

Universidad de Cienfuegos “Carlos Rafael Rodríguez ”
Facultad de Informática

Trabajo para optar por el título de Ingeniero en Informática

Título: Desarrollo de un prototipo de intranet para una Facultad de un Centro de Educación Superior: Módulo Docente.



*Autores: Daifyn Sosa López
Kadir Hector Ortiz*

*Tutores: Dr. Sc. Ernesto R. Fuentes Garí
Mcs. Lic. Dionel A. Acosta Duarte.
Ing. Rafael Velázquez Fuster*

***“Caminante, son tus huellas
el camino, y nada más;
caminante no hay camino
se hace camino al andar”***

Antonio Machado

A nustras Familias

Índice

<i>Introducción.....</i>	<i>1</i>
<i>Capítulo I: Fundamentación Teórica.....</i>	<i>4</i>
I.1. Principales conceptos asociados al dominio del problema.	4
I.2. Intranet.....	4
I.2.1. Razones para implantar una Intranet.....	5
I.2.2. Características de una Intranet	6
I.2.3. Seguridad en una intranet.....	6
I.2.3.1. Determinando los niveles de seguridad.....	10
I.2.3.2. Administración de la Intranet: Una función crítica.....	10
I.3. Objeto de estudio	12
I.4. Situación problemática	12
I.5. Problema.....	12
I.6. Idea a defender	13
I.7. Metodologías utilizadas.....	13
I.7.1. Lenguaje de Modelación Unificado (UML).....	13
I.7.2. Proceso Unificado de Desarrollo (RUP)	14
I.8. Paradigmas de programación.....	15
I.8.1. Programación por capas.	15
I.9. Tecnologías a utilizar.....	17
I.9.1. Interfaz de usuario (Capa de Presentación).....	17
I.9.2. Lógica del negocio (Capa de la Lógica del Negocio)	18
I.9.2.1. Lenguajes de programación web.....	18
I.9.2.2. Servicios web	20
I.9.2.3. SOAP.....	21
I.9.2.4. XML	22
I.9.2.5. Servidores de Aplicaciones Web.	24
I.9.3. Capa de acceso a datos.	25
I.10. Conclusiones del Capítulo	27
<i>Capítulo II: Modelo del negocio.</i>	<i>29</i>

II.1. Descripción del modelo de negocio.....	29
II.1.1. Descripción del proceso de negocio	29
II.1.2. Reglas del negocio	31
II.2. Modelo de casos de uso del negocio	32
II.2.1. Actores del negocio	33
II.2.2. Diagramas de casos de uso del negocio.....	33
II.2.3. Trabajadores del negocio	35
II.2.4. Realización de los casos de uso del negocio.....	35
II.3. Diagramas de Actividades.....	44
II.4. Diagrama de Clases del Modelo de Objetos.....	52
II.5. Conclusiones del Capítulo.....	53
<i>Capítulo III: Modelo de Sistema.</i>	54
III.1. Descripción General del Modelo de Sistema	54
III.1.1. Requisitos Funcionales	54
III.1.2. Requisitos no funcionales.....	56
III.2. Modelo de Casos de Uso del sistema.....	59
III.2.1. Actores del Sistema	59
III.2.2. Casos de Uso del Sistema.....	60
III.2.3. Descripción de los Casos de Uso.....	67
III.3. Conclusiones del Capítulo	82
<i>Capítulo IV: Diseño e Implementación del Sistema</i>	83
IV.1. Principios de Diseño del Sistema.....	83
IV.1.1. Estándares en la Interfaz de la Aplicación	83
IV.1.2. Formatos de Reportes.....	84
IV.1.3. Concepción General de la Ayuda.....	84
IV.1.4. Tratamiento de Excepciones	84
IV.2. Modelo de Clases Web	85
IV.3. Diseño de la Base de datos	86
IV.4. Diagrama del modelo lógico de datos.....	87
IV.5. Diagramas del modelo físico de datos.	87
IV.6. Diagramas de Implementación	87

IV.7. Conclusiones del Capítulo	88
<i>Conclusiones</i>	89
<i>Recomendaciones</i>	90
Anexos.	

Índice de Tablas

Tabla 1.	Comparación entre SGBD.....	26
Tabla 2.	Actores del negocio	33
Tabla 3.	Trabajadores del negocio	35
Tabla 4.	Actores del Sistema.....	60
Tabla 5.	Diagrama de Clases. Paquete Módulo General	85
Tabla 6.	Diagrama de Clases. Paquete de Matrícula	85
Tabla 7.	Diagrama de Clases Web: Paquete Codificadores.....	86
Tabla 8.	Diagrama de Clases Web: Paquete Evaluaciones	86
Tabla 9.	Diagrama de Clases Web: Paquete Reportes	86
Tabla 10.	Diagrama de Clases Web: Paquete Administrativo	86

Índice de Figuras

Fig. 1.-	Diagrama de Casos de Uso del Negocio.....	34
Fig. 2.-	Diagrama de Actividad. Caso de Uso Solicitar Matrícula	45
Fig. 3.-	Diagrama de Actividad. Caso de Uso Solicitar Licencia.....	46
Fig. 4.-	Diagrama de Actividad. Caso de Uso Solicitar Baja.....	47
Fig. 5.-	Diagrama de Actividad Caso de uso Solicitar Traslado.....	48
Fig. 6.-	Diagrama de Actividad. Caso de Uso Solicitar Cambio de Carrera.....	49
Fig. 7.-	Diagrama de Actividad. Caso de Uso Solicitar Re-Matrícula.....	50
Fig. 8.-	Diagrama de Actividad. Caso de Uso Actualizar Evaluaciones.....	51
Fig. 9.-	Diagrama de Actividad. Caso de Uso Solicitar Reportes.....	52
Fig. 10.-	Diagrama de Clases del Modelo de Objetos	53
Fig. 11.-	Diagrama de Casos de Usos por Paquetes	61
Fig. 12.-	Paquete: Módulo Docente	61
Fig. 13.-	Diagrama de caso de Uso del Sistema: Paquete Módulo General.....	62
Fig. 14.-	Diagrama de caso de Uso del Sistema: Paquete Matrícula	63
Fig. 15.-	Diagrama de caso de Uso del Sistema: Paquete Codificadores	63
Fig. 16.-	Diagrama de caso de Uso del Sistema: Paquete Evaluaciones.....	64
Fig. 17.-	Diagrama de caso de Uso del Sistema: Paquete Reportes.....	65
Fig. 18.-	Diagrama de caso de Uso del Sistema: Paquete Administrativo.....	66
Fig. 19.-	Diagrama de Implementación.....	88

Resumen

El presente trabajo está orientado a desarrollar un prototipo de Intranet para una facultad de un Centro de Enseñanza Superior. Con dicho sistema se pretende agilizar el proceso de manipulación de la información requerido por los diferentes usuarios de esta entidad.

Como antecedente del producto existe una intranet, que aunque se basaba en el concepto cliente/servidor, no hace uso de los servicios web como modelo base para la gestión de la información, lo cual trae como consecuencia que presente algunas limitaciones.

A raíz de la profundización en el conocimiento del problema, la versión que se propone está basada en la arquitectura cliente/servidor, mediante el empleo de la web, y utiliza como base el paradigma de los servicios web lo que contribuye a eliminar los problemas de la versión anterior.

Fue analizado el flujo de información y trabajo de la Facultad de Informática de la Universidad de Cienfuegos para el desarrollo de la primera versión del prototipo de la Intranet que se propone.

Palabras Claves: Intranet, servicios web, servidor, cliente

Introducción

El manejo de la información dentro de cualquier entidad resulta vital para la toma de decisiones y a la vez implica una alta responsabilidad teniendo en cuenta elementos como la oportunidad, fiabilidad y seguridad. Si esta información debe manejarse en un centro docente de altos estudios con grandes requerimientos, dotado de una red local y avanzados equipos de cómputo la solución puede ser el diseño de una intranet.

La Facultad de Informática de la Universidad de Cienfuegos utiliza desde hace varios años una versión de intranet a la cual se le han ido incorporando mejoras con el objetivo de brindar un servicio más eficiente.

Los avances gigantescos de la informática en los últimos años, fundamentalmente dirigido a la seguridad de las comunicaciones, las nuevas concepciones desde el punto de vista de programación y el aumento de las necesidades de los usuarios, motivaron a iniciar un proyecto de intranet que, a partir de un diseño novedoso, permitiera mantener las ventajas de la versión existente y diera respuesta a las limitaciones que se estaban presentando en cuanto a la obtención e intercambio de información y a la utilización de los servicio que ésta debe brindar.

