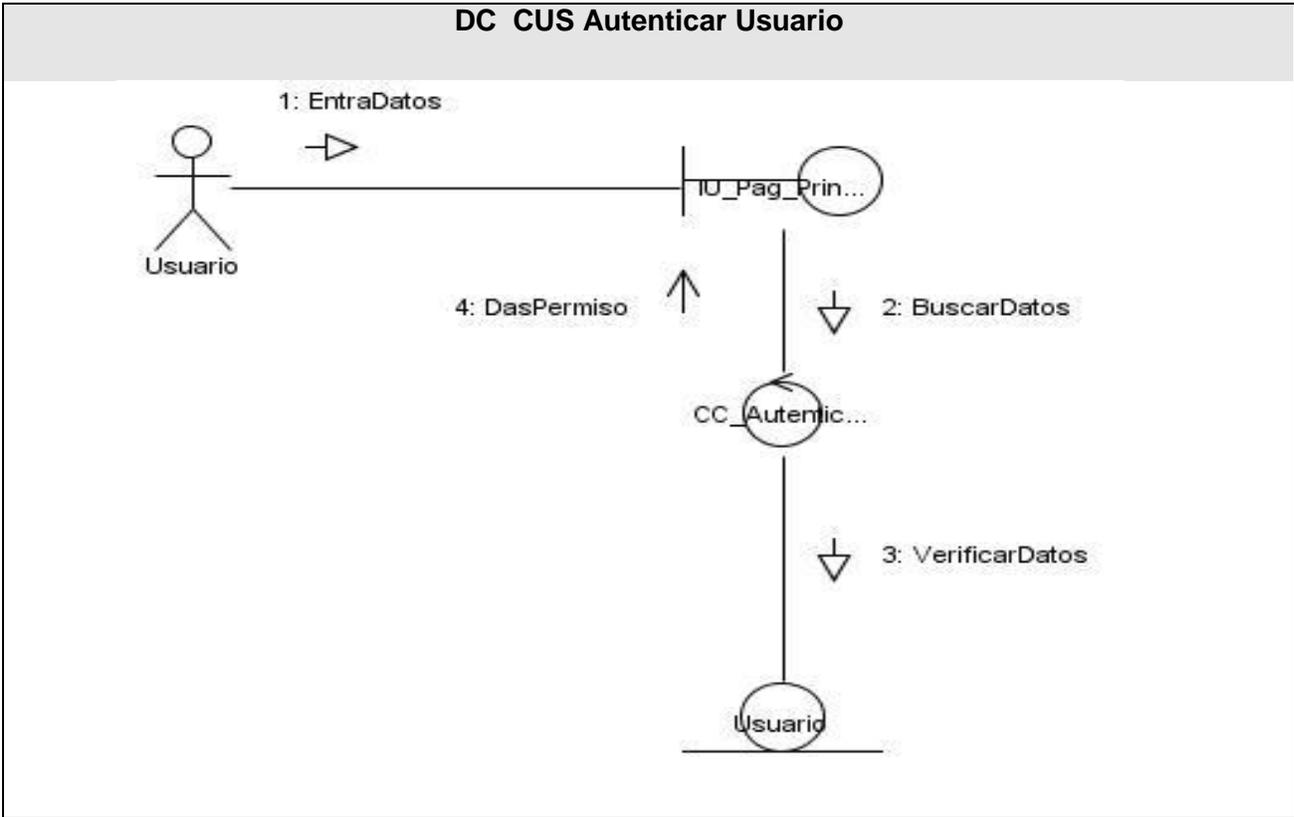


Figura 7 DC CUS Autenticar Usuario

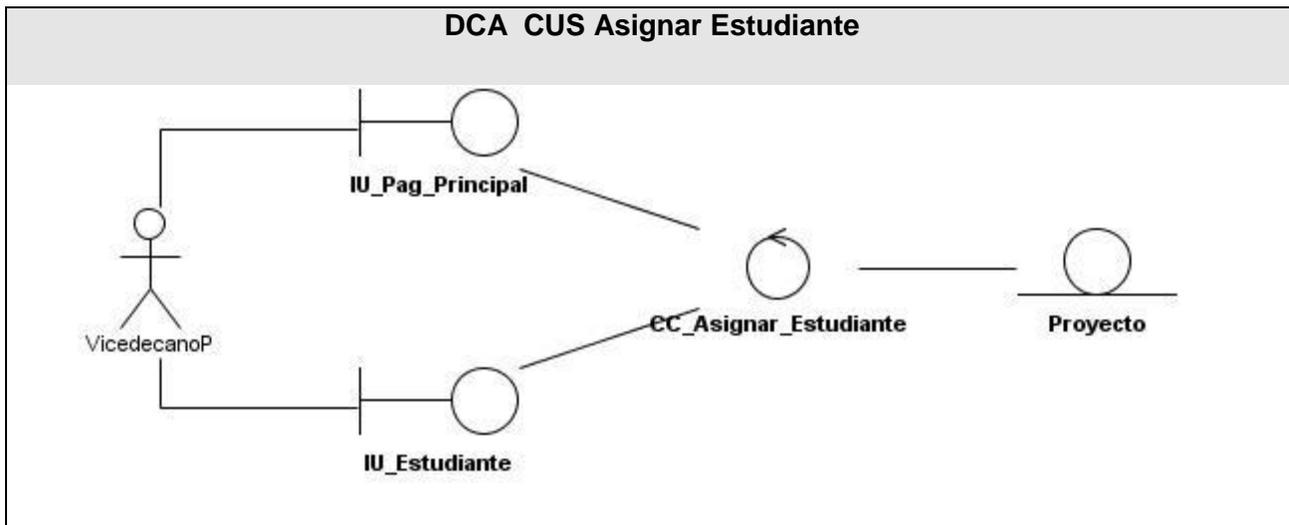


Figura 8 DCA CUS Asignar Estudiante

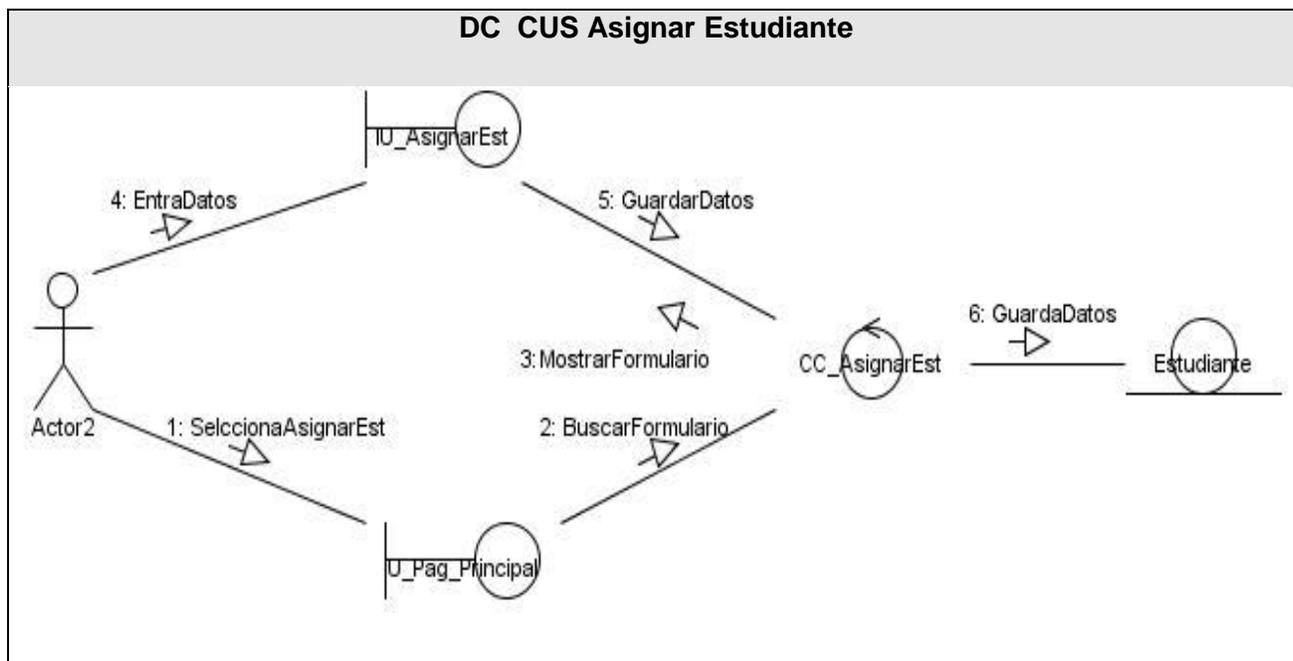


Figura 9 DC CUS Asignar Estudiante

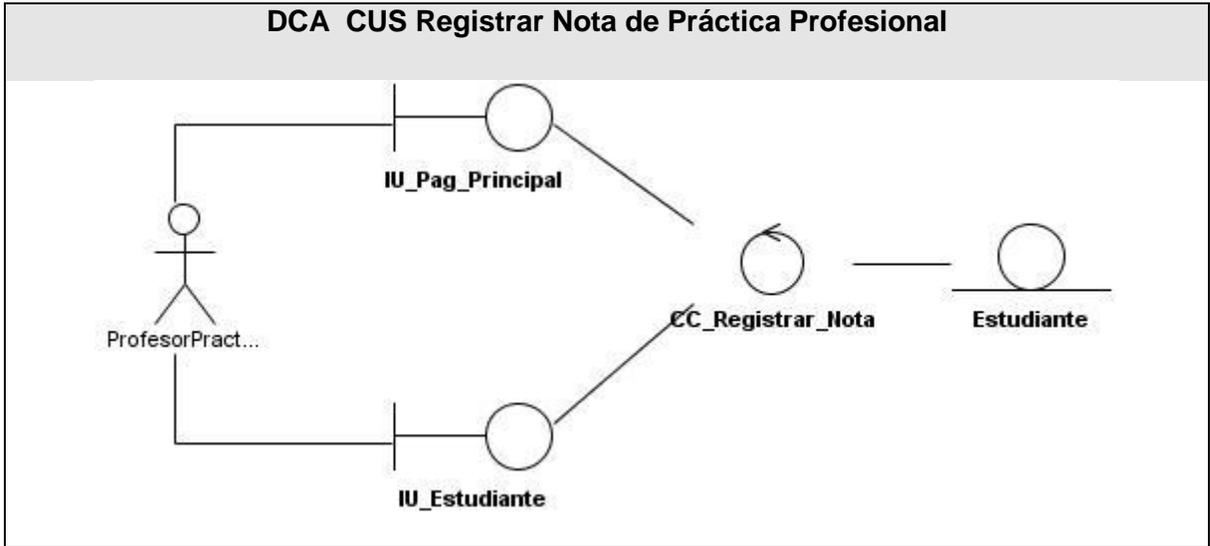


Figura 10 DCA CUS Registrar Nota de Práctica Profesional

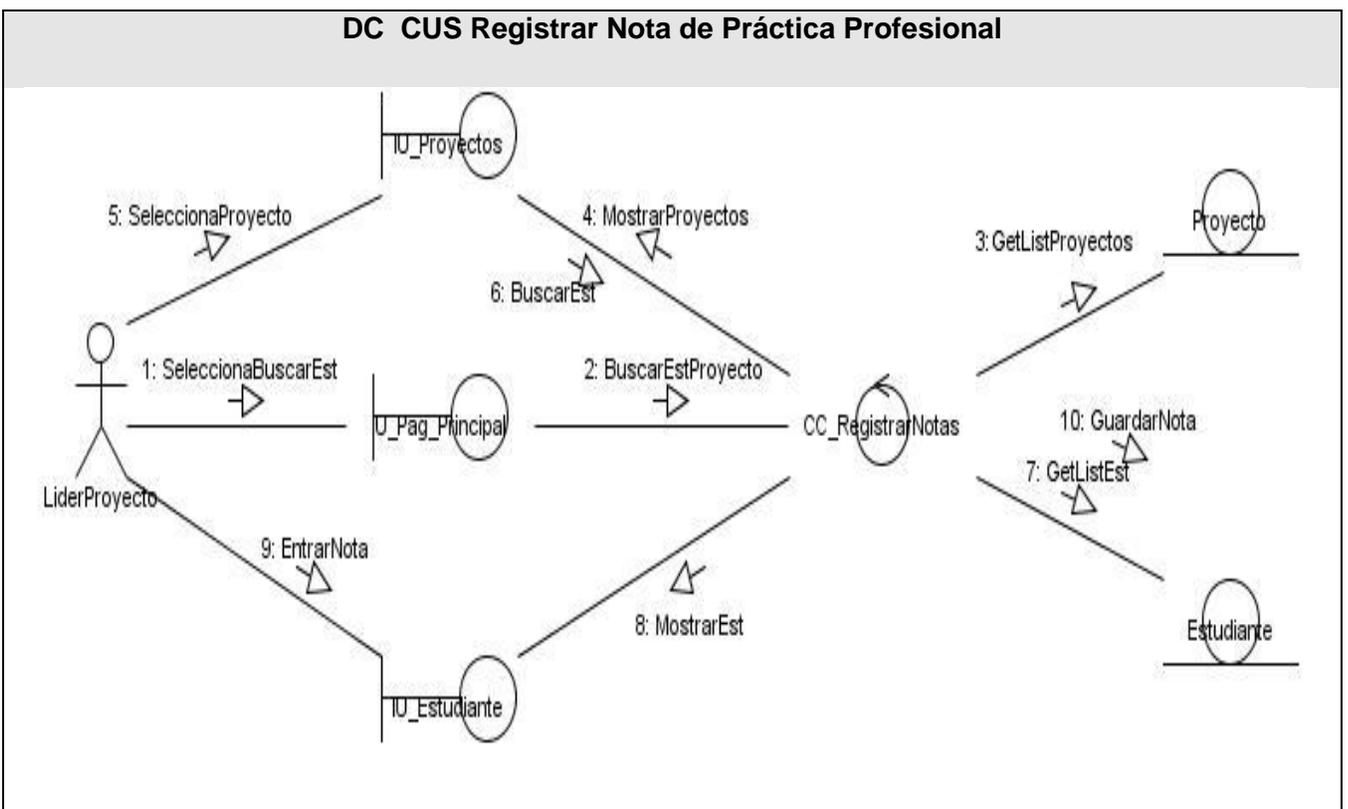


Figura 11 DC CUS Registrar Nota de Práctica Profesional

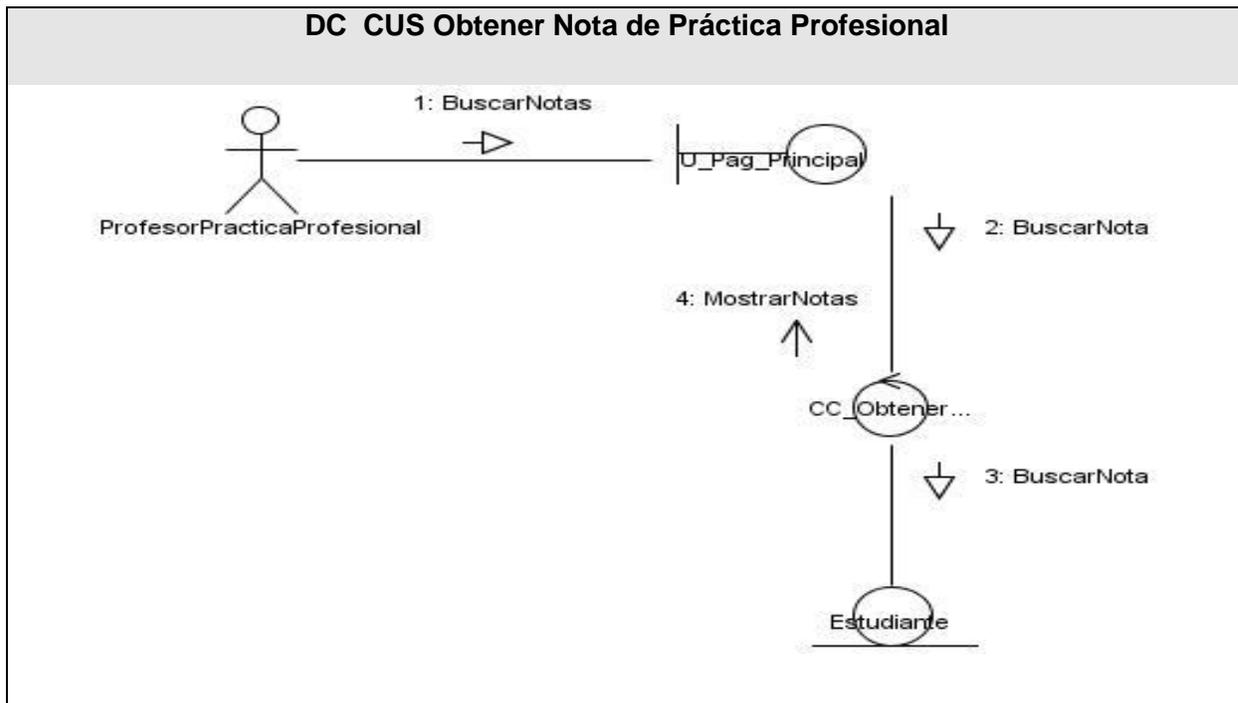


Figura 12 DC CUS Obtener Nota de Práctica Profesional

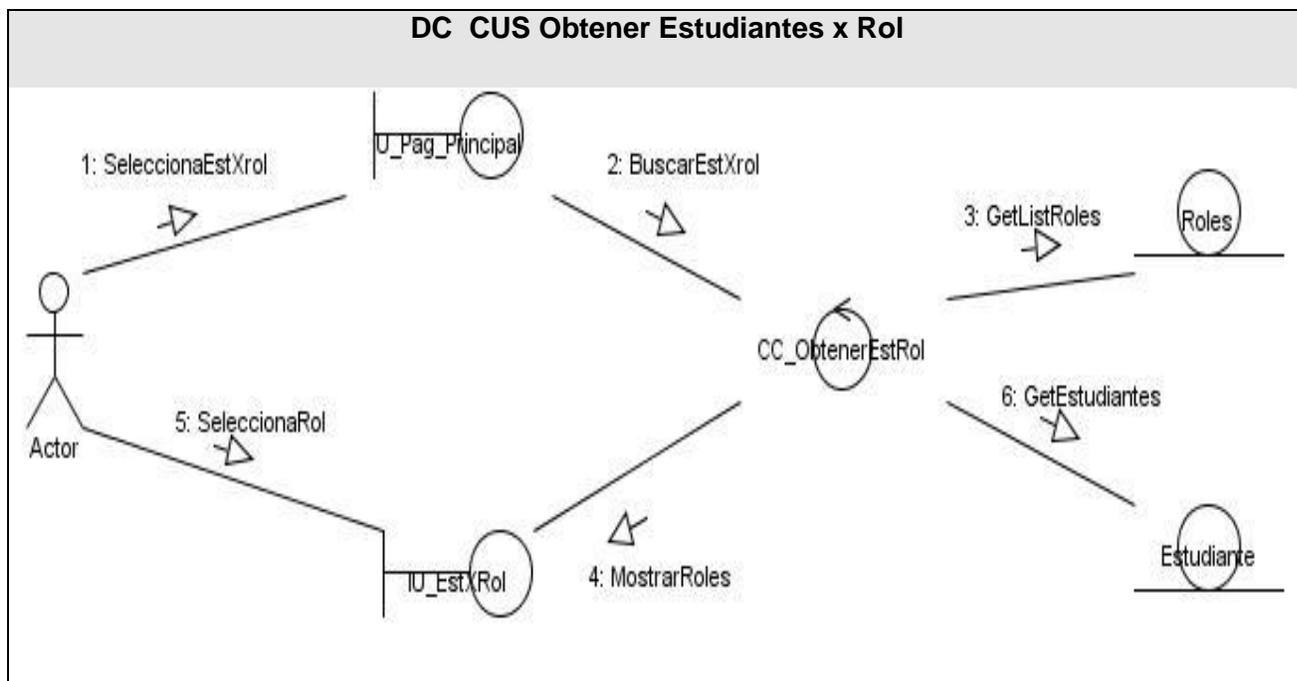


Figura 13 DC CUS Obtener Estudiantes x Rol

3.3 Patrones de diseño

Un buen diseño es la base fundamental para el desarrollo con éxito de una aplicación siendo el diagrama de clases de diseño un elemento fundamental del mismo, el cual describe la especificación de las clases y las interfaces en el software. También permite un visualizar un plano para la implementación y el uso de las técnicas de programación grafica a lo largo de la construcción del producto.