Teniendo en cuenta que la actividad fundamental de una facultad consiste en graduar estudiantes de diferentes modalidades de curso (pregrado y postgrado), se considera que los servicios fundamentales de la intranet que se propone están enmarcados en los aspectos docente y administrativo, siendo el primero la parte del universo al cual se le dedicará los esfuerzos de este trabajo.

De esta forma la facultad podría:

- Elevar su productividad y eficiencia en cuanto a la obtención e intercambio de información referente al proceso docente-educativo.

Para lograr una solución acertada y consistente siempre resulta aconsejable llevar a cabo una investigación en las áreas del conocimiento relacionadas con el tema. En el caso que compete se tiene como objeto de estudio los temas relacionados con:

- Organización y funcionamiento de las facultades
- Origen, creación, procesamiento y destino de los datos del proceso docente-educativo.
- Nivel de acceso a estos datos.

A partir de esta situación problemática se define el **objetivo general** del presente trabajo:

Analizar el flujo de información referente al proceso docente-educativo de una facultad, con lo cual se debe diseñar e implementar el módulo docente de una intranet, que incluya servicios de información que permitan a los usuarios de una red universitaria producir y consumir dicha información.

De este objetivo general se desprenden los siguientes **objetivos específicos**:

- Realizar el estudio del flujo de información del proceso docente-educativo, tomando como muestra la Facultad de Informática de la Universidad de Cienfuegos.
- Diseñar un modelo de datos que abarque el proceso docente-educativo.
- Diseñar y establecer niveles de acceso a la información.
- Informatizar y automatizar el proceso docente-educativo referido a los estudiantes y profesores
- Diseñar un sistema multiplataforma en cuanto a Sistema Operativo y Gestor de Base de datos se refiere.

Para cumplir estos objetivos se realizarán las siguientes tareas:

- Entrevistar a los estudiantes, trabajadores docentes, no docentes y directivos de las facultades de la UCf.
- Investigar y recopilar recomendaciones en los tema de accesibilidad, autenticación y personalización de la información del proceso docente-educativo.

- Investigar y comprobar el funcionamiento del mecanismo de autenticación integrado de la red en la UCF con los servidores de correo.
- Analizar el diseño de la base de datos existente.
- Analizar, diseñar y crear la interfaz gráfica de la aplicación.
- Implementar y poner a punto la aplicación.

El presente documento está estructurado en cuatro capítulos.

Capítulo 1: En este capítulo se abordan la fundamentación teórica del tema y los conceptos asociados al dominio del problema. Se explica en detalles el problema a resolver y se describe el campo de acción donde se desarrolla. Así como las tecnologías y metodologías utilizadas para su desarrollo teniendo en cuenta las tendencias actuales en el desarrollo de intranets.

Por la importancia que tiene la etapa de análisis se descompone en dos capítulos:

Capítulo 2: Este capítulo se centra fundamentalmente en analizar el modelo de negocio, así como la descripción de dicho proceso utilizando los artefactos de UML.

Capítulo 3: En este capítulo se muestran los diagramas y modelos de casos de uso utilizados en el sistema a construir, con su correspondiente descripción, así como los requisitos funcionales y no funcionales.

Capítulo 4: Se describe el diseño de la solución propuesta con sus correspondientes modelos y/o diagramas: modelo de clases, modelo de clases persistentes, modelo de implementación.

Capítulo I: Fundamentación Teórica.

En este capítulo se abordaran aspectos teóricos del tema que se va a analizar, exponiendo los principales conceptos asociados al dominio del mismo. Se describe el contexto donde se enmarca, las características y dificultades que lo acompañan, así como las metodologías y tecnologías utilizadas.

I.1. Principales conceptos asociados al dominio del problema.

Desde 1995 el modelo World Wide Web (WWW) ha experimentado una significativa evolución al considerarlo como una herramienta que puede mejorar notablemente la productividad, surgiendo así el concepto Intranet, la misma se deriva de Internet como un paso natural en su propia evolución. Utiliza los mismos protocolos y aplicaciones, y en particular el modelo cliente/servidor.

La arquitectura tradicional cliente / servidor implica dos niveles o capas: una capa cliente y una servidor. En la arquitectura de tres capas el software de aplicación esta distribuido en tres: Interfaz de Usuario (frontend), un servidor de capa intermedia llamado Servidor de Aplicaciones (business rules) y un Servidor de Bases de Datos (backend).

Entiéndase por cliente el proceso que solicita información. Al mismo tiempo el concepto de servidor se refiere al proceso que da una respuesta positiva o negativa a la solicitud de información del cliente.

Basado en los conceptos anteriores en el modelo de tres capas, la capa de Lógica del Negocio se comporta como proceso servidor de la capa de Interfaz de Usuario y como proceso cliente ante la capa de Acceso a Datos.

I.2. Intranet

Intranet es la aplicación de los estándares Internet dentro de un ámbito corporativo para mejorar la productividad, reducir costos y mantener los sistemas de información existentes. Es una forma de poner al alcance de los trabajadores todo el potencial

informativo de una entidad, para resolver problemas, mejorar los procesos, construir nuevos recursos o mejorar los ya existentes, divulgar información de manera rápida y convertir a estos trabajadores en miembros activos de una red corporativa, o sea da al usuario la información que éste necesita para su trabajo. La Intranet pretende que cada cual tenga la información necesaria en el momento oportuno sin que tenga que recurrir a terceros para conseguirla. Una Intranet es una copia de Internet dentro de una entidad. [1]

I.2.1. Razones para implantar una Intranet

Una Intranet permite a cualquier organización compartir eficazmente una gran cantidad de recursos de información que fluye de forma rápida y transparente desde las fuentes de origen hasta los puestos de trabajo de los empleados, esto con el mínimo de costo, tiempo y esfuerzo.

Teniendo en cuenta que los cambios organizativos son cada vez más frecuentes en el mundo actual, cada vez más dinámico y competitivo es preciso que la comunicación y el flujo de información sea lo más exacto y puntual posible. Surge entonces el concepto de Intranet como una respuesta a estos problemas, siendo una solución tecnológica que: permite la entrega de información bajo demanda y en el momento en que se requiere; garantiza la precisión y actualización de la información; asegura que la información se almacena en una única fuente lógica; y permite que la información sea mantenida por los propios departamentos o personas que la generaron.

Con el uso de una intranet se reducen de manera considerable los costos de distribución de la información, quedando obsoletos los mecanismos tradicionales, entiéndase correo ordinario, documentos, actas, normativas, proyectos, agendas comunes, etc., que pueden encontrarse centralizados en páginas web internas y ser compartidas por todos los interesados, que las consultarán en el momento más oportuno, siempre y cuando tengan acceso a la Intranet.

I.2.2. Características de una Intranet

Las aplicaciones desarrolladas bajo el concepto de intranet poseen características propias, que la diferencian de otros productos comerciales, algunas de estas se listan a continuación [1].

- Rápida implantación.
- Escalable (se puede diseñar en función de las necesidades).
- Fácil navegación.
- Accesible a través de la mayoría de las plataformas informáticas del mercado.
- Puede integrar entornos distribuidos.
- Se puede añadir a fuentes de información propietarias (bases de datos, documentos realizados con procesador de texto, bases de datos de grupomática (groupware)).
- Es extensible para aplicaciones con sonido, vídeo, interactivas, etc.
- Comunicación Interactiva en línea.
- Correo Electrónico Inteligente
- Consultas a Bases de Datos
- Políticas de Administración de Documentos
- Motores de Búsqueda
- Integración a las redes locales
- Rápida visualización en línea de cualquier documento.
- Actualización, seguridad y privacidad
- Servicios Internet, FTP, WWW, Mail, Telnet, etc.
- Transacciones de información rápidas y seguras
- Integración con aplicaciones comerciales

I.2.3. Seguridad en una intranet

En el desarrollo de este documento, la seguridad se refiere a las políticas, acciones, y sistemas necesarios para proteger la integridad de la información de los sistemas de comunicación. Varios niveles de seguridad son necesarios para cumplir estos

objetivos. Generalmente, los requisitos de seguridad de la información están agrupados en las siguientes categorías:

Integridad de la información: La garantía de que los datos no han sido alterados ni interceptados.

Confidencialidad: La garantía de que sólo las personas a las que van dirigidas los datos acceden a éstos.