Debido a que la aplicación fue desarrollada usando el CMS Drupal el diagrama de clases de diseño incluye los paquetes que permiten un correcto funcionamiento y configuración del mismo.

El código de Drupal está separado en módulos, cada uno de los cuales define sus propias funciones, que además manejan la inclusión de archivos. La idea es que se cargue la menor cantidad de código por petición para que resulte lo más eficiente posible. En Drupal las funciones son por lo tanto definidas dentro de otras funciones en tiempo de ejecución. Pero PHP no permite este tipo de anidamiento con la declaración de clases, eso significa que los ficheros que definan clases deben estar en el nivel más alto, y no dentro de ninguna función [4].

3.3.1 Estructura de Drupal

Para la realización del diseño del sistema se utilizará Drupal, el cual es un CMS. La configuración típica de un sitio Drupal contiene 4 ficheros principales: *Themes*, *Includes*, *Scripts* y *Modules*. Debido a que cada uno de estos ficheros contiene a su vez gran cantidad de archivos se hará referencia a los mismos con el término **paquete**.

- *Themes*: Contiene las distintas plantillas que permiten configurar el diseño de toda la capa de presentación del sistema; cuando se desee incluir un nuevo diseño, se copia la plantilla dentro de esta carpeta. El nombre del tema utilizado para la aplicación es “Denver” al cual se le realizaron modificaciones para adaptarlo a las características que se buscaban.
- *Includes*: Contiene un conjunto de ficheros indispensables para el funcionamiento de Drupal. Incluye todo un conjunto de implementaciones de clases, que son empleados por los módulos (se encuentran en el también las clase de acceso a datos).

- *Scripts*: Dentro de este se encuentran un conjunto de ficheros que se encargan de manejar el aspecto visual de la aplicación, fundamentalmente archivos CSS y java script.
- *Modules*: En él están ubicados todos los módulos, que permiten extender las distintas funcionalidades del CMS, cuando se desea incorporar un nuevo módulo sólo se tiene que copiar en dicha directorio y activarlo a través de la administración de Drupal.

Los módulos mas importantes que se incluyeron para el desarrollo del sistema fueron: CCK: Permite la creación de tipos de contenido personalizado, así como formularios para luego agregar contenidos del tipo especificado.

- *Views*: Facilita obtener vistas con reportes personalizados de los contenidos existentes, y tablas de contenidos configurando los campos que deben o no ser mostrados.
- *Ldap*: Utilizado para la autenticación de un dominio externo.
- *Print*: Imprime las salidas del sistema.
- *CCK_Permis*: Permite especificar permisos para que los usuarios solo tengan acceso a los campos que se les autorice en dependencia de sus roles.

Drupal básicamente contiene una única página servidor, la misma genera el contenido que se mostrara en la página final, de acuerdo a los argumentos con que se realicen la petición. Estas páginas generadas por Drupal pueden o no contener formularios, esto depende del módulo sobre el que se esté trabajando y el propósito del mismo. En la figura a continuación se representa el funcionamiento de Drupal como aplicación web.

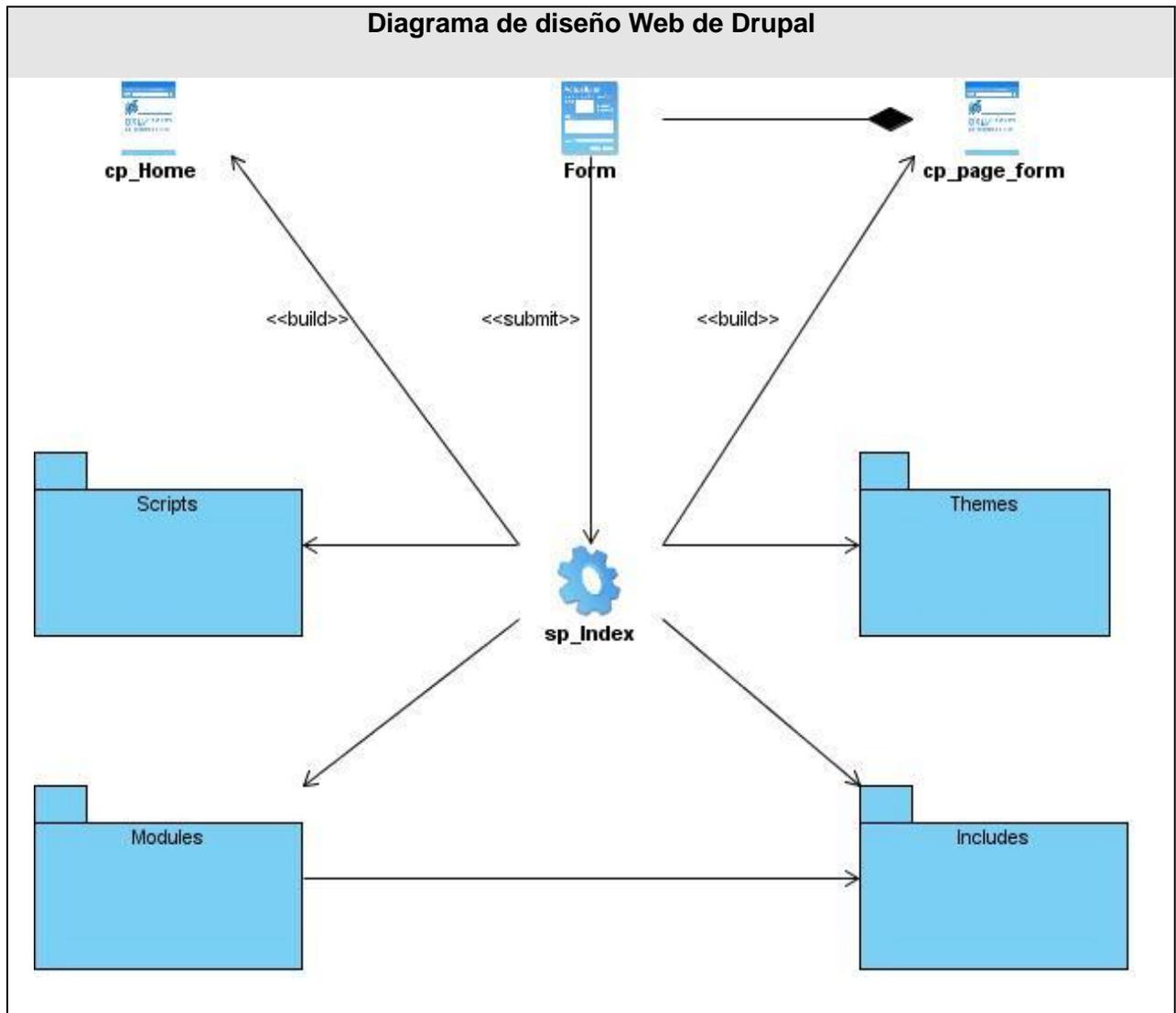


Figura 14 Diagrama de diseño Web de Drupal

Debido a que la mayoría de las funcionalidades de Drupal y de la aplicación desarrollada están integradas en los módulos, el paquete modules es el que se modelará para dar un mejor entendimiento de cómo funciona el sistema.

3.3.1.1 Paquete modules

El término módulo en Drupal se refiere a los archivos de código fuente que implementan las diferentes funcionalidades de la plataforma. Como cada módulo es independiente y diferente en

cuanto a funcionalidad, cada uno de ellos se mostrará por separado. Los módulos no son más que ficheros que poseen una extensión `.module` que contienen funciones escritas en PHP. “Estas funciones actúan como *hooks* (funciones gancho), que son llamadas por Drupal durante sus procesos habituales de gestión de contenido y construcción de páginas Web.”

El diagrama que se presenta a continuación muestra el paquete módulo que se ha usado en la aplicación, con los principales módulos que dan soporte a las clases del diseño de la aplicación.

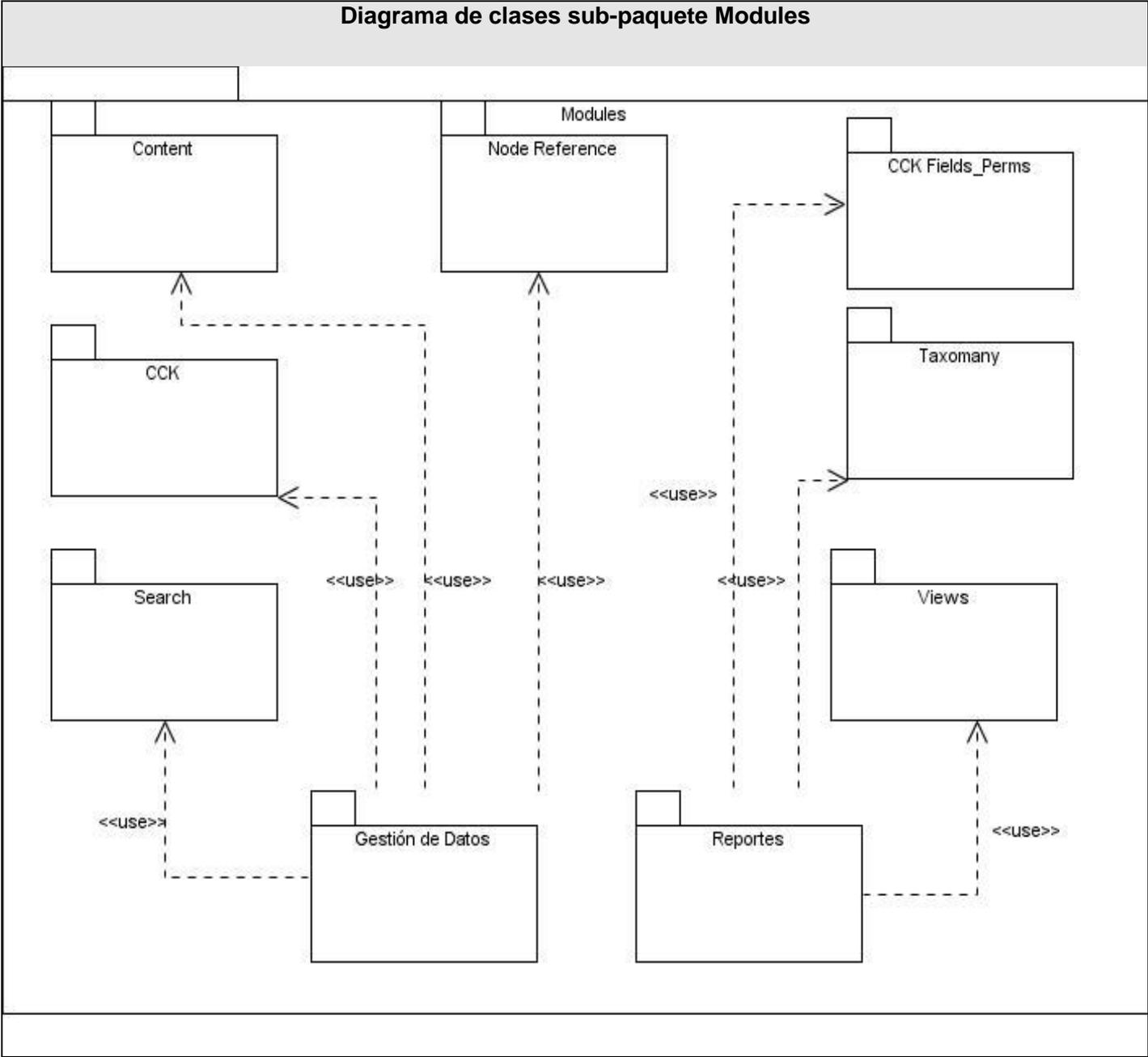


Figura 15 Diagrama de clases sub-paquete Modules

3.3.1.2 Módulos de soporte a usar

Los módulos que se utilizarán en la aplicación son algunos como, *Search*, *CCK (Content Construction Kit)*, *Views*, *Taxonomy*, los cuales se describen a continuación.

Search: Permite la recuperación y visualización de la información referida a contenido, usuarios y otros criterios de búsqueda.

CCK (Content Construction Kit): Permite la creación de campos personalizados y añadirlos a cualquier tipo de contenido definido. Incluye campos como email, dirección, referencia a nodos, y muchos más. Este módulo será el encargado de convertir las clases propuestas en nuevos tipos de contenido de un sitio Drupal, posibilitando además personalizar los campos de estos contenidos, incluyendo relaciones entre los mismos, usando el tipo de dato *node reference*, el cual permite establecer relaciones entre los nodos.

Views: Proporciona un método flexible a los diseñadores para controlar la forma en que se lista el contenido de un sitio Drupal. Este módulo es esencialmente un constructor de consultas inteligentes, que proporcionándole la suficiente información, es capaz de construir la consulta adecuada, ejecutarla y mostrar los resultados. El presente trabajo propone el uso de este módulo basándose en la utilidad que presenta para mostrar listados de contenido y para la obtención de reportes, dos funcionalidades que tienen un peso fundamental en la arquitectura del sistema propuesto.

Taxonomy: Constituye uno de los módulos más fuertes y más populares de Drupal debido a sus características y los beneficios que brinda. Permite a los usuarios autorizados etiquetar contenido usando etiquetas personalizadas y clasificar automáticamente nuevos contenidos basados en esta taxonomía. Esto posibilita que la clasificación y la recuperación de información sean verdaderamente flexibles. En la presente investigación se tendrá en cuenta este módulo para etiquetar el contenido teniendo en cuenta el glosario de términos, lo cual permitirá que los usuarios finales puedan usar la misma terminología del negocio a la hora de realizar reportes o búsquedas.

3.3.2 Paquetes del Diseño propuestos

Un paquete del diseño está básicamente contenido por una colección de diagramas, clases y realizaciones de casos de uso, además puede contener otros paquetes. Su objetivo principal es agrupar los elementos del diseño que estén relacionados en piezas más manejables.

Se agruparon las clases del diseño en dos paquetes principales, teniendo en cuenta lo anteriormente expuesto, para lograr así un mejor entendimiento y simplificar los diagramas. Los paquetes son los siguientes:

- Gestión de datos.
- Reportes.

3.4 Diagrama de clases persistentes

Por lo general las clases persistentes se originan de las clases entidades en el análisis, ya que estas se encargan de modelar la información y el comportamiento asociado a algún fenómeno o concepto, una persona, un suceso o un objeto.

Una vez determinadas las clases entidad del análisis, las clases persistentes se muestran en el siguiente diagrama, son usadas como tipos de contenido en la aplicación.

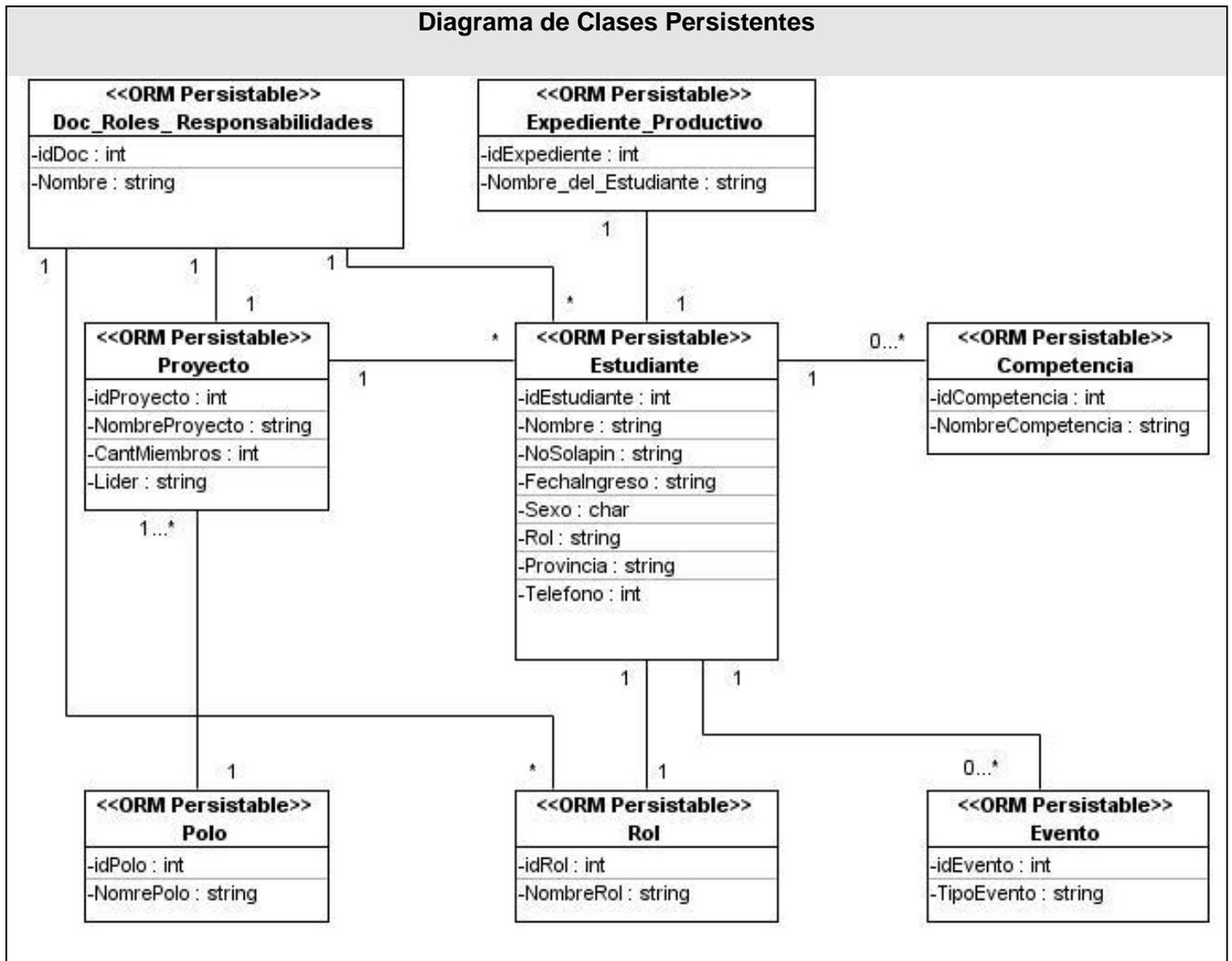


Figura 16 Diagrama de Clases Persistentes

3.5 Modelo de Datos

3.5.1 Diagrama Entidad Relación de la BD

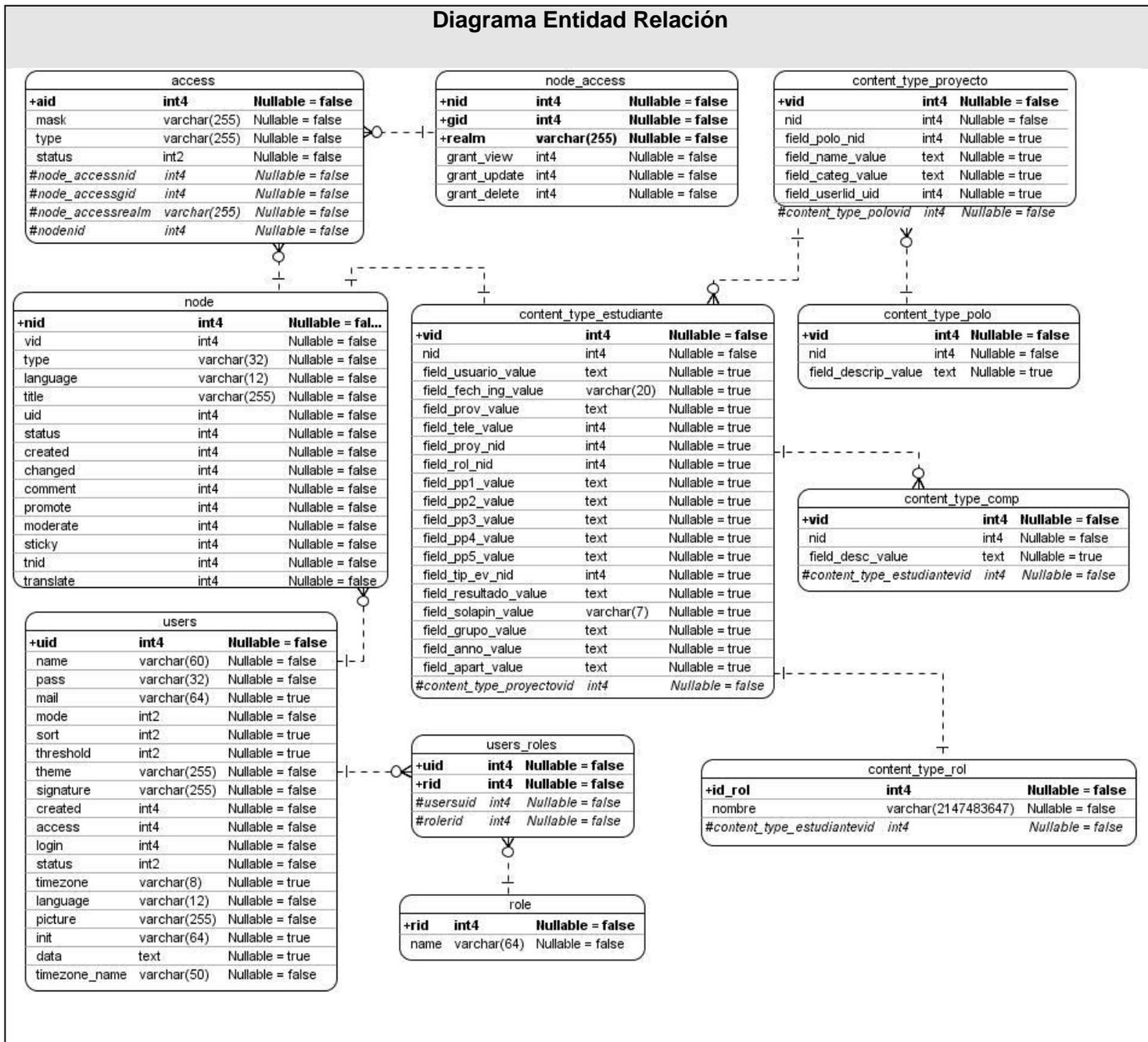


Figura 17 Diagrama Entidad Relación

3.5.1.1 Descripción de las tablas del modelo de datos

| Nombre: node | | |
|---|-------------|---|
| Descripción: Esta es la tabla donde se almacenan todos los contenidos del sistema. | | |
| Atributo | Tipo | Descripción |
| nid | int | Id del contenido. |
| vid | int | Id del vocabulario. |
| type | varchar | Tipo de contenido que guarda el nodo. |
| tittle | varchar | Título del contenido almacenado en el nodo. |
| uid | int | Usuario que posteo el contenido. |
| status | int | Estado del contenido, si está o no publicado. |
| created | int | Fecha de creación. |
| changed | int | Fecha de última actualización. |
| comment | int | Si permite o no el envío de comentarios relacionados con ese tipo de contenido. |
| promoted | int | Si está promocionado o no en la página principal. |
| moderated | int | Si está en la cola de moderación. |
| stiky | int | Si tiene prioridad sobre otros contenidos. |

Tabla 3.0.1 node

| Nombre: user | | |
|---|-------------|--------------------|
| Descripción: Tabla que almacena los datos de los usuarios. | | |
| Atributo | Tipo | Descripción |
| uid | int | Id del usuario |

| | | |
|-----------|---------|--|
| name | varchar | Nombre del usuario |
| pass | varchar | Contraseña |
| mail | varchar | Correo del usuario |
| mode | tyntint | Se guarda el control de los usuarios |
| sort | tyntint | Rol que tiene asignado ese usuario. |
| theme | varchar | Plantilla web definida para ese usuario. |
| signature | varchar | Firma del usuario en caso que estos la registren. |
| created | int | Fecha de creación del usuario en el sistema. |
| access | int | Fecha del último acceso del usuario. |
| login | int | Fecha y hora de la ultima vez que se logueó el usuario en formato Unix |
| status | tyntint | Estado del usuario, si está logueado o no. |
| timezone | varchar | Zona horario que usa el usuario. |
| language | varchar | Idioma definido por o para el usuario. |
| picture | varchar | Foto para el usuario. |

Tabla 3.0.2 user

| | | |
|--|-------------|--------------------|
| Nombre: user_roles | | |
| Descripción: Relaciones entre los usuarios y los roles. | | |
| Atributo | Tipo | Descripción |
| uid | int | Id del usuario. |
| rid | int | Id del rol. |

Tabla 3.0.3 user_roles

| Nombre: role | | |
|-----------------------------------|-------------|--------------------|
| Descripción: Datos del rol | | |
| Atributo | Tipo | Descripción |
| rid | int | Id del rol |
| name | varchar | Nombre del rol |

Tabla 3.0.4 role

| Nombre: access | | |
|---|-------------|--------------------------------------|
| Descripción: Tabla que guarda el tipo de acceso. | | |
| Atributo | Tipo | Descripción |
| aid | int | Id del tipo de acceso |
| mask | varchar | De donde se accedió al sistema. |
| type | varchar | Tipo de acceso. |
| status | int | Estado si han accedido o no al nodo. |

Tabla 3.0.5 access

| Nombre: node_access | | |
|--|-------------|---|
| Descripción: Guarda la relación entre los nodos y los tipos de acceso a los mismos. | | |
| Atributo | Tipo | Descripción |
| nid | int | Id del nodo |
| gid | int | Id del permiso que existe para acceder al nodo. |

| | | |
|--------------|---------|--|
| realm | varchar | Dice si se puede crear o no un nodo. |
| grant_view | int | Si tiene o no permiso para solo ver o leer el nodo. |
| grant_update | int | Si tiene o no permiso para actualizar o editar el nodo. |
| grant_delete | int | Si tiene o no permiso para eliminar el nodo al acceder al mismo. |

Tabla 3.0.6 node_access

| | | |
|--|-------------|---|
| Nombre: content_type_proyecto | | |
| Descripción: Guarda los datos de los proyectos. | | |
| Atributo | Tipo | Descripción |
| vid | int | Id del vocabulario para el proyecto. |
| nid | int | Id del nodo en el cual se guarda el proyecto. |
| field_polo_nid | int | Id del polo al que pertenece el proyecto. |
| field_lider_uid | int | Id del usuario del líder del proyecto. |
| field_categoria_value | text | Categoría del líder del proyecto. |

Tabla 3.0.7 content_type_proyecto

| | | |
|--|-------------|---|
| Nombre: content_type_estudiante | | |
| Descripción: Guarda los datos de los estudiantes. | | |
| Atributo | Tipo | Descripción |
| vid | int | Id del vocabulario |
| nid | int | Id del estudiante o del nodo que guarda ese estudiante. |
| field_proyecto_nid | int | Id del proyecto al que pertenece el estudiante |

| | | |
|--------------------------------------|---------|--|
| field_solapin_value | int | Número del solapín del estudiante |
| field_direccion_particular_value | text | Dirección particular del estudiante. |
| field_telefono_value | int | Número de teléfono del estudiante |
| field_provincia_value | Text | Provincia a la que pertenece el estudiante. |
| field_fecha_de_ingreso_value | varchar | Fecha de ingreso del estudiante al proyecto. |
| field_eventos_nid | int | Evento que se le acredita al estudiante. |
| field_nombre_del_evento_value | text | Nombre de la presentación del estudiante en el evento. |
| field_resultado_obtenido_value | text | Resultado obtenido por el estudiante en un evento. |
| field_tipo_evento_nid | int | Id del tipo de evento. |
| field_practica_profesional_I_value | text | Nota de Práctica Profesional I del estudiante. |
| field_practica_profesional_II_value | text | Nota de Práctica Profesional II del estudiante. |
| field_practica_profesional_III_value | text | Nota de Práctica Profesional III del estudiante. |
| field_practica_profesional_IV_value | text | Nota de Práctica Profesional IV del estudiante. |
| field_practica_profesional_V_value | text | Nota de Práctica Profesional V del estudiante. |
| field_rol_nid | int | Rol del estudiante dentro del proyecto. |

Tabla 3.0.8 content_type_estudiante

| | | |
|--|-------------|--------------------|
| Nombre: content_type_polo | | |
| Descripción: Datos del polo productivo. | | |
| Atributo | Tipo | Descripción |
| nid | int | Id del polo. |

| | | |
|------|---------|------------------------------|
| vid | int | Id del vocabulario del polo. |
| name | varchar | Nombre del polo. |

Tabla 3.0.9 content_type_polo

| | | |
|---|-------------|-----------------------------|
| Nombre: content_type_rol | | |
| Descripción: Datos del rol(rol del estudiante dentro del proyecto) | | |
| Atributo | Tipo | Descripción |
| vid | int | Id del vocabulario del rol. |
| nid | int | Id del rol. |
| name | varchar | Nombre del rol. |

Tabla 3.0.10 content_type_rol

| | | |
|--|-------------|--------------------------------|
| Nombre: content_type_tipoevento | | |
| Descripción: Datos de los tipos de eventos que se le acreditan a los estudiantes. | | |
| Atributo | Tipo | Descripción |
| vid | int | Id del vocabulario del evento. |
| nid | int | Id del tipo de evento. |
| name | varchar | Nombre del tipo de evento. |

Tabla 3.0.11 content_type_tipoevento

| | | |
|--|--|--|
| Nombre: content_type_competencia | | |
| Descripción: Datos de las competencias que se le asignan a los estudiantes. | | |

| Atributo | Tipo | Descripción |
|--------------------------------|---------|--------------------------------------|
| vid | int | Id del vocabulario |
| nid | int | Id de la competencia |
| Field_rol_caracteristico_value | text | Rol al que pertenece esa competencia |
| name | varchar | Nombre de la competencia |

Tabla 3.0.12 content_type_competencia

3.6 Patrones de diseño que se aplican en el sistema

Más allá de clasificar Drupal el CMS en el cual se ha desarrollado el sistema en uno u otro padrón, digamos que se habla en términos de "modelo-vista-controlador" en el sentido de que Drupal hace una excelente separación limpia entre la persistencia de los datos y los formularios (modelo), la lógica (ciclo de respuesta al pedido http, o sea, ciclo de vida de un "request" a un sitio Drupal), y la vista que emerge como la respuesta HTML al explorador del internet (browser). En cada esfera, hay muchas oportunidades para efectuar modificaciones de manera limpia [5].

No obstante, se hace uso de algunos patrones de diseño característicos de sistemas orientados a objetos, particularmente los patrones GOF. Dentro de estos últimos existen varias categorías como son: creacionales, estructurales y de comportamiento, todos ellos contribuyen a la construcción de un diseño más elegante y robusto. Específicamente los patrones que se manifiestan en el funcionamiento de Drupal son: *Singleton*, *Decorator*, *Observer*, *Bridge* y *Chain of Responsibility*, proporcionando al diseño de Drupal una amplia flexibilidad y extensibilidad, haciendo que su funcionamiento tenga características similares al funcionamiento de los sistemas orientados a objetos, lo cual lo convierte en un potente CMS para la construcción de aplicaciones web.

3.7 Seguridad

Drupal posee un sistema de seguridad y acceso basado en roles, que permite controlar el acceso a las diferentes funcionalidades según los permisos que el administrador le haya definido a cada

rol. Teniendo en cuenta que el sistema propuesto va a ser usado por diferentes usuarios del dominio, Drupal debe integrarse con el directorio activo de la universidad mediante el protocolo LDAP. Para lograr esta integración se usará el módulo *LDAP_Integration*, el cual cuenta con todas las funcionalidades necesarias para trabajar con el protocolo requerido.

3.8 Tratamiento de errores

El tratamiento de errores se puede definir como uno de los pasos más importantes para el buen funcionamiento del sistema. Para el tratamiento de errores se tuvo en cuenta los mensajes de texto, explicando de esta forma en qué consiste el error y dando la oportunidad de corregirlo y seguir trabajando fácilmente.

Conclusiones

En este capítulo luego de comprobar la factibilidad del proyecto se comienza el desarrollo de la solución propuesta, al analizar los procesos del negocio y los requisitos que debe tener el sistema, quedan representados los diagramas de análisis del sistema, diagramas de colaboración y del diseño Web, así como el modelo de datos y el diagrama de clases persistentes, los cuales forman parte del diseño de la base de datos del sistema. De esta forma quedan creadas las condiciones para comenzar con la implementación del sistema.

Capítulo 4

Implementación y Prueba

4.1 Introducción

Este capítulo aborda lo referente a los flujos de trabajo Implementación y Prueba, para ello se exponen el diagrama de componente, haciendo una representación de la implementación de las clases de diseño en términos de componente y cómo estos se organizan de acuerdo a los nodos específicos en el modelo de despliegue. Además se realizará un análisis de los casos de prueba para comprobar los errores que puede tener el sistema, corregirlos y obtener un óptimo funcionamiento.

4.2 Diagrama de despliegue

Los diagramas de despliegue se usan para modelar la configuración de los elementos de procesado en tiempo de ejecución y de los componentes, procesos y objetos de software que viven en ellos. Se modelan los nodos físicos y las asociaciones de comunicación que existe entre ellos.