Autenticación: La garantía de que el usuario o el grupo de trabajo que pide acceder a otro usuario, o grupo de trabajo, recurso o servicio es realmente ese usuario o grupo de trabajo. Además, la garantía de que la información descrita y asociada con el autor, o administrador, de un objeto digital no sea desconocida.

Verificación: Comprobar que los mecanismos de seguridad son sólidos, potentes y que están correctamente implementados.

Disponibilidad: Garantizar que los recursos estén disponibles cuando se necesiten.

Valor estratégico y táctico: Internet e Intranet presentan una serie de oportunidades muy interesantes para alcanzar una rica variedad de propósitos de comunicación de carácter estratégico y táctico. Si 1995 ha sido considerado el “año de Internet”, desde 1996 se configura la era Intranet. Miles de empresas de todo el mundo se han dado cuenta de que las redes internas pueden ser de gran utilidad., y esto supone mayor ventaja competitiva para la compañía, elevar la moral del trabajador y obtener más información de los clientes y proveedores

Seguridad de la información: La confidencialidad e integridad de la información ha de basarse en un estricto control de los accesos. Esto es posible mediante privilegios del sistema y de objetos. Los privilegios pueden ser encapsulados en roles.

Diversos controles de acceso contribuyen a establecer el privilegio mínimo, es decir, que el usuario tenga únicamente aquel privilegio que necesita para hacer su trabajo. La integridad de los datos se garantiza a través de mecanismos de consistencia de datos: para ejecutar una transacción hay que confirmar una serie de datos.

Seguridad en los servidores: Las empresas operan en Internet y almacenan datos de sus clientes en una base de datos que reside detrás de un cortafuego (*firewall*)

dentro de una Intranet. El reto consiste en poder acceder a la información situada en la Intranet, protegida por el cortafuego, salvaguardando al mismo tiempo la confidencialidad e integridad de los datos.

Si la seguridad en la base de datos es un requisito previo y necesario para la seguridad en Internet, asegurar los datos frente a miradas acechantes mientras viajan a través de la red es también muy importante. Es posible cifrar mediante técnicas criptográficas la comunicación entre navegadores y servidores Web en Internet, o en una Intranet, utilizando el nivel SSL 3.0 (Secure Sockets Layer 3.0) y el sistema de encriptación SET (Secure Electronic Transaction).

En un sistema, la seguridad es un punto débil, es el corazón y la principal causa de fallos y pérdidas de la información. Para garantizar la seguridad en una Intranet se deben observar dos aspectos fundamentales: dónde y cómo están almacenados los datos de nuestra organización; para así garantizar hacia donde se van a mover y en que momento.

Las conexiones son otro punto esencial en la seguridad. Los datos viajan por usuarios particulares, en computadoras particulares, en determinados momentos y formas variables, por lo que una combinación usuario/ computadora/ momento/ aplicación puede ser un gran hueco suficientemente largo para comprometer la seguridad del sistema. Como solución a esto se pueden establecer entonces diferentes horarios para las salvallas y restauras de la información, o para las actualizaciones automáticas, que garanticen la no convergencia del personal con estos procesos.

Formas de autenticación.

Un sistema sin autenticación no es seguro. Como se había dicho anteriormente la autenticación no es más que el proceso de suministrarle al sistema la identidad del usuario, por lo que existen diversas formas de hacerlo. A continuación se enuncian algunas de ellas:

Chequeo de nombre fijado (Post Name Check)

Esta es la forma más simple de autenticación. El sistema verifica desde donde está accediendo el usuario y verifica esta información contra una lista de ordenadores de confianza.

Autenticación de nombre de usuario

Una forma un poco más segura es la verificación del nombre de usuario. Esta consiste en teclear el nombre del usuario y si está en la lista de usuarios válidos, entonces tendrá acceso al sistema.

Kerberos

Implementado por primera vez en Unix, el nombre proviene de la mitología, como una traducción del nombre con el que era denominado el perro de tres cabezas que cuidaba la puerta del reino de Hades. El usuario le brinda al sistema su cuenta y contraseña, los cuales son validados en la estación de trabajo. Luego dicha estación solicita al servidor Kerberos un identificador que es válido sólo para situación y tiempo determinados. Este identificador es conocido como “identificador de sesión de trabajo”. Es muy ventajoso ya que nunca se envía el nombre y contraseña por la red, sino que envía el identificador.

Smart cards:

Smart cards, Smart keys o lo que es considerado como *sistemas de desafío y respuesta* son un método similar al Kerberos pero las autenticaciones se realizan en la estación de trabajo.

La diferencia consiste en que este sistema no necesita de un servidor para crear el identificador que viaja por la red.

Contraseñas:

Las contraseñas son un modelo muy utilizado de autenticación, generalmente vienen acompañados del nombre del usuario y se chequean en el servidor.

Este método tiene que ser especialmente utilizado ya que existen algunas normas específicas para la generación de contraseñas, que evitan el descubrimiento casual o dificultan el intencional.

Concluyendo este punto se puede decir que existen muchas y variadas formas de protección, cada una de ellas posee sus ventajas y desventajas, es entonces una responsabilidad del usuario escoger cual usará.

I.2.3.1. Determinando los niveles de seguridad.

Existen tres niveles básicos para garantizar la seguridad de una Intranet [1]

El *primero* es denominado “vale todo”. Este es aplicado generalmente si su servidor WWW está en el exterior de su barrera de fuego y no es de su interés el hecho que el servidor “viva” o “muera”.

El *segundo* es el llamado Acercamiento Draconiano. Todo acceso al sitio es eliminado. Es necesario entonces ir habilitando bits o elementos, definidos por el programador de la Intranet, hasta lograr el acceso deseado.

El *tercero* y final es conocido como “alguien en el medio”. Es el más utilizado de todos y también el más difícil de mantener implementado si no se ha seguido una política correcta de planificación de los accesos. Consiste en dar acceso sólo a los lugares permitidos dentro del sitio, pero no a todo.

Permisos de ficheros.

Un último elemento que se debe tener en cuenta para garantizar la adecuada seguridad de un sitio web en una Intranet, es fijar los permisos a las páginas utilizando la seguridad integrada que brinda el sistema operativo al servidor WWW utilizado.

I.2.3.2. Administración de la Intranet: Una función crítica.

Para el adecuado mantenimiento de una Intranet son necesarias algunas funciones administrativas, que no dejan de ser más o menos importantes dado que todas

forman un bloque íntegro que garantiza el nivel de seguridad adecuado para la Intranet.

Contraseñas y administración de acceso.

El administrador es la persona encargada de adicionar nuevos usuarios a la Intranet y asignar los permisos según el nivel de acceso que se les haya concedido. Este le dará a cada usuario un identificador (cuenta), que puede ser de dominio público y una contraseña, que generalmente es privada.

La *administración de acceso* no es más que el hecho propio de la asignación de diferentes niveles de acceso a los usuarios. El administrador será entonces un usuario que tendrá el derecho de entrar a cualquier nivel de la Intranet, en cualquier momento. Esto es necesario para poder resolver posibles conflictos que aparezcan.

Contenido y administración de archivos.

Una de las funciones principales del administrador en la Intranet es justamente la tarea del mantenimiento de la información y los archivos, de forma tal que sea más fácil el uso de la Intranet por parte del usuario.

El mantenimiento puede significar tomar la información menos consultada y archivarla, actualizar los ficheros para que la información se corresponda en el tiempo, realizar cambios a los elementos del sitio para un mejor rendimiento o simplemente para que luzcan más atractivos.

Independientemente de que en la organización se decida trabajar con un administrador o varios, todos ellos deben entender su papel dentro de la Intranet y respetar el potencial que se les ha asignado. Por definición, ellos poseen acceso regular a un amplio rango de la información de la compañía, que generalmente, no está disponible para ser publicada. Los usuarios deben sentirse seguros del dominio de la Intranet.

I.3. Objeto de estudio

Descripción general

La Facultad de Informática (FINF) tiene aproximadamente 346 usuarios entre estudiantes y trabajadores docentes y no docentes, con una red que interconecta alrededor de 90 computadoras entre laboratorios de estudiantes y locales de trabajadores. El funcionamiento de la Facultad de Informática se detalla en el capítulo 2, en el tópico “Descripción de los procesos del negocio”.

I.4. Situación problemática

Por la importancia que representa para cualquier centro contar con una adecuada Intranet, la facultad ha tenido varios intentos de construcción de una propia, pero ninguna ha cumplido con las expectativas para una buena herramienta de este tipo.