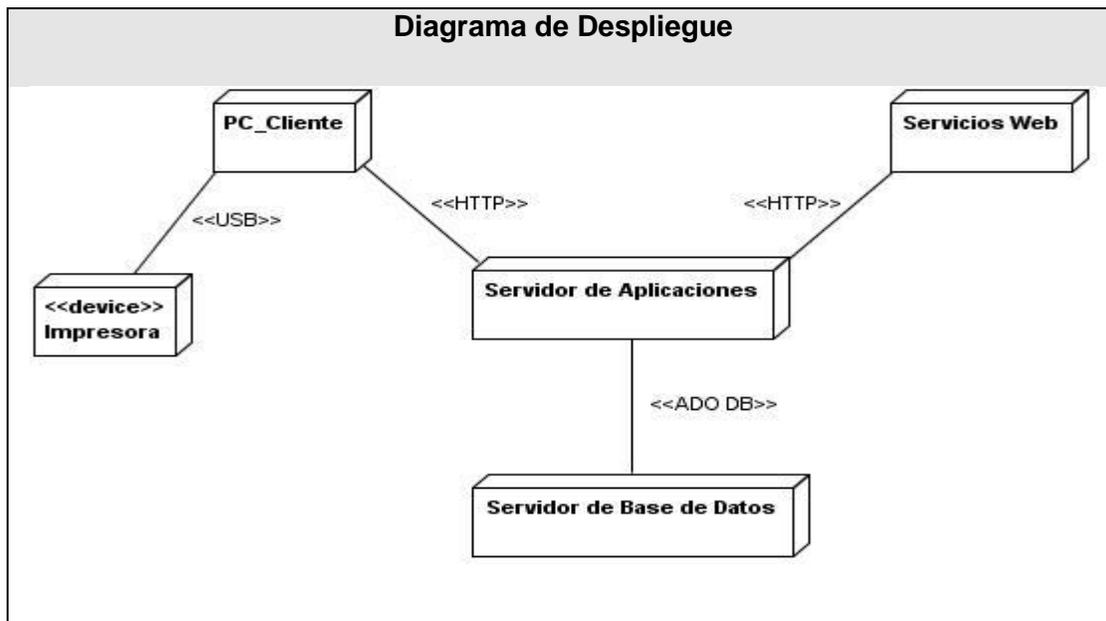


Figura 18 Diagrama de Despliegue

4.3 Diagrama de Componentes

Los diagramas de componentes describen los elementos físicos del sistema y su relación, mostrando las dependencias lógicas entre componentes software. El diagrama de componente hace parte de la vista física de un sistema, la cual modela la estructura de implementación de la aplicación por sí misma, su organización en componentes y su despliegue en nodos de ejecución. La vista de implementación se representa con los diagramas de componentes.

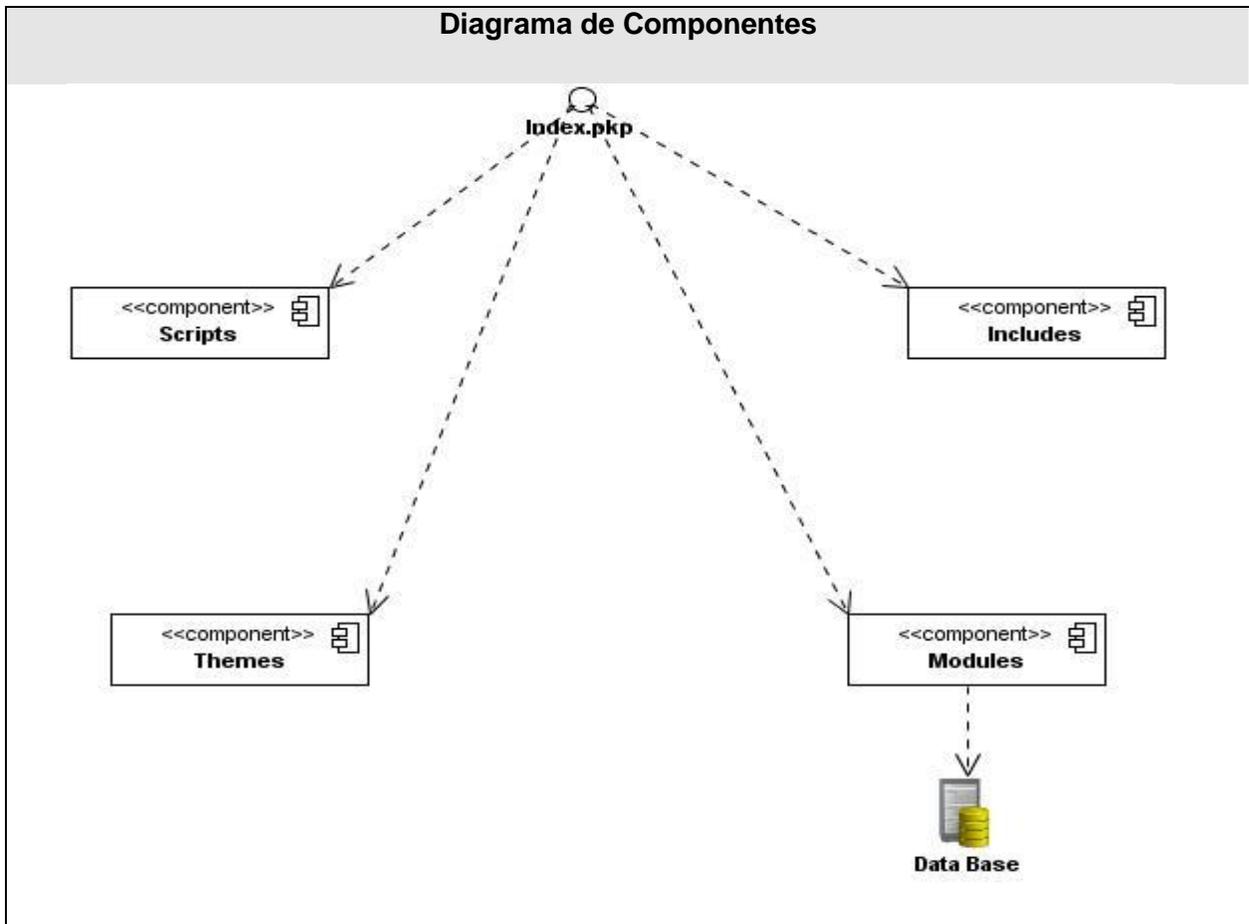


Figura 19 Diagrama de Componentes

4.4 Modelo de Prueba

A continuación se describen los casos de prueba desarrollados de los casos de uso definidos, definiendo los resultados y las condiciones que debe dar el sistema.

4.4.1 Descripción de los casos de prueba de integración

Nombre del caso de uso: Asignar estudiantes a proyecto.

| Entrada | Resultados | Condiciones |
|---|--|--|
| El Vicedecano de producción desea asignar un nuevo estudiante a un proyecto productivo pero no introduce los datos (Nombre completo, solapín) | El sistema emite un mensaje de error informando los datos que este necesita (los campos obligatorios que dejó vacíos). | La operación se repite hasta que el Vicedecano de producción corrige el error. |

Tabla 4.0.1 Descripción del caso de prueba asignar estudiante a proyecto

Nombre del caso de uso: Registrar Notas de Práctica Profesional.

| Entrada | Resultados | Condiciones |
|---|--|---|
| El Líder de Proyecto desea registrar las notas Práctica Profesional de los estudiantes pero no introduce la nota. | El sistema emite un mensaje de error informando los datos que este necesita (los campos obligatorios que dejó vacíos). | La operación se repite hasta que el Líder de Proyecto corrige el error. |

Tabla 4.0.2 Descripción del caso de prueba Registrar Nota de Práctica Profesional

Nombre del caso de uso: Gestionar rol a usuarios.

| Entrada | Resultados | Condiciones |
|--|---|---|
| El Vicedecano de producción desea crear, modificar o eliminar el rol de un usuario específico pero no selecciona el rol deseado. | El sistema emite un mensaje de error informando los datos que este necesita (seleccionar el rol). | La operación se repite hasta que el Líder de Proyecto corrige el error. |

Tabla 4.3 Descripción del caso de prueba Gestionar rol a usuarios

Conclusiones

Se han abordado los flujos de trabajo Implementación y Prueba propuestos, en el flujo de trabajo Implementación se realizó la modelación de los diagramas de Componentes y el de Despliegue donde se especifica la distribución de los sistemas de cómputo que conforman la aplicación con sus características. En el flujo de Prueba se realizaron las pruebas pertenecientes a los casos de uso documentados en el trabajo.

Conclusiones

Con este trabajo la utilidad y validez de emplear las tecnologías informáticas para apoyar las labores que se desarrollan en cualquier tipo de esfera. Una vez terminada la investigación quedan cumplidas las tareas propuestas entre ellas:

- Estudio de la documentación existente sobre la trayectoria productiva de los estudiantes en la facultad 1.
- Análisis y selección de las herramientas que permitan obtener una solución óptima.
- Se realizó el análisis y diseño de un sistema que respalda el estudio anteriormente citado.
- Se implementó la propuesta de solución.

La utilización del sistema implementado permite que el proceso de gestión de la trayectoria productiva de los estudiantes en la facultad 1 sea más eficiente, ya que de este modo es mejor la organización y almacenamiento de los datos involucrados, además logra integrar procesos que hasta el momento se realizaban de forma individual. Hace más fácil y rápido el trabajo para los usuarios implicados y contribuye de esta forma al progreso de la informatización en la universidad.

Recomendaciones

Agregar nuevas funcionalidades teniendo en cuenta las necesidades del Vicedecano de Producción de la facultad, para tener un mayor control de la trayectoria productiva de los estudiantes. Mejorar el diseño de los formularios y así obtener una mejor apariencia visual. Incluir el sistema como un módulo a la intranet de la facultad y que de esta manera queden integradas la trayectoria productiva de los estudiantes, con la gestión de recursos humanos y otras actividades desarrolladas en el área de producción, facilitando así el trabajo de los usuarios.

Referencias bibliográficas

- [1] <http://www.rrhhweb.com/>
- [2] <http://www.tecnosime.co.cu/Pages/gmi.htm>
- [3] <http://www.rational.com.ar/herramientas/rup.html>
- [4] <http://www.virtualizada.com/drupal-desde-la-perspectiva-de-la-poo.htm>
- [5] <http://www.nabble.com/Re:-Arquitectura-Drupal-p16857089.html>

Bibliografía consultada

1. [En línea] <http://infoaleph.wordpress.com/tag/definiciones/page/2/>.
2. [En línea] <http://httpd.apache.org/docs/2.0/es/> .
3. [En línea] <http://www.rational.com.ar/herramientas/rup.html> .
4. [En línea] <http://www.visual-paradigm.com/documentation/> .
5. [En línea] <http://www.postgresql.org/about/press/presskit83.html.es>.
6. [En línea] <http://www.php.net> .
7. [En línea] <http://www.mysql-hispano.org/index.php?m=read&id=2865>.
8. [En línea] <http://drupal.org/press/drupal-6.0/es> .
9. [En línea] <http://api.drupal.org/api/file/CHANGELOG.txt/6/source> .
10. [En línea] <http://drupal.org/project/es> .
11. [En línea] <http://www.oracle.com/database/number-one-database.html> .
12. [En línea] <http://www.visual-paradigm.com/documentation/vpumluserguide2.jsp>.
13. [En línea] <http://www.rational.com.ar/herramientas/roseenterprise.html>.
14. JACOBSON, I. B., G. Y RUMBAUGH, J. . El Proceso Unificado de Desarrollo de software. 2000. p. [En línea] <http://www.uml.org/>.

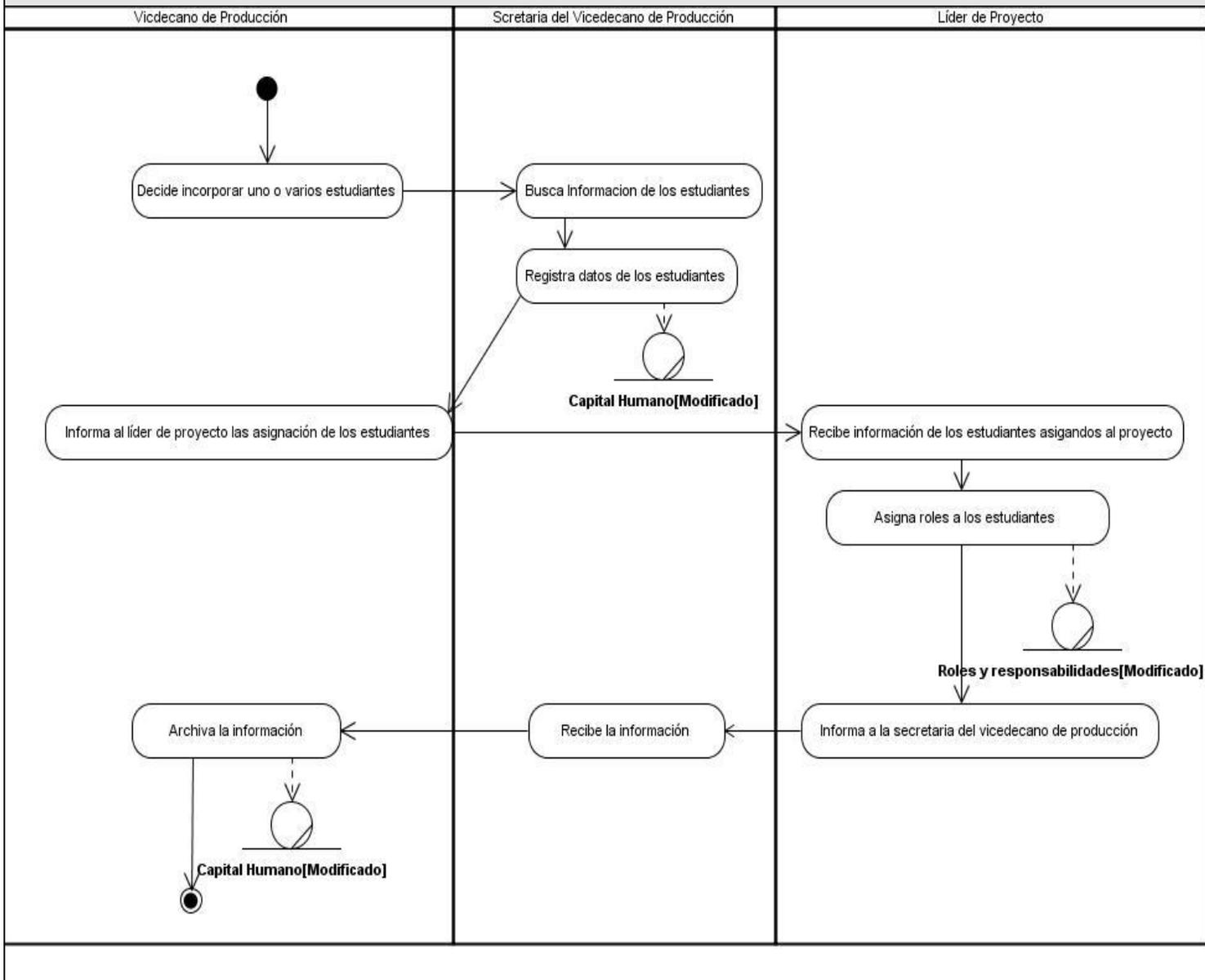
15. Introducción a los Sistemas de Gestión de Contenidos (CMS). [En línea]
<http://www.uoc.edu/mosaic/articulos/cms1204.html>.

Anexos

Anexo I

Diagramas de Actividades de los casos de uso del negocio.

Diagrama de Actividades del Caso de Uso del Negocio "Asignar Estudiante"



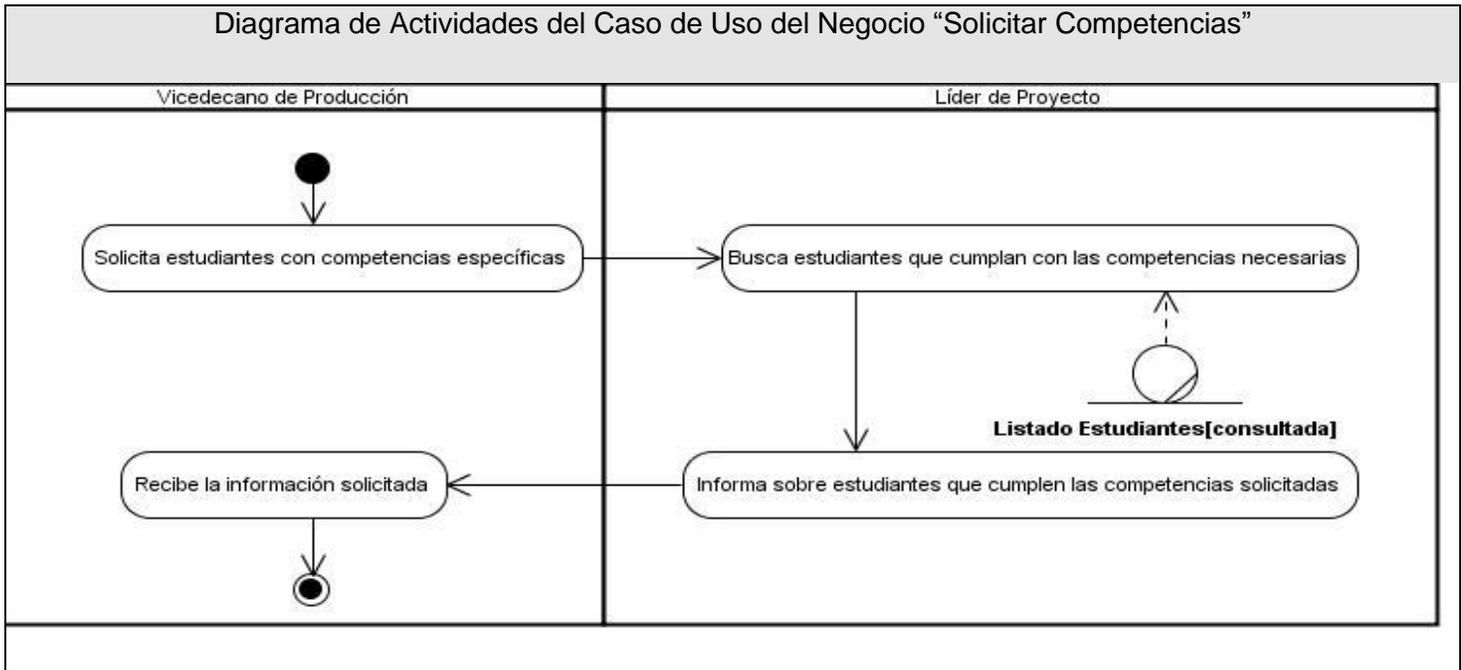


Diagrama de Actividades del Caso de Uso del Negocio “Registrar Curso Optativo”