La actual versión de la Intranet de la facultad de Informática no tiene implementadas varias de las funcionalidades que pudiesen garantizar un mejor uso de la misma, por ejemplo: no cuenta con intercambio de información docente-educativa, un servicio completo de noticias, efemérides, búsqueda de usuarios, carece de la automatización de procesos esenciales para la facultad como el trabajo con la matrícula de los estudiantes. Necesitándose realmente una herramienta que garantice un flujo de información segura que cumpla con las características propias de una INTRANET.

I.5. Problema

En las facultades se encuentra como uno de los problemas fundamentales que el manejo de la información referente al proceso docente-educativo es llevado de forma manual con los inconvenientes asociados a la manipulación de los datos de esta forma.

I.6. Idea a defender

Analizando la información anterior se ha llegado a la conclusión que el uso de diferentes tecnologías para la construcción de la intranet de la facultad no ha dado la solución del problema, pues no se ha enfocado desde la óptica adecuada. Por lo tanto, se defiende la siguiente idea:

La construcción de una solución basada en el paradigma de los servicios web, con procesos clientes/servidores distribuidos permitirá construir un modelo de intranet para la facultad que pueda irse integrando paso a paso y permita abarcar, en un momento dado, todo el universo del manejo de la información de dicha entidad.

I.7. Metodologías utilizadas.

I.7.1. Lenguaje de Modelación Unificado (UML)

El Lenguaje de Modelado Unificado (UML - Unified Modeling Language) permite modelar, construir y documentar los elementos que forman un producto de software que responde a un enfoque orientado a objetos. Este lenguaje fue creado por un grupo de estudiosos de la Ingeniería de Software formado por: Ivar Jacobson, Grady Booch y James Rumbaugh en el año 1995. Desde entonces, se ha convertido en el estándar internacional para definir, organizar y visualizar los elementos que configuran la arquitectura de una aplicación orientada a objetos. Con este lenguaje, se pretende unificar las experiencias acumuladas sobre técnicas de modelado e incorporar las mejores prácticas actuales en un acercamiento estándar.

UML no es un lenguaje de programación sino un lenguaje de propósito general para el modelado orientado a objetos y también puede considerarse como un lenguaje de modelado visual que permite una abstracción del sistema y sus componentes. [2]

Entre sus objetivos fundamentales se encuentran: [3]

1. Ser tan simple como sea posible, pero manteniendo la capacidad de modelar toda la gama de sistemas que se necesita construir.

2. Necesita ser lo suficientemente expresivo para manejar todos los conceptos que se originan en un sistema moderno, tales como la concurrencia y distribución, así como también los mecanismos de la ingeniería de software, como son el encapsulamiento y el uso de componentes.
3. Debe ser un lenguaje universal, como cualquier lenguaje de propósito general.
4. Imponer un estándar mundial.

I.7.2. Proceso Unificado de Desarrollo (RUP)

El Proceso Unificado de Desarrollo (RUP, por su denominación en inglés Rational Unified Process), fue creado por el mismo grupo de expertos que crearon UML, Ivar Jacobson, Grady Booch y James Rumbaugh en el año 1998. El objetivo que se perseguía con esta metodología era producir software de alta calidad, es decir, que cumpla con los requerimientos de los usuarios dentro de una planificación y presupuesto establecidos. Esta metodología concibió desde sus inicios el uso de UML como lenguaje de modelado.

Es un proceso dirigido por casos de uso, este avanza a través de una serie de flujos de trabajo, está centrado en la arquitectura y es iterativo e incremental. Además cubre el ciclo de vida de desarrollo de un proyecto y toma en cuenta las mejores prácticas a utilizar en el modelo de desarrollo de software.

A continuación se muestran estas prácticas. [4]

- Desarrollo de software en forma iterativa.
- Manejo de requerimientos.
- Utiliza arquitectura basada en componentes.
- Modelación del software visualmente
- Verifica la calidad del software.
- Controla los cambios.

Para apoyar el trabajo con esta metodología ha sido desarrollada por la compañía norteamericana Rational Corporation la herramienta CASE (Computer Aided Assisted Automated Software Engineering) Rational Rose. Esta herramienta integra todos los elementos que propone la metodología para cubrir el ciclo de vida de un proyecto.

I.8. Paradigmas de programación.

I.8.1. Programación por capas.

La programación por capas es un estilo de programación en la que el objetivo primordial es la separación de la lógica de negocios de la lógica de diseño, un ejemplo básico de esto es separar la capa de datos de la capa de presentación al usuario. La ventaja principal de este estilo, es que el desarrollo se puede llevar a cabo en varios niveles y en caso de algún cambio, sólo se modifica el nivel requerido sin tener que revisar todo el código. Además permite distribuir el trabajo de creación de una aplicación por niveles, de este modo, cada grupo de trabajo está totalmente abstraído del resto de niveles, simplemente es necesario conocer la Interfaz de Programación de Aplicaciones (Application Programming Interface - API por sus siglas en Inglés) que existe entre los niveles.

En el diseño de sistemas informáticos actual se suele usar las arquitecturas multinivel o Programación por capas, teniendo en cuenta las ventajas que esta ofrece. En dichas arquitecturas a cada nivel se le confía una misión simple, lo que permite el diseño de arquitecturas escalables es decir que pueden ampliarse con facilidad en caso de que las necesidades aumenten.

El diseño más usado en la actualidad es el de tres capas [5].

1. Capa de presentación
2. Capa de lógica de negocio
3. Capa de datos

1.- Capa de presentación: Esta capa provee la interfaz de usuario (IU). Aquí es donde la aplicación presenta información a los usuarios y acepta entradas o visualiza

respuestas. Idealmente, la IU no desarrolla ningún procesamiento de negocios o reglas de validación de negocios. Por el contrario, la IU debería relegar sobre la capa de negocios para manipular estos asuntos. Esto es importante, especialmente hoy en día, debido a que es muy común para una aplicación tener múltiples IU.

2.- Capa de negocio: es donde residen los programas que se ejecutan, recibiendo las peticiones del usuario y enviando las respuestas tras el proceso. Se denomina capa de negocio (e incluso de lógica del negocio - LN) pues es aquí donde se establecen todas las reglas que deben cumplirse. Esta capa se comunica con la capa de presentación, para recibir las solicitudes y presentar los resultados, y con la capa de datos, para solicitar al gestor de base de datos el almacenamiento o la recuperación de los mismos.

3.- Capa de datos: La capa de acceso a datos (AD) está formada por uno o más Sistemas Gestores de Bases de Datos (SGBD) que se encargan del almacenamiento y recuperación de los datos que necesita la capa de negocio.

Todas estas capas pueden residir en un mismo equipo. Lo más usual es que haya una multitud de computadoras donde resida la capa de presentación, es decir los clientes de la arquitectura cliente/servidor, y una computadora central que ejerce la función de servidor en dicha arquitectura. Las capas de negocio y de datos pueden residir en un mismo servidor y si el crecimiento de las necesidades lo aconseja se pueden separar en dos o mas servidores. Así mismo, si el tamaño o la complejidad de la base de datos aumentara, se puede separar en varios servidores de datos, los cuales recibirán las peticiones del servidor donde reside la capa de negocio.

Si por el contrario fuese la complejidad en la capa de negocio lo que obligase a la separación, esta capa de negocio podría residir en una o más computadoras que realizarían solicitudes a una única base de datos. En sistemas muy complejos se llega a tener una serie de ordenadores sobre los cuales corre la capa de datos, y otra serie de ordenadores sobre los cuales corre la base de datos. En este caso se está haciendo referencia a Sistemas de Aplicaciones Distribuidas.

En una arquitectura de tres niveles, los términos Capas o Niveles no significan lo mismo ni son similares. El término capa hace referencia a la forma como una

solución es segmentada desde el punto de vista lógico: Presentación/ Lógica de Negocio/ Datos. El término nivel, corresponde a la forma como las capas lógicas, se encuentran distribuidas de forma física.

I.9. Tecnologías a utilizar

I.9.1. Interfaz de usuario (Capa de Presentación)

Para desarrollar la interfaz de usuario de cualquier aplicación existen varias herramientas como Borland Delphi, Visual Fox Pro, Microsoft FrontPage, Macromedia Dreamweaver entre otras.[6]

Generalmente para el desarrollo de interfaz de intranets se utilizan herramientas web por las comodidades que ofrecen. Se realizó un análisis de las características fundamentales de Macromedia Dreamweaver MX 2004 para justificar el porqué de su selección para la realización del diseño de la interfaz de este proyecto.

Macromedia Dreamweaver

Macromedia Dreamweaver MX es uno de los editores de desarrollo Web más utilizado a nivel profesional para la creación de sitios Web. Su amplio abanico de herramientas permite crear desde la más simple página Web personal hasta el sitio Web más completo y complejo para una gran empresa y utilizar casi todos los recursos de la Web. Este editor de HTML profesional para el diseño, codificación y desarrollo de páginas, sitios y aplicaciones Web; permite la edición visual, o sea, crear páginas rápidamente sin escribir código, así como también la codificación manual. Dreamweaver ayuda además a construir aplicaciones Web dinámicas apoyadas en bases de datos, es completamente personalizable. Se pueden crear objetos y comandos propios, modificar los accesos directos de teclado, e incluso escribir código *script* (guión) para extender las capacidades de las páginas web creadas con nuevos comportamientos.

Dreamweaver soporta varias tecnologías del servidor para la construcción de aplicaciones Web, tales como: Macromedia ColdFusion, Microsoft ASP (Active

Server Page), Microsoft ASP.NET, Sun JavaServer Pages (JSP) y PHP (Profesional Home Page Tools).

I.9.2. Lógica del negocio (Capa de la Lógica del Negocio)

En la capa de lógica del negocio es necesario seleccionar un lenguaje de programación y un servidor de aplicaciones web. Teniendo en cuenta que existen varios lenguajes de programación web del lado del servidor (*Server Side Scripts*) como ASP, JSP, PHP entre otros, se analizaron las características fundamentales que permitieran seleccionar cual utilizar y en el caso de los servidores web se analizaron las características fundamentales de los dos servidores más difundidos en el mundo: Internet Information Server (IIS) y Apache.

I.9.2.1. Lenguajes de programación web

ASP

Active Server Pages (ASP) es una tecnología del lado servidor de Microsoft para páginas web generadas dinámicamente, que ha sido comercializada como un anexo a Internet Information Services (IIS). ASP ha pasado por cuatro iteraciones mayores, ASP 1.0 (distribuido con IIS 3.0), ASP 2.0 (distribuido con IIS 4.0), ASP 3.0 (distribuido con IIS 5.0) y ASP.NET (parte de la plataforma .NET de Microsoft). En el último ASP clásico, ASP 3.0, hay seis objetos integrados disponibles para el programador, Application, ASPError, Request, Response, Server y Session. Cada objeto corresponde a un grupo de funcionalidades frecuentemente usadas y útiles para crear páginas web dinámicas. Las páginas pueden ser generadas mezclando código de scripts del lado del servidor (incluyendo acceso a base de datos) con HTML y código del lado del servidor. [7]

JSP

JSP es un acrónimo de Java Server Pages, (Páginas de Servidor Java). Es una tecnología orientada a crear páginas web con programación en Java. Con JSP

pueden crearse aplicaciones web que se ejecuten en variados servidores web, de múltiples plataformas, ya que Java es un lenguaje multiplataforma. Las páginas JSP están compuestas de código HTML/XML mezclado con etiquetas especiales para programar scripts de servidor en sintaxis Java. [8]

PHP

PHP es un lenguaje de programación el cual se ejecuta en los servidores Web y que permite crear contenido dinámico en las páginas HTML, con un lenguaje propietario derivado del Perl.

Al principio, PHP sólo estaba compuesto por algunas macros que facilitaban el trabajo a la hora de crear una página Web. Hacia mediados de 1995 se creó el analizador sintáctico y se llamó PHP/F1 Versión 2, y sólo reconocía el texto HTML y algunas directivas de MySQL. A partir de este momento, la contribución al código fue pública. El crecimiento de PHP desde entonces ha sido exponencial, y han surgido versiones nuevas como los actuales, PHP3 y PHP4.

Dispone de múltiples herramientas que permiten acceder a bases de datos de forma sencilla, por lo que es ideal para crear aplicaciones para Internet.

Es multiplataforma, funciona tanto para Unix como para Windows de forma que el código que se haya creado para una de ellas no tiene porqué modificarse al pasar a la otra.

El lenguaje PHP es un lenguaje de programación de estilo clásico, con variables, sentencias condicionales, bucles, funciones, entre otras. La sintaxis que utiliza la toma de otros lenguajes muy extendidos como C y Perl. [9]

El lenguaje PHP, como todos los del tipo “guiones del lado del servidor”, envía al navegador código HTML, por lo que transparente al usuario final.

PHP se encuentra libre en el mercado y se puede acceder a él por medio de Internet. Cuenta además con un repositorio muy completo de clases, PEAR (Repositorio de Aplicaciones y Extensiones de PHP), en el cual se puede encontrar desde clases para manejar ecuaciones matemáticas o para generar gráficos, hasta clases para

generar hojas de cálculo en Excel, de una forma fácil y con funciones pocas veces vista con PHP. PEAR cuenta con 273 clases (hasta el momento de la revisión bibliográfica). [10]

I.9.2.2. Servicios web

Un servicio web es una entidad programable que proporciona alguna funcionalidad determinada, y es accesible a cualquier número de sistemas que usen las normas de Internet. Un servicio web puede ser usado internamente por una aplicación o ser publicado hacia Internet. Estos servicios permiten la ejecución de sus funcionalidades sin importar la plataforma, sistema operativo, o lenguaje en el cual estén implementados. [11]

Los servicios web se pueden utilizar para intercambiar datos en redes de ordenadores como Internet. La interoperabilidad se consigue mediante la adopción de estándares abiertos. Las organizaciones OASIS y W3C son los comités responsables de la arquitectura y reglamentación de los servicios web. Para mejorar la interoperabilidad entre distintas implementaciones de servicios web se ha creado el organismo WS-I, encargado de desarrollar diversos perfiles para definir de manera más exhaustiva estos estándares. [11]

Los servicios web tienen una interfaz descrita en un formato que puede ser procesado por una máquina (específicamente WSDL) y otros sistemas interactúan con el servicio web utilizando mensajes SOAP.

Los servicios web brindan grandes ventajas dentro de las aplicaciones distribuidas como son: [11]

- Aportan interoperabilidad entre aplicaciones de software independientemente de sus propiedades o de las plataformas sobre las que se instalen.
- Los servicios web fomentan los estándares y protocolos basados en texto, que hacen más fácil acceder a su contenido y entender su funcionamiento.
- Al apoyarse en HTTP, los servicios Web pueden aprovecharse de los sistemas de seguridad firewall sin necesidad de cambiar las reglas de filtrado.

- Permiten que servicios y software de diferentes compañías ubicadas en diferentes lugares geográficos puedan ser combinados fácilmente para proveer servicios integrados.

Atendiendo a las ventajas, antes mencionadas, que brindan el uso de webservice se incluye su utilización en el desarrollo de este proyecto.

Los servicios web pueden ser utilizados a través de dos protocolos fundamentalmente: RPC (*Remote Procedure Call*) y SOAP (*Simple Object Access Protocol* – Protocolo de Acceso Simple a Objetos). El protocolo RPC tiene una dependencia de la plataforma y la tecnología que se utiliza para ejecutar los servicios web, en cambio SOAP es un protocolo que permite la independencia de plataforma y tecnología.

I.9.2.3. SOAP

SOAP, siglas de *Simple Object Access Protocol*, es un protocolo estándar creado por Microsoft, IBM y otros, está actualmente bajo el auspicio de la W3C que define cómo dos objetos en diferentes procesos pueden comunicarse por medio de intercambio de datos XML (eXtensible Markup Language). SOAP es uno de los protocolos utilizados en los servicios Web. [12]

A diferencia de DCOM y CORBA, que son binarios, SOAP usa el código fuente en XML, que facilita la eliminación de errores, pero es menos efectivo. El intercambio de mensajes se realiza mediante tecnología de componentes. El término *Object* en el nombre significa que se adhiere al paradigma de la programación orientada a objetos.

SOAP es un marco extensible y descentralizado que permite trabajar sobre múltiples pilas de protocolos de redes informáticas. Los procedimientos de llamadas remotas pueden ser modelados en la forma de varios mensajes SOAP interactuando entre sí.