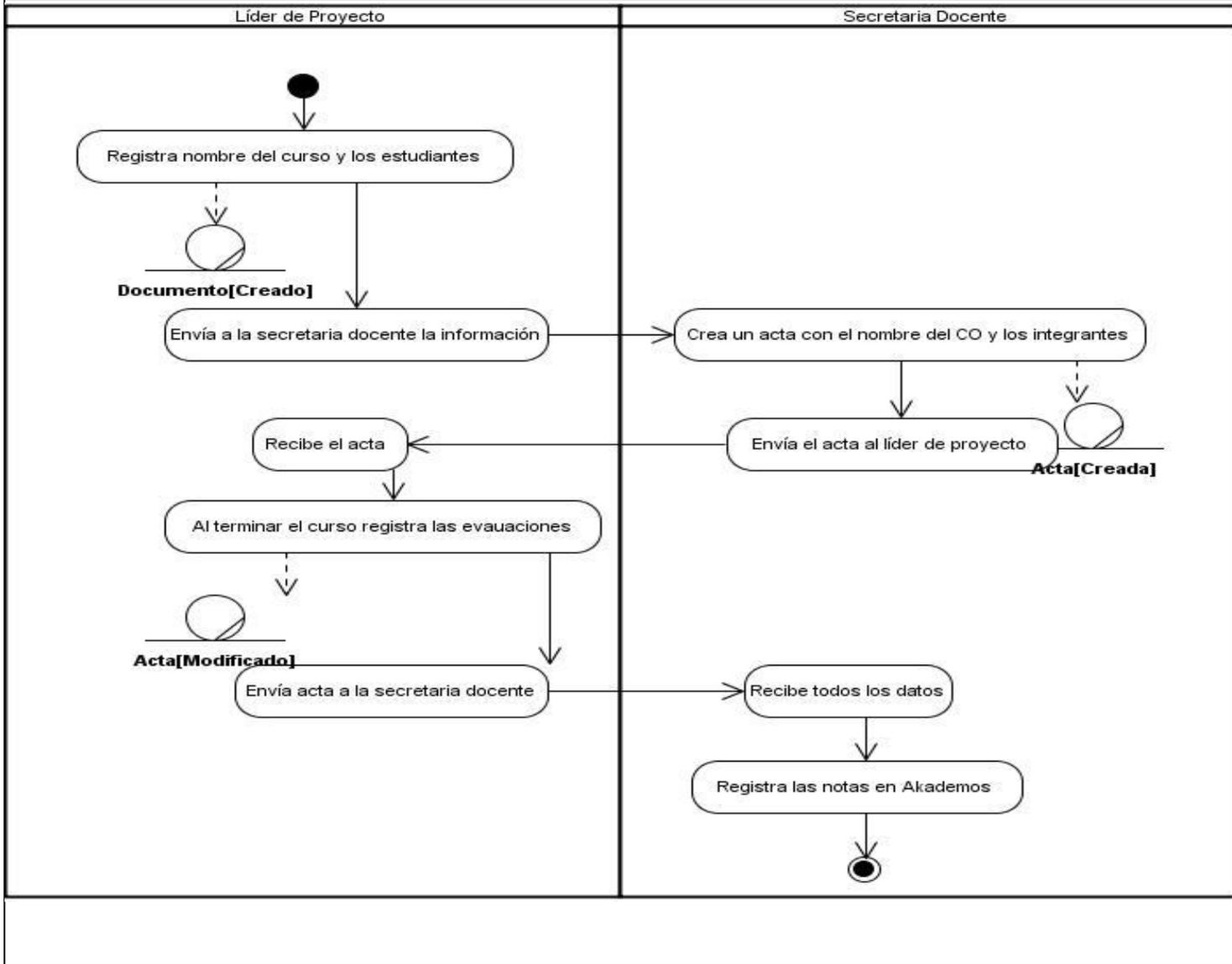
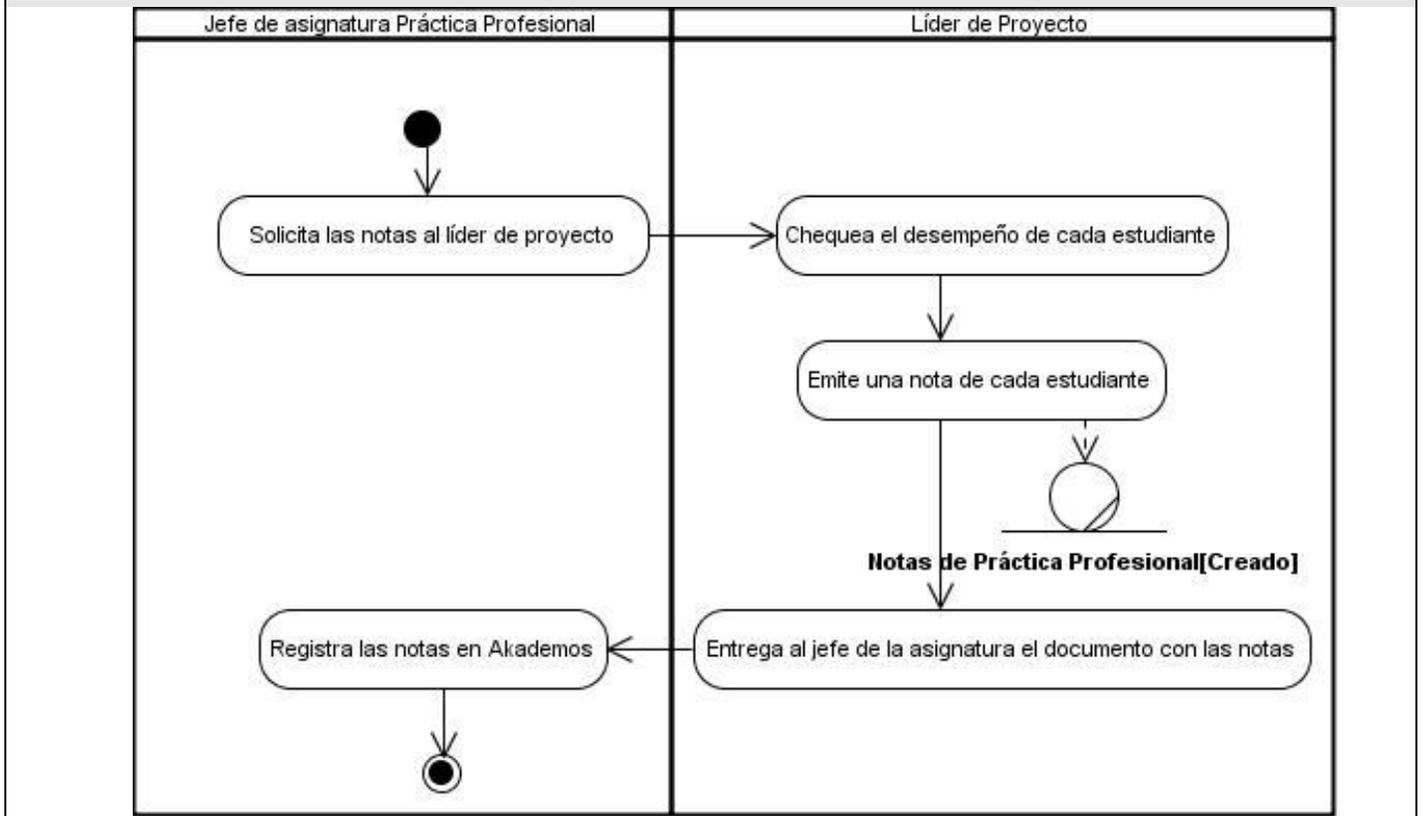


Diagrama de Actividades del Caso de Uso del Negocio “Registrar evaluaciones de Practica Profesional”



Anexo II

Descripción de los casos de uso del sistema

Descripción del caso de uso del sistema “Autenticar Usuario”:

| | | |
|---|--|--|
| Caso de Uso: | Autenticar Usuario | |
| Actores: | Usuario | |
| Propósito: | Permite a los usuarios del sistema identificarse previamente para hacer uso de él, según los permisos que posee y por último, el cierre de sesión al terminar su trabajo. | |
| Resumen: | El caso de uso se inicia cuando el usuario (que puede ser vicedecano, el líder de proyecto o el profesor de práctica profesional) introduce los datos para autenticarse, se verifica que los datos introducidos son correctos. En caso contrario no se permite la autenticación del usuario. | |
| Referencia: | RF1 | |
| Precondiciones: | ----- | |
| Flujo Normal de Eventos | | |
| Acción del Actor | Respuesta del Sistema | |
| 1. El usuario se encuentra frente a la página principal de la aplicación Web. | <p>1.1 El sistema muestra el formulario en el cual el usuario debe introducir los datos necesarios para ser identificado (usuario y contraseña) y así otorgarle los permisos.</p> <p>2.1 El sistema verifica usuario y contraseña introducidos.</p> | |

| | |
|--|--|
| <p>2. El usuario introduce los datos que ha solicitado el sistema (usuario y contraseña para ser identificado como usuario del sistema) y presiona el botón "Aceptar".</p> | <p>2.2 Si los datos introducidos por el usuario se corresponden con los de un usuario registrado en la base de datos el sistema le permite acceder al sistema de acuerdo a los permisos que posea.</p> |
| <p>Flujos Alternos</p> | |
| <p>Acción del Actor</p> | <p>Respuesta del Sistema</p> |
| <p>2. a</p> | <p>2.2 Si los datos introducidos por el usuario no se corresponden con los de un usuario registrado en la base de datos el sistema muestra un mensaje de error y le indica al usuario que no es válido (le deniega el acceso).</p> |
| <p>Pos condiciones:</p> | <p>El usuario queda autorizado para acceder a los recursos del sistema de acuerdo con los permisos que posea.</p> |
| <p>Prioridad:</p> | <p>Critico.</p> |

Descripción del caso de uso del sistema "Asignar estudiantes a proyecto"

| | |
|---------------------|---------------------------------------|
| <p>Caso de Uso:</p> | <p>Asignar estudiantes a proyecto</p> |
| <p>Actores:</p> | <p>Vicedecano de producción</p> |

| | | |
|--|--|--|
| Propósito: | Que el vicedecano de producción pueda asignar estudiantes a los proyectos a la vez que estos datos quedan registrados en la aplicación. | |
| Resumen: | El vicedecano de producción cuando decide asignar uno a varios estudiantes va a la opción insertar estudiantes y el sistema le muestra un formulario donde el vicedecano entra los datos referentes a este estudiante comenzando así la confección del expediente productivo de ese estudiante. | |
| Referencia: | RF5 | |
| Flujo Normal de Eventos | | |
| Acción del Actor | Respuesta del Sistema | |
| <p>1. El vicedecano de producción marca la opción estudiante en el menú insertar.</p> <p>3. El vicedecano de producción entra los datos en el sistema.</p> <p>5. El líder de proyecto pulsa sobre el botón enviar.</p> | <p>2. El sistema le muestra un formulario para que entre toda la información del estudiante, nombre completo, sexo, proyecto en el que va a estar, las competencias que pueda tener ya el estudiante y datos de localización del mismo.</p> <p>4. El sistema muestra las opciones de enviar al final del formulario.</p> <p>6. El sistema carga los datos en su base de datos terminando así el caso de uso.</p> | |
| Flujos Alternos | | |
| Acción del Actor | Respuesta del Negocio | |
| | | |

| | |
|------------------|---|
| Pos condiciones: | Quedan registrado el estudiante en la base de datos y creado su expediente productivo con los datos básicos que debe tener sobre los estudiantes. |
| Prioridad: | Critico |

Descripción del caso de uso del sistema “Obtener estudiantes con una determinada competencia”:

| | | |
|--|---|--|
| Caso de Uso: | Obtener estudiantes con una determinada competencia. | |
| Actores: | Vicedecano de Producción | |
| Propósito: | Permite al vicedecano de producción obtener los estudiantes que tienen competencias alcanzadas en un tema específico de su interés. | |
| Resumen: | Este caso de uso se inicializa cuando el vicedecano de producción solicita a los líderes que informen sobre las competencias alcanzadas por los estudiantes en un tema específico que sea de su interés, luego el líder de proyecto busca en sus estudiantes los que tengan competencias alcanzadas en ese tema y se los entrega al vicedecano de producción. | |
| Referencia: | RF4 | |
| Precondiciones: | | |
| Flujo Normal de Eventos | | |
| Acción del Actor | Respuesta del Sistema | |
| 1. El vicedecano de producción selecciona la opción obtener en el menú reportes. | 2. El sistema muestra las opciones de obtener reportes: | |

| | |
|--|---|
| <p>3. El vicedecano de producción selecciona la opción “Competencias”.</p> <p>5. El vicedecano de producción entra el nombre de la competencia y da clic en el botón enviar.</p> | <ul style="list-style-type: none"> • Estudiantes por proyecto. • Estudiantes por rol. • Notas de Practica Profesional. • Todos los proyectos. • Todos los estudiantes. • Competencias <p>4. El sistema muestra un campo de texto para que entre el nombre de la competencia que desea buscar.</p> <p>6. El sistema muestra los estudiantes que tienen el tipo de competencia buscada.</p> |
| <p>Flujos Alternos</p> | |
| <p>Acción del Actor</p> | <p>Respuesta del Negocio</p> |
| <p>a)</p> | <p>El sistema no encuentra estudiantes con el tipo de competencia deseado.</p> <p>Muestra un mensaje especificando que no existen resultados para esa búsqueda.</p> |
| <p>Pos condiciones:</p> | |
| <p>Prioridad:</p> | <p>Secundario.</p> |

Descripción del caso de uso del sistema “Registrar Notas de Práctica Profesional”:

| | | |
|---|--|--|
| Caso de Uso: | Registrar Notas de Práctica Profesional | |
| Actores: | Líder de Proyecto | |
| Propósito: | Permite al líder de proyecto gestionar con mayor facilidad y rapidez la nota de práctica profesional de cada estudiante del proyecto. | |
| Resumen: | El líder de proyecto se introduce en la aplicación, busca la trayectoria de los estudiantes, para luego dar una evaluación y guardar los datos. | |
| Referencia: | RF7 | |
| Precondiciones: | Que el líder de proyecto esté autenticado en la aplicación. | |
| Flujo Normal de Eventos | | |
| Acción del Actor | Respuesta del Negocio | |
| <p>1. Selecciona la opción Buscar Estudiante en el menú reportes.</p> <p>3. Selecciona el proyecto.</p> <p>5. Selecciona la opción editar nota.</p> | <p>2. El sistema muestra un listado con los distintos proyectos de la facultad.</p> <p>4. El sistema muestra un listado de todos los estudiantes pertenecientes al proyecto.</p> | |

| | |
|------------------|--|
| | 6. Guarda los datos en la base de datos. |
| Flujos Alternos | |
| Acción del Actor | Respuesta del Negocio |
| | |
| Pos condiciones: | |
| Prioridad: | Secundario |

Descripción del caso de uso del sistema “Obtener Notas de Práctica Profesional”:

| | |
|-----------------|--|
| Caso de Uso: | Obtener Notas de Práctica Profesional |
| Actores: | Profesor de Práctica Profesional |
| Propósito: | Permite al Profesor de Práctica Profesional obtener con mayor facilidad y rapidez la nota de práctica profesional de cada estudiante del proyecto. |
| Resumen: | El profesor de práctica profesional ingresa al sistema para obtener las notas de los estudiantes. |
| Referencia: | RF8 |
| Precondiciones: | Que el profesor de práctica profesional esté autenticado en la aplicación. |

| | |
|--|---|
| Flujo Normal de Eventos | |
| Acción del Actor | Respuesta del Negocio |
| 1. Selecciona la opción Notas de Práctica Profesional en el menú reportes. | 2. El sistema muestra un listado con los estudiantes y sus notas de Práctica Profesional. |
| Flujos Alternos | |
| Acción del Actor | Respuesta del Negocio |
| | |
| Pos condiciones: | |
| Prioridad: | Secundario |

Descripción del caso de uso del sistema “Obtener estudiantes por proyecto”:

| | |
|--------------|---|
| Caso de Uso: | Obtener estudiantes por proyecto |
| Actores: | Vicedecano de Producción |
| Propósito: | Permite al vicedecano de producción tener un control más completo de los estudiantes que pertenecen a un proyecto determinado. |
| Resumen: | Este caso de uso se inicia cuando el vicedecano de producción desea obtener los estudiantes que pertenecen a un proyecto determinado. |
| Referencia: | RF11 |

| | | |
|---|--|--|
| Precondiciones: | Que el vicedecano de producción esté autenticado en la aplicación. | |
| Flujo Normal de Eventos | | |
| Acción del Actor | Respuesta del Negocio | |
| <p>1. El vicedecano de producción selecciona la opción obtener en el menú reportes.</p> <p>3. El vicedecano de producción selecciona la opción “Estudiantes por proyecto”.</p> <p>5. El vicedecano de producción selecciona el proyecto y da clic en el botón enviar.</p> | <p>2. El sistema muestra las opciones de obtener reportes:</p> <ul style="list-style-type: none"> • Estudiantes por proyecto. • Estudiantes por rol. • Notas de Practica Profesional. • Todos los proyectos. • Todos los estudiantes. • Competencias <p>4. El sistema muestra una lista con todos los proyectos.</p> <p>6. El sistema muestra los estudiantes que pertenecen al proyecto seleccionado.</p> | |
| Flujos Alternos | | |
| Acción del Actor | Respuesta del Negocio | |
| | | |
| Pos condiciones: | | |

| | |
|------------|------------|
| Prioridad: | Secundario |
|------------|------------|