SOAP corre sobre cualquier protocolo de Internet, generalmente HTTP. SOAP tiene como base XML, con un diseño que cumple el patrón Cabecera-Desarrollo de diseño de software, como otros muchos diseños, verbigracia HTML. La cabecera (*header*)

es opcional y contiene metadatos sobre enrutamiento (*routing*), seguridad o transacciones. El desarrollo (*body*) contiene la información principal, que se conoce como carga útil (*payload*). La carga útil se acoge a un esquema de XML propio.

I.9.2.4. XML

El Lenguaje Extensible de Marcas, abreviado XML (*eXtensible Markup Language*), describe una clase de objetos de datos llamados documentos XML y describe parcialmente el comportamiento de los programas de computadora que los procesan. XML es un "perfil de aplicación" o una forma restringida de SGML, el Lenguaje Estándar Generalizado de Marcación [ISO 8879]. Por construcción, los documentos XML son documentos SGML conformados. [13]

Los documentos XML están compuestos por unidades de almacenamiento llamadas entidades, que contienen tanto datos analizados como no analizados. Los datos analizados están compuestos de caracteres, algunos de los cuales, de la forma datos carácter, y otros de la forma etiquetas. Las etiquetas codifican una descripción de la estructura de almacenamiento del documento y su estructura lógica

XML fue desarrollado por un Grupo de Trabajo (originalmente conocido como "*SGML Editorial Review Board*" – Comité Editorial de Revisión del SGML) formado bajo los auspicios del Consorcio World Wide Web (W3C), en 1996. Fue presidido por Jon Bosak de Sun Microsystems con la participación activa de un Grupo Especial de Interés en XML (previamente conocido como Grupo de Trabajo SGML (*Standard Generalized Markup Language* – Lenguaje Estándar Generalizado de Marcado) también organizado en el W3C.

Los objetivos de diseño de XML son [13]:

1. XML debe ser directamente utilizable sobre Internet.
2. XML debe soportar una amplia variedad de aplicaciones.
 - XML debe ser compatible con SGML.

- Debe ser fácil la escritura de programas que procesen documentos XML.
- El diseño de XML debe ser formal y conciso.
- Los documentos XML deben ser fácilmente creables.

XML es la solución a un problema de comunicación entre programas de computadoras. XML intenta ser un formato absolutamente genérico, con el que se puede describir cualquier tipo de archivo.

Características de XML **[13]**:

- Es una arquitectura más abierta y extensible. No se necesita versiones para que puedan funcionar en futuros navegadores. Los identificadores pueden crearse de manera simple y ser adaptados en el acto en internet/intranet por medio de un validador de documentos (parser).
- Mayor consistencia, homogeneidad y amplitud de los identificadores descriptivos del documento con XML (los RDF Resource Description FrameWork), en comparación a los atributos de la etiqueta <META> del HTML.
- Integración de los datos de las fuentes más dispares. Se podrá hacer el intercambio de documentos entre las aplicaciones tanto en el propio PC como en una red local o extensa.
- Datos compuestos de múltiples aplicaciones. La extensibilidad y flexibilidad de este lenguaje nos permitirá agrupar una variedad amplia de aplicaciones, desde páginas web hasta bases de datos.
- Gestión y manipulación de los datos desde el propio cliente web.
- Los motores de búsqueda devolverán respuestas más adecuadas y precisas, ya que la codificación del contenido web en XML consigue que la estructura de la información resulte más accesible.
- Se desarrollarán de manera extensible las búsquedas personalizables y subjetivas para robots y agentes inteligentes. También conllevará que los

clientes web puedan ser más autónomos para desarrollar tareas que actualmente se ejecutan en el servidor.

- Se permitirá un comportamiento más estable y actualizable de las aplicaciones web, incluyendo enlaces bidireccionales y almacenados de forma externa.
- El concepto de "hipertexto" se desarrollará ampliamente (permitirá denominación independiente de la ubicación, enlaces bidireccionales, enlaces que pueden especificarse y gestionarse desde fuera del documento, hiperenlaces múltiples, enlaces agrupados, atributos para los enlaces, etc. Creado a través del Lenguaje de enlaces extensible (XLL).
- Exportabilidad a otros formatos de publicación (papel, web, cd-rom, etc.). El documento maestro de la edición electrónica podría ser un documento XML que se integraría en el formato deseado de manera directa.

I.9.2.5. Servidores de Aplicaciones Web.

Se realizó un estudio de dos de los servidores de aplicaciones web mas difundidos para determinar cuál será usado en el proyecto.

Apache

El **servidor Apache** es un servidor HTTP de código abierto para plataformas Unix (BSD, GNU/Linux, etcétera), Windows y otras, que implementa el protocolo HTTP/1.1 (RFC 2616) y la noción de sitio virtual. Cuando comenzó su desarrollo en 1995 se basó inicialmente en código del popular NCSA HTTPd 1.3, pero más tarde fue reescrito por completo. El servidor Apache se desarrolla dentro del proyecto HTTP Server (httpd) de la Apache Software Foundation.

Apache presenta entre otras características mensajes de error altamente configurables, bases de datos de autenticación y negociado de contenido, carece de una interfaz gráfica que ayude en su configuración.

En la actualidad, Apache es el servidor HTTP más usado, siendo el servidor HTTP del 70% de los sitios web en el mundo y creciendo aún su cuota de mercado.

Internet Information Services

Internet Information Services (IIS) engloba una serie de herramientas administrativas que permite controlar sitios Web, FTP, SMTP (correo saliente) y Servicio de noticias (NNTP). Dispone también del soporte para crear páginas dinámicas (ASP), tecnología para el desarrollo de aplicaciones para Internet ampliamente extendida.

IIS es un servidor de distribución gratuita, pero tiene como condición que corre solamente sobre plataforma Windows. Al igual que Apache posee mensajes de error configurables, y a diferencia del mismo si posee una interfaz gráfica que ayuda en la configuración del sitio en su totalidad.

Después de analizar las características fundamentales de Apache e IIS se decide usar como servidor de aplicaciones web Apache teniendo en cuenta los siguientes aspectos:

- Se encuentra gratis en Internet,
- Posee una alta compatibilidad con PHP,
- Y cumple con uno de los objetivos específicos trazados al inicio de la investigación, que es lograr un sistema multiplataforma.

I.9.3. Capa de acceso a datos.

Sistemas Gestores de Base de Datos

Una Base de Datos (BD) es un conjunto de datos interrelacionados, almacenados con carácter más o menos permanente en la computadora, puede ser considerada una colección de datos variables en el tiempo. **[14]**

Un **Sistema Gestor de Base de Datos** (SGBD) es el software que permite la utilización y/o la actualización de los datos almacenados en una (o varias) base(s) de datos por uno o varios usuarios desde diferentes puntos de vista y a la vez. **[14]**

El objetivo fundamental de un SGBD consiste en suministrar al usuario las herramientas que le permitan manipular, en términos abstractos, los datos, o sea, de forma que no le sea necesario conocer el modo de almacenamiento de los datos en la computadora, ni el método de acceso empleado.

En el mercado existen un sinnúmero de gestores de base de datos, todos con sus características que lo hacen una opción a escoger, pero la elección, la mayoría de las veces, depende del cliente y no del desarrollador.

A continuación mostramos una tabla (Tabla 1) comparativa de algunos de los SGBD en cuanto a si son Multiplataforma o no, Gratis o no y en cuanto a la disponibilidad del código fuente.

	Multiplataforma	Gratis	Código Fuente
MSSQL	No	No	No
MSDE	No	Si	No
MySQL	Sí	Si	Sí
Postgres	Sí	Si	Sí
Firebird	Sí	Si	Si
Interbase	Si	No	Sí
SyBase	Si	No	No
INFORMIX	Sí	No	No
Oracle	Sí	no	No

Tabla 1. Comparación entre SGBD.