Descripción del caso de uso del sistema “Obtener estudiantes por rol”:

| | | |
|--|---|--|
| Caso de Uso: | Obtener estudiantes por rol | |
| Actores: | Vicedecano de producción | |
| Propósito: | Permite al vicedecano de producción obtener de forma rápida todos los estudiantes que tienen un rol determinado. | |
| Resumen: | Este caso de uso se inicia cuando el vicedecano desea obtener todos los estudiantes que desempeñen un rol determinado. | |
| Referencia: | RF10 | |
| Precondiciones: | ----- | |
| Flujo Normal de Eventos | | |
| Acción del Actor | Respuesta del Negocio | |
| 1. El vicedecano de producción selecciona la opción obtener en el menú reportes. | 2. El sistema muestra las opciones de obtener reportes: <ul style="list-style-type: none"> • Estudiantes por proyecto. • Estudiantes por rol. • Notas de Practica Profesional. • Todos los proyectos. • Todos los estudiantes. | |

| | |
|--|---|
| <p>3. El vicedecano de producción selecciona la opción “Estudiantes por rol”.</p> <p>5. El vicedecano de producción selecciona el rol.</p> | <ul style="list-style-type: none"> • Competencias <p>4. El sistema muestra una lista con todos los roles.</p> <p>6. El sistema muestra los estudiantes que desempeñan el rol seleccionado.</p> |
| <p>Flujos Alternos</p> | |
| <p>Acción del Actor</p> | <p>Respuesta del Negocio</p> |
| | |
| <p>Pos condiciones:</p> | |
| <p>Prioridad:</p> | <p>Secundario</p> |

Descripción del caso de uso del sistema “Asignar Competencias”:

| | |
|---------------------|--|
| <p>Caso de Uso:</p> | <p>Asignar Competencias</p> |
| <p>Actores:</p> | <p>Líder de Proyecto</p> |
| <p>Propósito:</p> | <p>Permite al líder de proyecto llevar un mejor control de los estudiantes y sus competencias alcanzadas.</p> |
| <p>Resumen:</p> | <p>Este caso de uso se inicia cuando el líder de proyecto decide insertar una o varias competencias a un estudiante.</p> |
| <p>Referencia:</p> | <p>RF3</p> |

| | |
|--|--|
| Precondiciones: | ----- |
| Flujo Normal de Eventos | |
| Acción del Actor | Respuesta del Negocio |
| <p>1. El líder de proyecto selecciona la opción “Insertar Competencia”</p> <p>3. Selecciona las competencias que desea insertarle al estudiante.</p> | <p>2. El sistema muestra un listado de las competencias.</p> <p>4. El sistema guarda los cambios.</p> |
| Flujos Alternos | |
| Acción del Actor | Respuesta del Negocio |
| <p>a)</p> <p>2. Entra el nombre de la competencia que se desea insertar en el campo de texto “Otras competencias” y envía la información.</p> | <p>1. El sistema no tiene registrado el tipo de competencia que se desea añadir.</p> <p>3. Guarda los datos.</p> |
| Pos condiciones: | |
| Prioridad: | Secundario |

Descripción del caso de uso del sistema “Registrar evento”:

| | |
|--------------|-------------------|
| Caso de Uso: | Registrar evento |
| Actores: | Líder de proyecto |

| | | |
|--|---|--|
| Propósito: | Permite al líder de proyecto tener un control más completo de los estudiantes que pertenecen a su proyecto en la participación de eventos científicos e investigativos. | |
| Resumen: | Este caso de uso se inicia cuando el líder de proyecto desea registrar la participación de un estudiante en un evento determinado. | |
| Referencia: | RF6, RF6.3 | |
| Precondiciones: | Que el líder de proyecto esté autenticado en la aplicación. | |
| Flujo Normal de Eventos | | |
| Acción del Actor | Respuesta del Negocio | |
| <p>1. Selecciona la opción Buscar Estudiante.</p> <p>3. Selecciona el estudiante al cual le va a asignar el evento.</p> <p>5. El líder selecciona la opción editar.</p> <p>7. Selecciona la opción eventos.</p> <p>9. Introduce el nombre del trabajo presentado, el tipo de evento y el lugar obtenido.</p> | <p>2. El sistema muestra un listado con los estudiantes del proyecto.</p> <p>4. El sistema muestra la información del estudiante.</p> <p>6. El sistema muestra los datos permitiendo editar los mismos.</p> <p>8. Muestra el un formulario.</p> <p>10. El sistema registra los datos.</p> | |
| Flujos Alternos | | |

| | |
|------------------|--|
| Acción del Actor | Respuesta del Negocio |
| | |
| Pos condiciones: | Queda registrado el nuevo tipo de evento |
| Prioridad: | Secundario |

Descripción del caso de uso del sistema “Registrar incidencias”:

| | |
|---|---|
| Caso de Uso: | Registrar incidencias |
| Actores: | Líder de proyecto |
| Propósito: | Permite al líder de proyecto tener un control mas completo de los estudiantes que pertenecen a su proyecto registrando así las incidencias. |
| Resumen: | Este caso de uso se inicia cuando el líder de proyecto desea registrar alguna incidencia cometida por un estudiante en su proyecto. |
| Referencia: | RF12 |
| Precondiciones: | Que el líder de proyecto esté autenticado en la aplicación. |
| Flujo Normal de Eventos | |
| Acción del Actor | Respuesta del Negocio |
| 1. Selecciona la opción Incidencias en el menú registrar. | 2. El sistema muestra un listado con los estudiantes del |

| | |
|--|--|
| 3. Selecciona la opción incidencias. | proyecto con la opción incidencias. |
| 5. Registra la incidencia y hace clic sobre el botón enviar. | 4. El sistema muestra área de texto en la que el líder describe la incidencia. 6. Guarda los datos en la base de datos. |
| Flujos Alternos | |
| Acción del Actor | Respuesta del Negocio |
| | |
| Pos condiciones: | La incidencia queda registrada en el sistema |
| Prioridad: | Secundario |

Descripción del caso de uso del sistema “Registrar Cursos Optativos”:

| | |
|-----------------|--|
| Caso de Uso: | Registrar Cursos Optativos |
| Actores: | Líder de Proyecto |
| Propósito: | Permite al líder de proyecto tener un control mas completo de los estudiantes que pertenecen a su proyecto. |
| Resumen: | Este caso de uso se inicia cuando el líder de proyecto desea registrar un nuevo curso optativo impartido a estudiantes del proyecto. |
| Referencia: | RF6, RF6.2 |
| Precondiciones: | Que el líder de proyecto esté autenticado en la aplicación. |

| Flujo Normal de Eventos | |
|---|---|
| Acción del Actor | Respuesta del Negocio |
| <p>1. Selecciona la opción Buscar Estudiante.</p> <p>3. Selecciona al estudiante al cual le va a acreditar el curso optativo y da en la opción editar.</p> <p>5. Llena el formulario entrando el nombre del curso optativo y la nota.</p> | <p>2. El sistema muestra un listado con los estudiantes del proyecto.</p> <p>4. El sistema muestra un formulario para entrar los datos del curso.</p> <p>6. Guarda los datos en la base de datos.</p> |
| Flujos Alternos | |
| Acción del Actor | Respuesta del Negocio |
| | |
| Pos condiciones: | El Curso Optativo queda registrado en el sistema. |
| Prioridad: | |

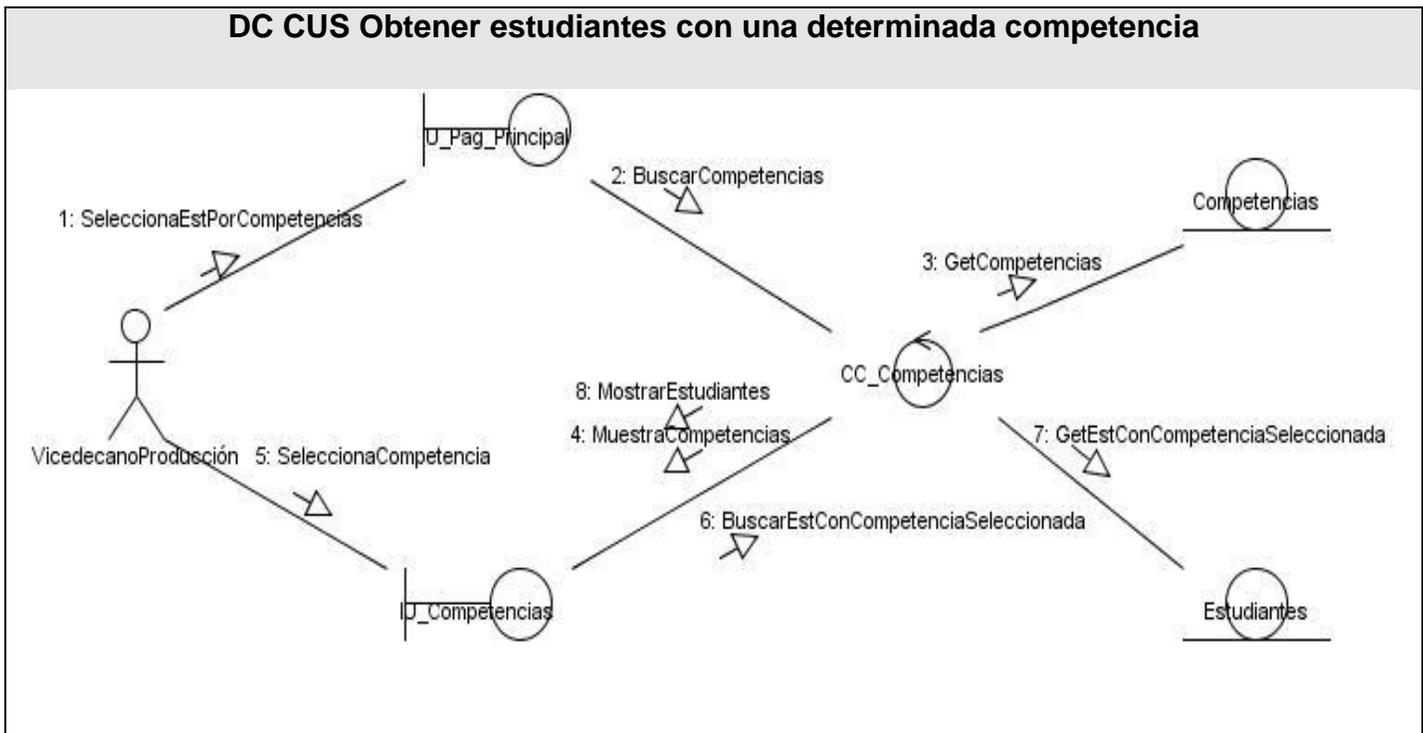
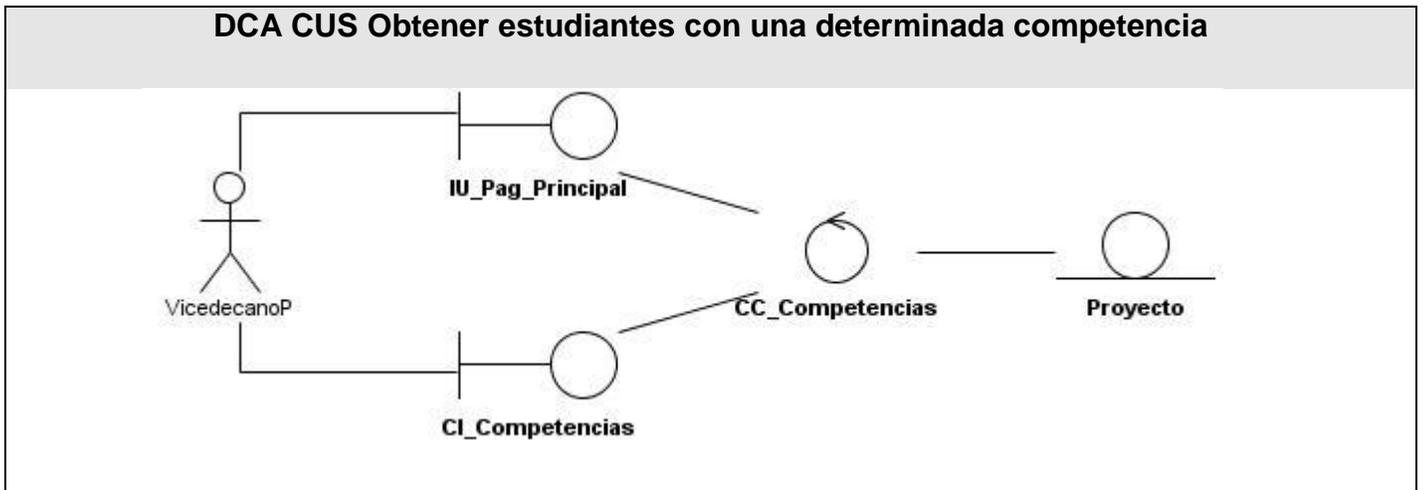
Descripción del caso de uso del sistema “Gestionar roles a usuarios”:

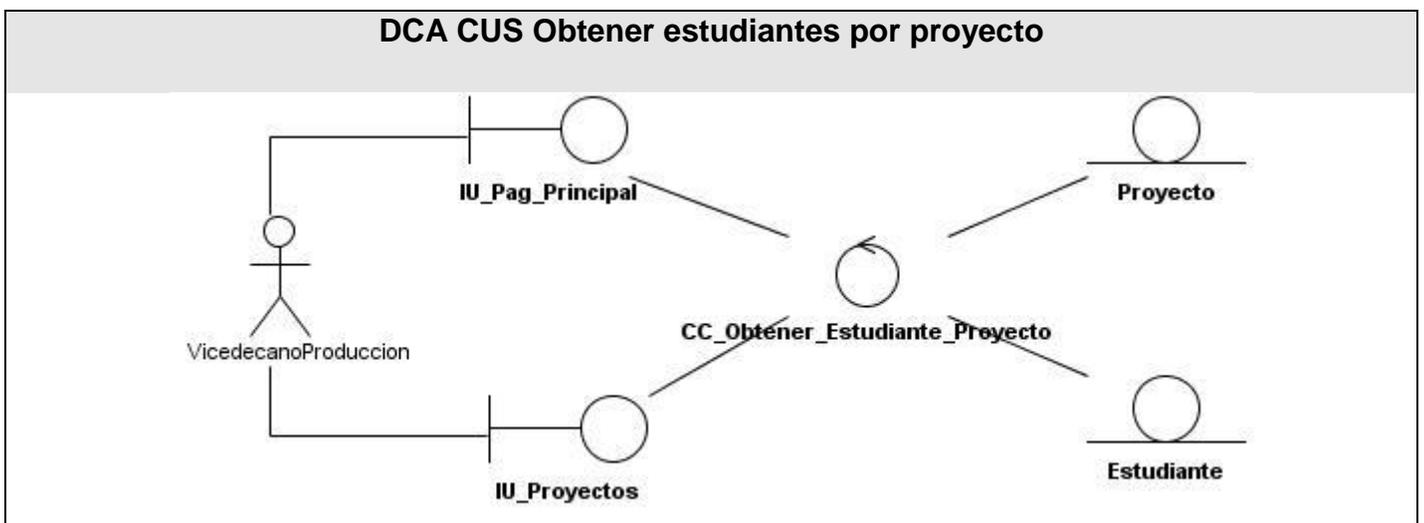
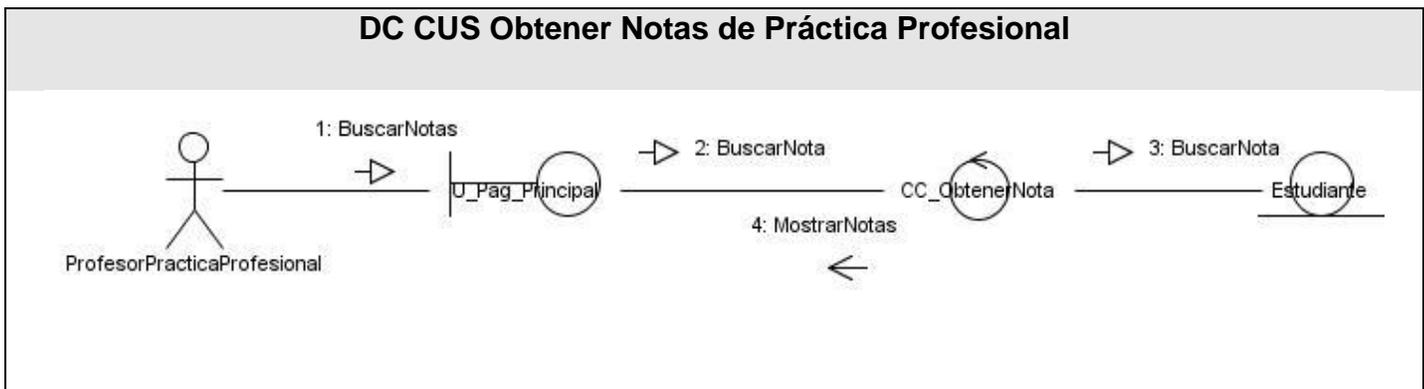
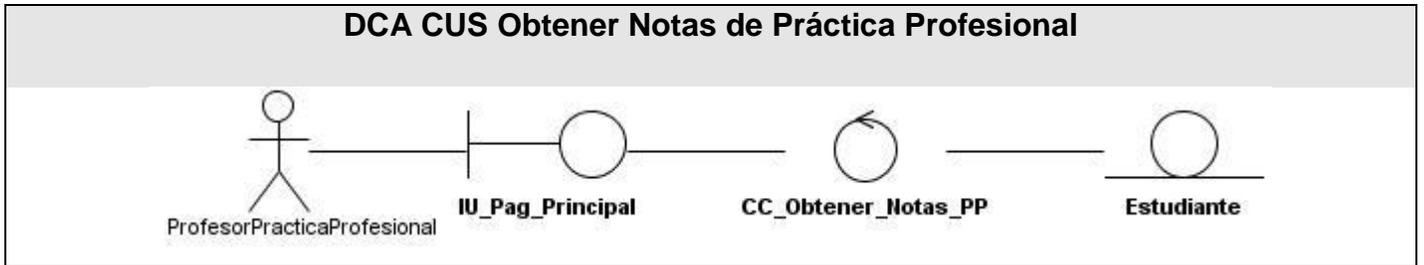
| | |
|--------------|---|
| Caso de Uso: | Gestionar roles a usuarios |
| Actores: | Vicedecano de producción |
| Propósito: | Permite al Vicedecano de producción asignar roles a los diferentes usuarios que |

| | | |
|---|---|--|
| | interactúan con el sistema. | |
| Resumen: | Este caso de uso se inicia cuando el Vicedecano de producción va a delegar un rol a un usuario determinado. | |
| Referencia: | RF2 | |
| Precondiciones: | Que el Vicedecano de producción esté autenticado en la aplicación y que el usuario al cual se le asigne el rol este registrado en la base de datos. | |
| Flujo Normal de Eventos | | |
| Acción del Actor | Respuesta del Negocio | |
| <p>1. Selecciona la opción gestionar usuario.</p> <p>3. Selecciona el usuario deseado.</p> <p>4. En el componente “Opciones de actualización” selecciona la opción y hace clic sobre el botón actualizar.</p> | <p>2. El sistema muestra un listado con los usuarios actuales del sistema.</p> <p>5. Guarda los datos en la base de datos.</p> | |
| Flujos Alternos | | |
| Acción del Actor | Respuesta del Negocio | |
| | | |
| Pos condiciones: | El usuario queda guardado en el sistema con el nuevo rol asignado. | |
| Prioridad: | Primario | |