I.10. Conclusiones del Capítulo

Debido a la gran variedad de plataformas en cuanto a Sistema Operativo (SO) y en cuanto a soporte para datos (SGBD) se pretende con el presente trabajo elaborar un proyecto que permita la multiplataforma en ambos casos, en la medida de las posibilidades reales de implementación. Por lo tanto después de un exhaustivo análisis de las tecnologías antes expuestas llegamos a las siguientes conclusiones preeliminares:

1. La solución que propone este proyecto se basa en la construcción de una INTRANET centrándose fundamentalmente el desarrollo de la primera versión del módulo docente de una facultad.
2. Se propone utilizar la programación orientada a objetos por los beneficios que está brinda, por lo que se considera RUP (Proceso Unificado de Desarrollo) la metodología más apropiada para el desarrollo del proyecto y UML como el lenguaje de modelación necesario en este caso.
3. Debe estar basada en el modelo de programación de tres capas, que permite el desarrollo independiente del proyecto en cuanto a: Diseño de Interfaz, Lógica del Negocio y Acceso a Datos.
4. La solución a proponer debe usar la tecnología cliente/servidor para descargar el peso fundamental del proyecto en el servidor de la aplicación.
5. En el caso del servidor de aplicaciones, la opción del Servidor Apache es la idónea, por ser este servidor multiplataforma, gratuito y el más usado en Internet.
6. Una vez que se ha escogido Apache como servidor de aplicación, se selecciona PHP como el lenguaje para desarrollar los módulos necesarios, pues es este quien mejor funciona con el servidor escogido además de que no requiere de pago por su uso.
7. La tecnología de los servicios web permite desarrollar aplicaciones multiplataforma y construir un sistema grande a partir de pequeños módulos con funcionalidades propias.

8. Para el uso de esta tecnología de servicios web de forma fácil y que en un futuro pueda ser escalable se escoge el protocolo SOAP y al mismo tiempo el uso del lenguaje de marcado XML para el intercambio de datos entre módulos. Ambos (SOAP y XML) están estandarizados globalmente en Internet.
9. Dentro de las bibliotecas para PHP disponibles en Internet, una de la más completa es PEAR, contando con una serie de módulos que implementan las tareas más comunes en la ejecución de una aplicación dada. Es decir, que en la mayoría de los casos sólo es necesario reutilizar estos módulos y añadirles las nuevas funcionalidades de nuestra aplicación. Teniendo en cuenta el presente proyecto, PEAR ya cuenta con módulos para desarrollo de servicio web sobre SOAP, manejo XML, etc...
10. A nivel de la capa de datos, las opciones son varias y cualquiera de ellas resulta beneficiosa, por lo que se decide trabajar en función de lograr un sistema multiplataforma, aunque para el desarrollo concreto del proyecto se escoge MySQL.
11. Por último la elección de la interfaz de usuario, para la cual se puede utilizar cualquier explorador web que soporte HTML 4.0, debido a que el peso fundamental de la aplicación está del lado del servidor.

Capítulo II: Modelo del negocio.

En este capítulo se describen los procesos de negocio de manera general que se llevan a cabo en cualquier facultad de la Universidad de Cienfuegos. Para realizar esta descripción se utilizará UML como lenguaje de modelado y se seguirán los pasos dados por la metodología RUP.

II.1. Descripción del modelo de negocio

El modelado del negocio es una técnica que permite comprender los procesos de negocio de la organización y se desarrolla en dos pasos: **[15]**.

1. Confección de un modelo de casos de uso del negocio que identifique los actores y casos de uso del negocio que utilicen los actores.
2. Desarrollo de un modelo de objetos del negocio compuesto por trabajadores y entidades del negocio que juntos realizan los casos de uso del negocio..

II.1.1. Descripción del proceso de negocio

El objetivo fundamental de una universidad es formar profesionales de calidad, resultando la gestión de la información docente-educativa imprescindible para el correcto funcionamiento de cualquier Centro de Educación Superior (CES).

Este objetivo se puede dividir en dos de acuerdo al sistema de enseñanza:

1. Pregrado
2. Postgrado

El sistema de pregrado persigue el objetivo de graduar estudiantes de nivel superior como ingenieros y licenciados.

El sistema de postgrado pretende elevar el nivel científico a los graduados de la enseñanza superior, a través de los diferentes tipos de cursos de postgrados, como diplomados, maestrías, doctorados, etc.

Pregrado

Cuando a un estudiante se le otorga una carrera, (en este caso Ingeniería Informática), debe matricularse en la secretaría docente de la facultad a la cual pertenece esa carrera. La matrícula se hace de forma manual, tomando todos los datos del estudiante en un modelo diseñado para este proceso. Al comenzar cada año el estudiante debe ir a la secretaría a realizar la re-matrícula que es el proceso mediante el cual se actualizan los datos del estudiante.

El estudiante tiene que cursar asignaturas cuyas evaluaciones son otorgadas por los profesores. La secretaria con estas notas conforma el registro de calificaciones que es un documento donde se resumen las evaluaciones de los estudiantes en las asignaturas.

Los estudiantes, según el tipo de curso, se clasifican en: estudiantes del Curso Regular Diurno (CRD), Estudiantes de Curso Para Trabajadores (CPT) y Estudiantes de las Sedes Universitarias Municipales (SUM), aunque en el caso de la facultad a analizar solo hay en estos momentos CRD.

Los estudiantes durante la carrera pueden solicitar licencia docente, traslado, cambio de carrera, y/o baja, siempre exponiendo las causas, las cuales se analizan por los directivos de la facultad.

El personal de la facultad solicita diversos reportes a la secretaria como reportes de asistencia, listado de estudiante, actas de evaluaciones y otros reportes de interés para el funcionamiento de la Facultad.

Anteriormente se explicó como funciona en la actualidad la gestión de la información docente en la Facultad de Informática; identificando entonces los siguientes procesos de negocio:

- Solicitar Matrícula: Proceso mediante el cual el estudiante solicita la matrícula a una carrera.
- Realizar Re-Matrícula: Proceso que se realiza cada año con el objetivo de actualizar los datos de los estudiantes.

- Solicitar Licencia: Proceso mediante el cual el estudiante solicita una licencia docente.
- Solicitar Traslado: Proceso mediante el cual el estudiante solicita un traslado a otro centro docente, ya sea por un período de tiempo o definitivo.
- Solicitar Cambio de Carrera: Proceso mediante el cual el estudiante decide pedir traslado a otra carrera del mismo centro u otro centro.
- Solicitar Baja: Proceso mediante el cual el estudiante solicita la baja docente del centro.
- Solicitar Reportes: Proceso mediante el cual el estudiante, el profesor y/o el Consejo de Dirección solicitan reportes a la secretaría docente.
- Actualizar Evaluaciones: Proceso mediante el cual los profesores actualizan las evaluaciones de los estudiantes.

II.1.2. Reglas del negocio

Después de identificar los procesos de negocio se definen las siguientes reglas del negocio:

El estudiante que se dirige a la secretaría a realizar matrícula debe estar en el listado de pre-matrícula, proceder de otra facultad u otro Centro de Educación Superior (CES). Una vez matriculado el estudiante se le asigna un grupo.

La re-matrícula solo se realiza a los estudiantes que ya están matriculados, con el objetivo de mantener los datos del mismo actualizados.

El estudiante tiene que cursar asignaturas que pertenecen a una disciplina y dicha disciplina pertenecen a un plan de estudio. Estas asignaturas, según el plan de estudio, deben ser cursadas en un determinado semestre de un curso escolar, aunque se pueden hacer variaciones ante situaciones excepcionales que tengan lugar en la Facultad. Las asignaturas pueden ser evaluadas a través de trabajos de control, trabajos de cursos, seminarios, etc. Los profesores son los encargados de asignar las evaluaciones referentes a la asignatura impartida. El registro de

evaluaciones de los estudiantes lo actualiza la secretaria. Los estudiantes pueden solicitar a la secretaría sus notas al terminar cada año.

Los estudiantes del CRD deben cumplimentar como mínimo 5 años lectivos para obtener el título final que los acredita como graduados de enseñanza superior, mientras que los estudiantes del CPT y SUM lo hacen como mínimo en 6 años.

Para un estudiante solicitar licencia docente tiene que presentar razones que justifiquen su decisión y lo debe autorizar la dirección de la facultad. La licencia docente solo es válida por un año.

El traslado de un estudiante se realiza autorizado por la secretaría general del centro al analizar las causas presentadas, entre las cuales se encuentra cambio de dirección, cambio de carrera a estudiar en otro centro, entre otras que justifiquen la solicitud.

La solicitud de cambio de carrera se hace antes de comenzar el segundo año de la carrera que se está estudiando. Es autorizada por la dirección de la Facultad conjuntamente con la dirección de la Universidad.

Un estudiante solicita baja cuando decide abandonar la carrera por algún motivo, el cual es analizado por la dirección de la Facultad y del centro.

A la secretaría de la Facultad se le solicitan reportes referentes al funcionamiento docente-educativo, estos reportes los pueden solicitar el Consejo de Dirección, los estudiantes o los profesores en dependencia del nivel de acceso a la información que estos tengan.