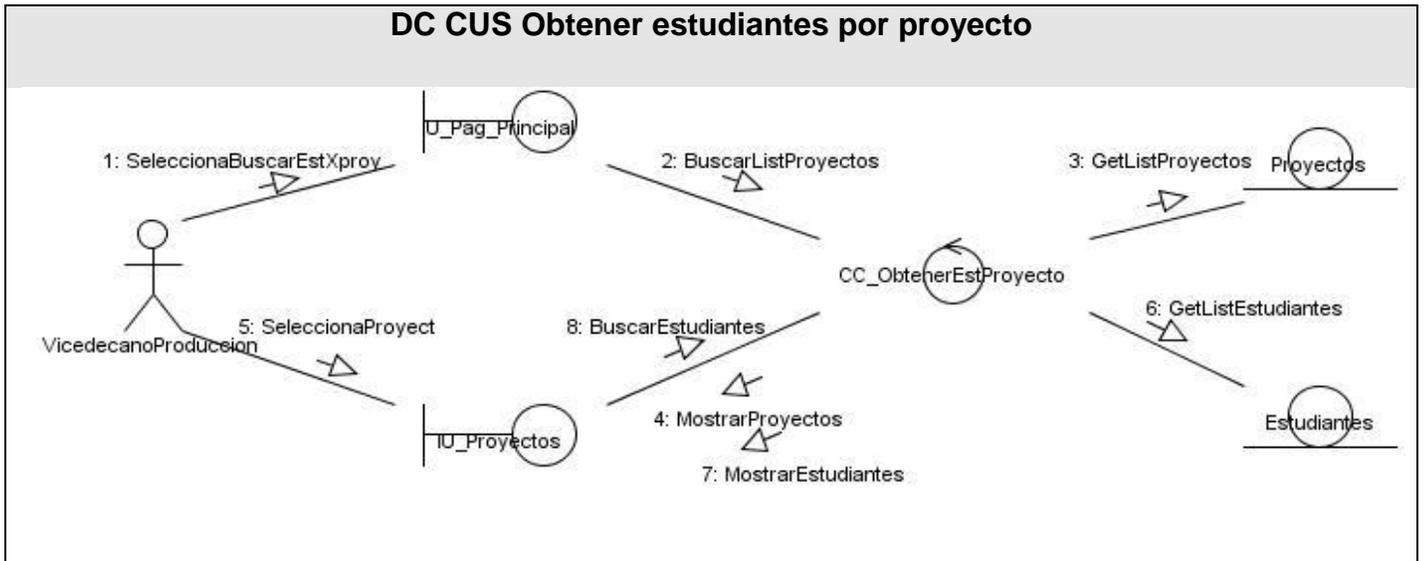
Anexo III

Diagramas de Clases de Análisis y de Colaboración.

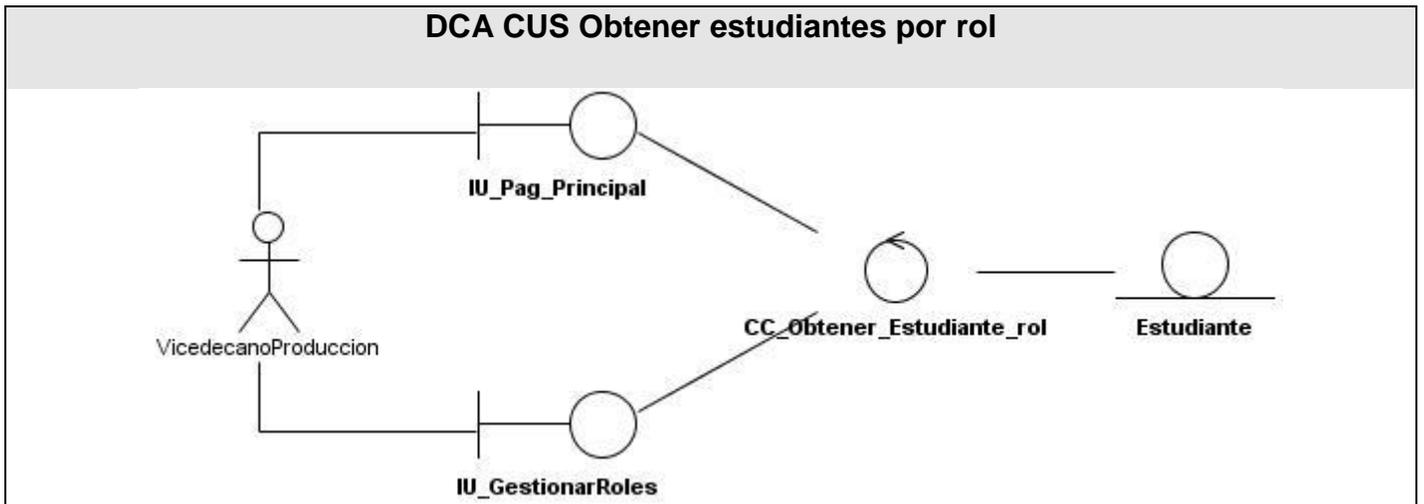


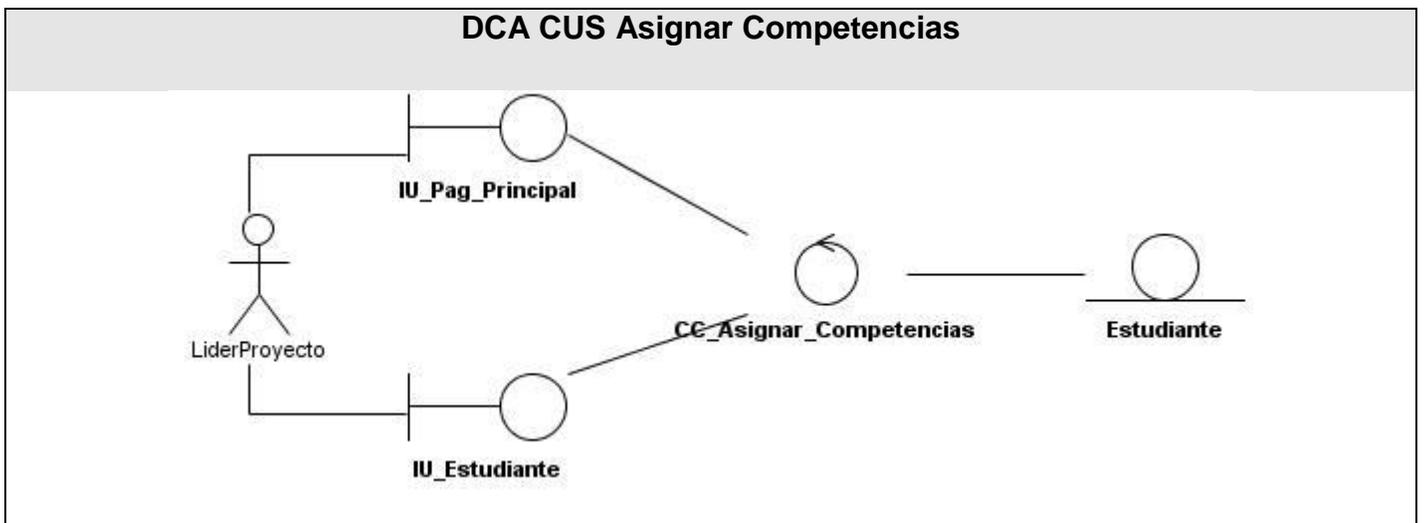
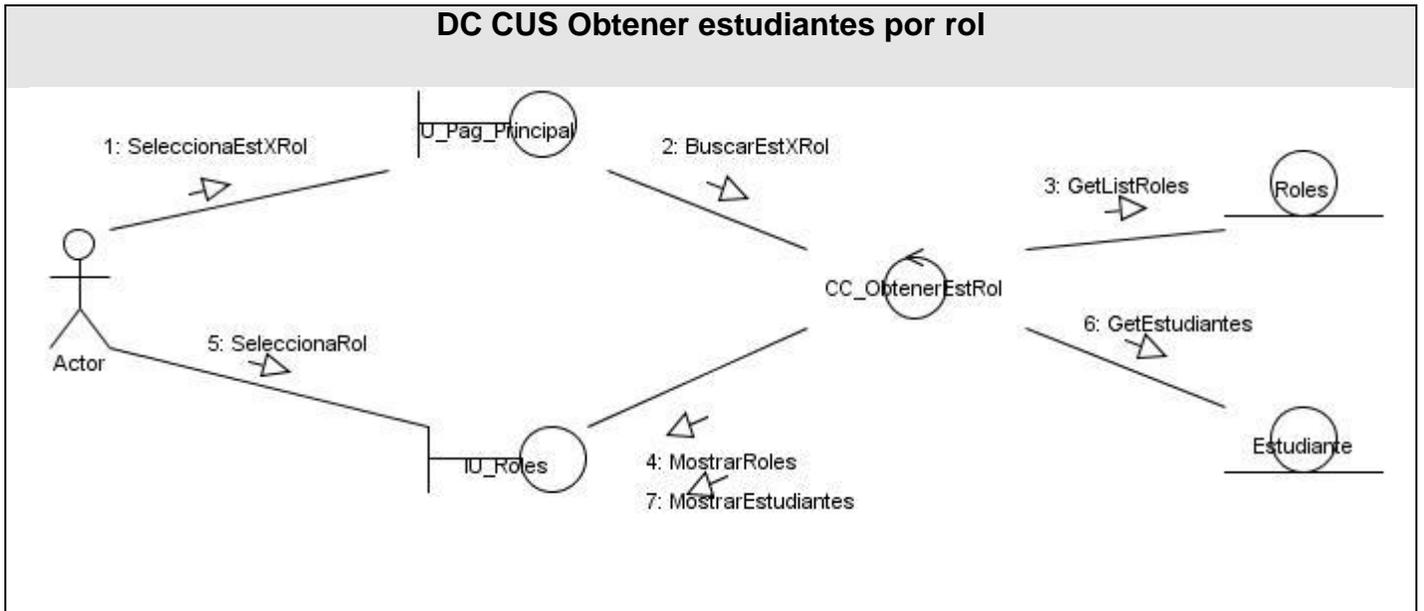


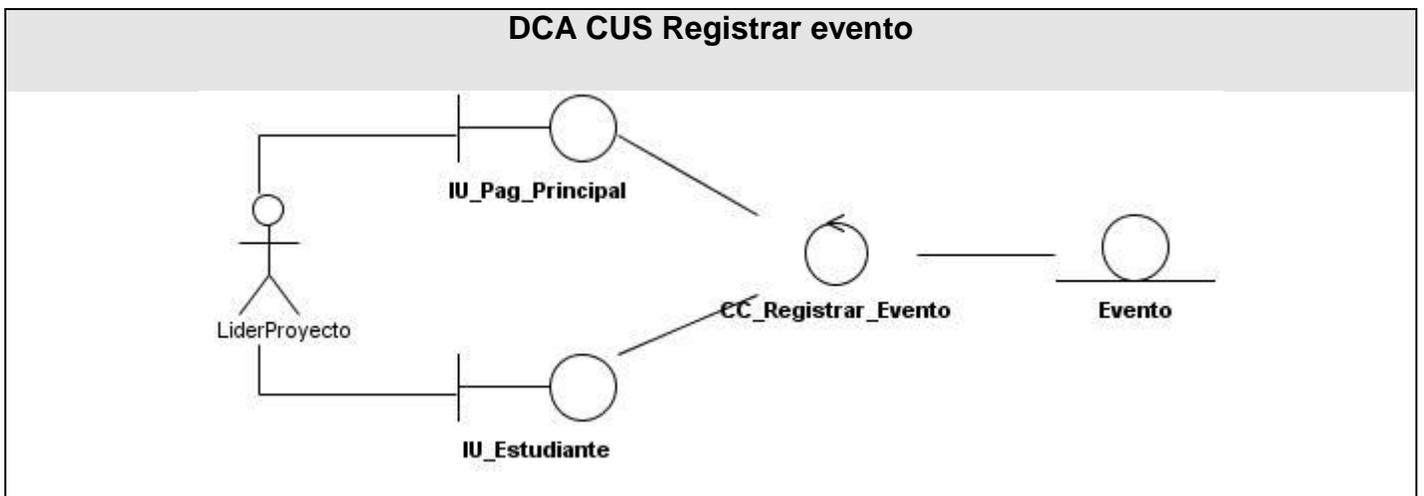
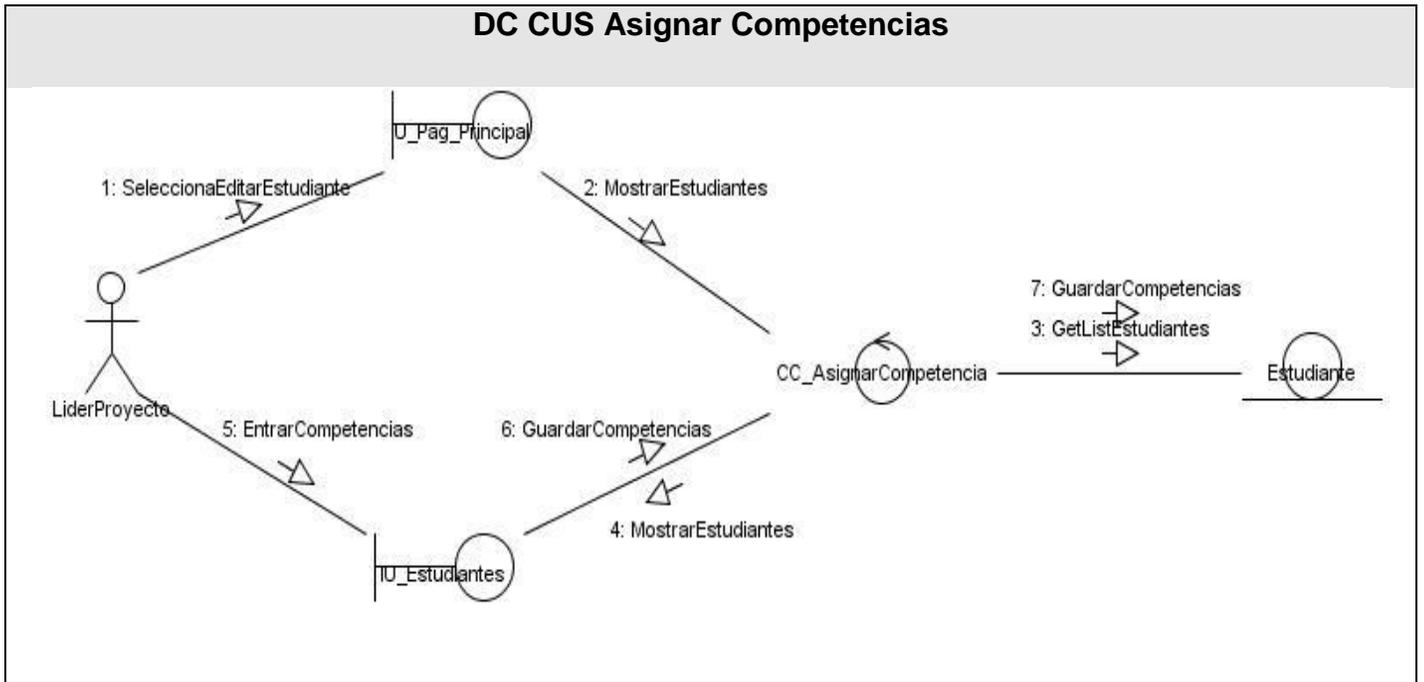
DC CUS Obtener estudiantes por proyecto

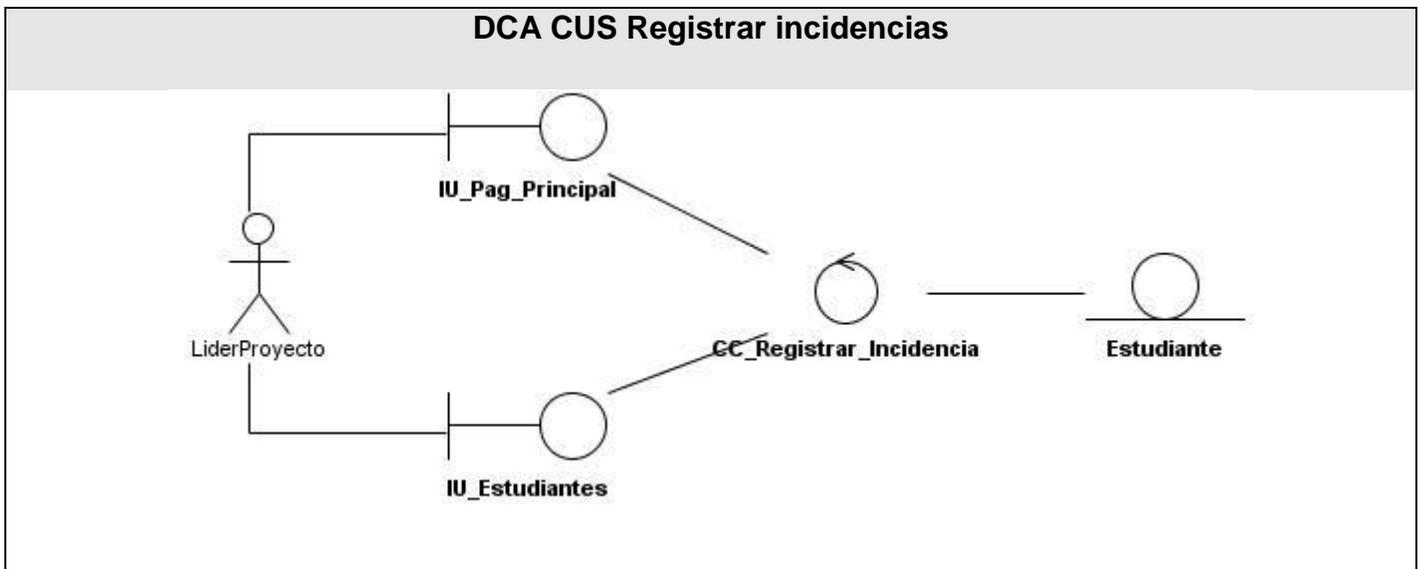
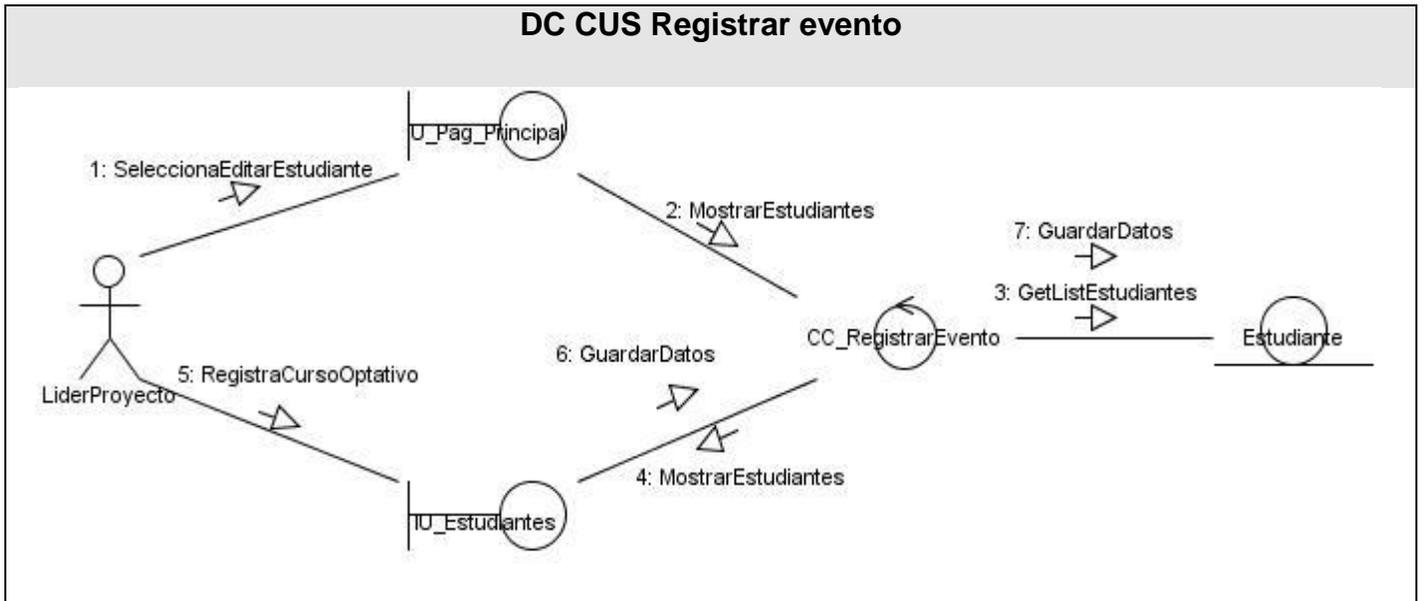


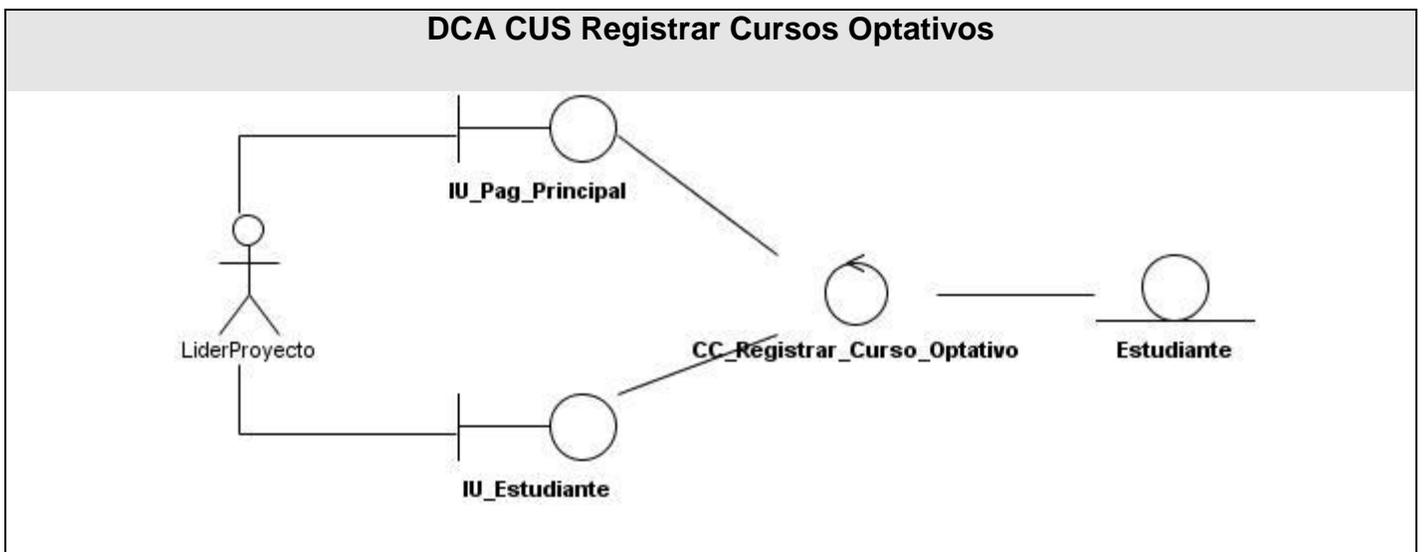
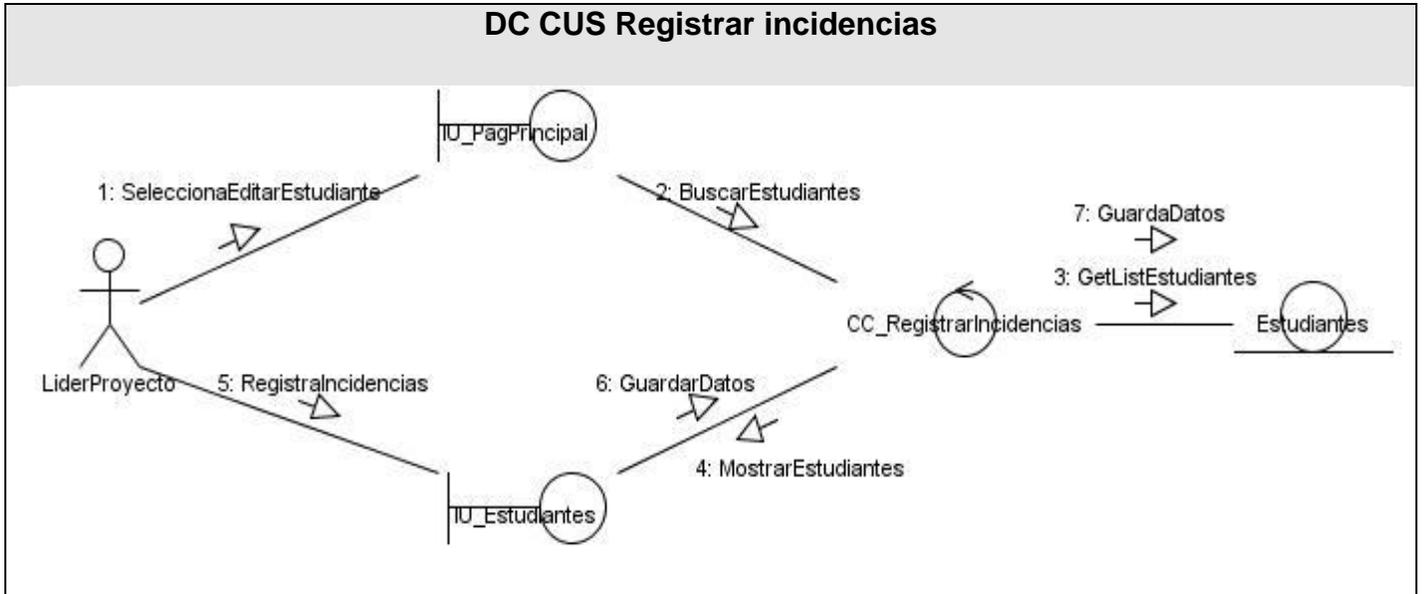
DCA CUS Obtener estudiantes por rol



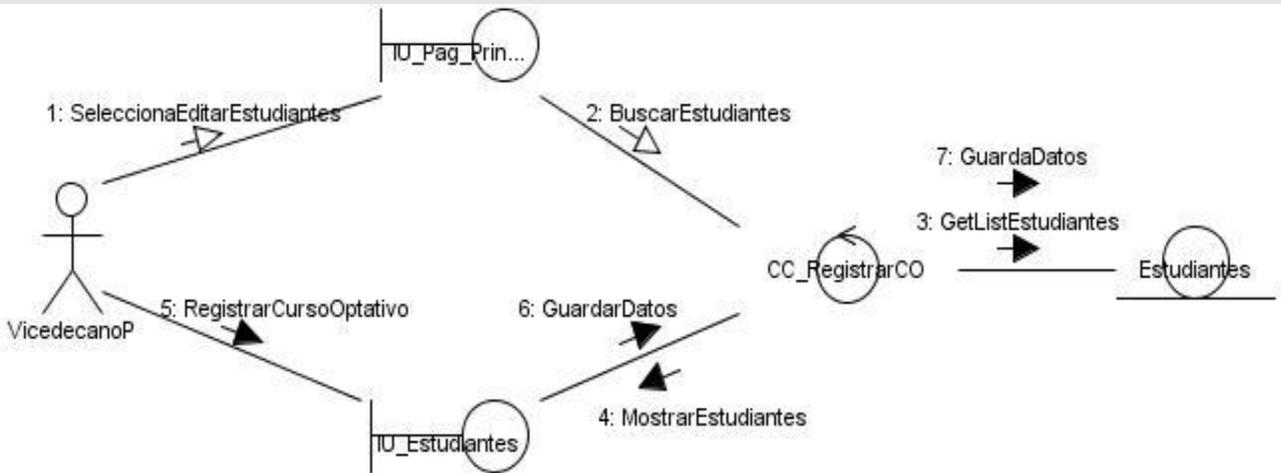




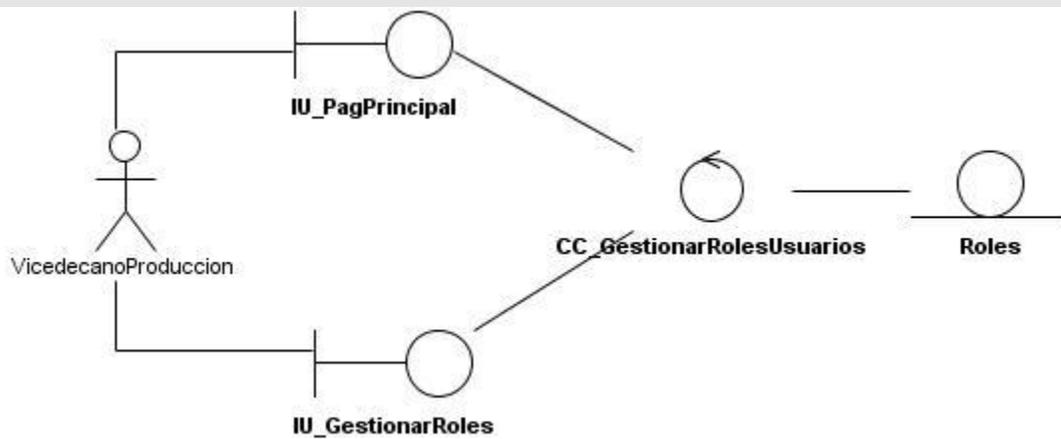


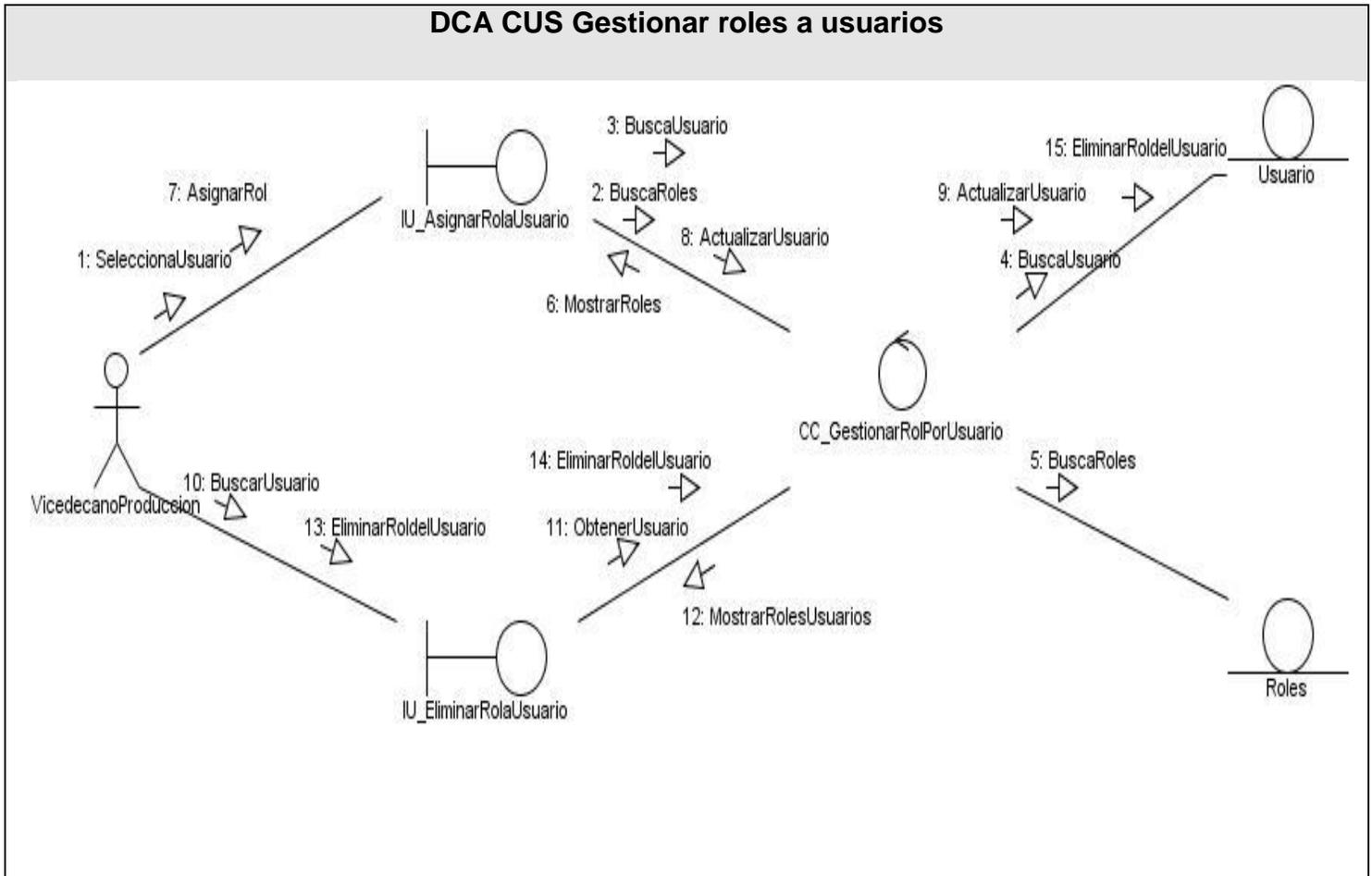


DC CUS Registrar Cursos Optativos



DCA CUS Gestionar roles a usuarios





Glosario de Términos

Proceso: Un proceso se define como un conjunto de tareas, actividades o acciones interrelacionadas entre sí.

Producto: Es cualquier cosa que puede ser ofrecida al mercado para su compra.

Proyecto: Esfuerzo temporal, dirigido a crear un producto, servicio o resultado único.

Requerimiento: Una condición o capacidad que debe estar presente en el sistema o componentes del sistema para satisfacer un contrato estándar, especificación u otro documento formal.

RUP: RUP es una metodología que describe a gran detalle todas las actividades, roles, responsabilidades, productos de trabajo y herramientas para definir quién hace qué y en qué momento en un proyecto de desarrollo de software.

Taxonomía: Las Taxonomías son las clasificaciones ordenadas de elementos de acuerdo a sus relaciones supuestas.