II.2. Modelo de casos de uso del negocio

El modelo de Casos de Uso del Negocio (CUN) describe los procesos de una empresa en términos de casos de uso y actores del negocio en correspondencia con los procesos del negocio y los clientes, respectivamente. El modelo de casos de uso presenta un sistema desde la perspectiva de su uso y esquematiza cómo proporciona valor a sus usuarios. Este modelo permite a los modeladores comprender mejor qué valor proporciona el negocio a sus actores. [16].

Este modelo es definido a través de tres elementos: el diagrama de casos de uso del negocio, la descripción de los casos de uso del negocio y el diagrama de actividades.

II.2.1. Actores del negocio

Un actor del negocio es cualquier individuo, grupo, entidad, organización, máquina o sistema de información externos; con los que el negocio interactúa. Lo que se modela como actor es el rol que se juega cuando se interactúa con el negocio para beneficiarse de sus resultados [17]

Los actores de nuestro negocio se listan a continuación

Actor	Justificación
Estudiante	Interesado en matricular y modificar sus datos en sentido general.
Profesor	Encargado de dar los datos a procesar del estudiante, de actualizar sus notas y todo lo referente a la asignatura que imparte.
Consejo de Dirección	Personas que tienen acceso a revisar la información referente al funcionamiento de la facultad.

Tabla 2. Actores del negocio

II.2.2. Diagramas de casos de uso del negocio.

Para comprender los procesos de negocio se construye el diagrama de casos de uso del negocio en el que aparece cada proceso del negocio relacionado con su actor.

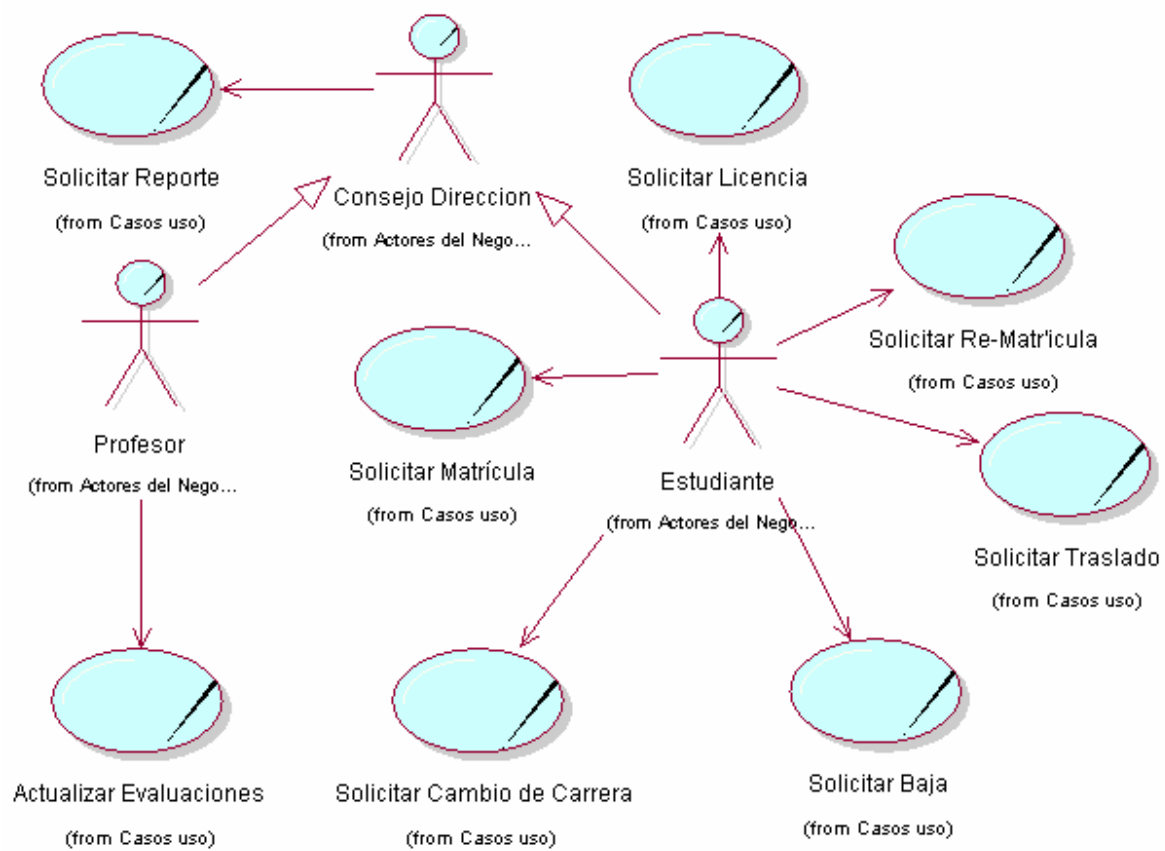


Fig. 1.- Diagrama de Casos de Uso del Negocio

II.2.3. Trabajadores del negocio

Un trabajador del negocio es una abstracción de una persona (o grupo de personas), una máquina o un sistema automatizado; que actúa en el negocio realizando una o varias actividades, interactuando con otros trabajadores del negocio y manipulando entidades del negocio. Representa un rol. [18]

Los trabajadores de nuestro negocio se listan a continuación.

Trabajador	Justificación
Secretaria	Encargada de realizar la matrícula, actualizar los datos del estudiante y emitir reportes.
Directivos	Encargado de analizar las solicitudes de Baja, Traslado, Licencia y Cambio de Carrera.

Tabla 3. Trabajadores del negocio

II.2.4. Realización de los casos de uso del negocio.

Caso de uso del negocio	Solicitar Matrícula
Actores	Estudiante (Inicia)
Propósito	Matricular un estudiante nuevo
Resumen El caso de uso se inicia cuando el estudiante obteniendo una carrera previamente, llega a la secretaría de la facultad donde se imparte la carrera a la que va a matricular, entrega los datos solicitados a la secretaria, esta determina si están completos, si no es así los rechaza, de lo contrario el estudiante queda matriculado. El caso de uso termina cuando el estudiante es matriculado.	
Acción del actor	Respuesta del proceso de negocio
1. El estudiante llega a la secretaría docente de la	

facultad en la cual va a matricular. 3. El estudiante entrega los datos solicitados. 7. El estudiante recibe la notificación de que ha sido matriculado.	2. La secretaria le pide los datos necesarios para llenar la planilla de matrícula. 4. La secretaria verifica un listado de pre-matrícula. 5. La secretaria analiza los datos entregados. 6. La secretaria llena la planilla de matrícula e informa al estudiante que ha sido matriculado.
Prioridad	Es el principal proceso del negocio.
Flujos alternos	
Línea 4	Si el estudiante no se encuentra en el listado de pre-matrícula se verifica si procede de otro centro o de otra facultad, si es así se pasa al paso 5 sino se le notifica el rechazo de su matrícula finalizando así el caso de uso.
Línea 5	Si los datos del estudiante están incompletos se le notifica el rechazo de su matrícula finalizando así el caso de uso.
Mejoras	La matrícula se realizará automatizada, disminuyendo el tiempo y los esfuerzos para este proceso, permitiendo sacar a partir de la planilla de matrícula múltiples reportes y los datos del estudiante quedarán almacenados para su posterior consulta. Como resultado del proceso saldrá una planilla similar a la que se usa manualmente pero con los datos completamente llenados con el objetivo de imprimirla y adjuntarla al expediente.

Caso de uso del negocio	Solicitar Licencia
Actores	Estudiante (inicia)
Propósito	Solicitar una licencia docente con el objetivo de separarse de la docencia por un año.
Resumen	
El caso de uso se inicia cuando el estudiante decide pedir una licencia docente, esta solicitud es analizada por la dirección de la facultad quien decide si la solicitud es aceptada o no, en caso de ser aceptada se le otorga la licencia docente al estudiante.	
Acción del actor	Respuesta del proceso de negocio
1. El estudiante entrega a la secretaria de la facultad la solicitud de una licencia docente. 8. El estudiante recibe la notificación de la licencia aceptada.	2. La secretaria informa dicha solicitud a la dirección de la facultad quienes analizan las causas de la solicitud. 3. Los directivos deciden otorgar la licencia al estudiante. 4.La dirección informa a la secretaria la concesión de la licencia docente al estudiante. 5. La secretaria llena la planilla de licencia. 6. La secretaria adjunta la planilla al expediente y retira al estudiante del grupo. 7. La secretaria informa al estudiante que la licencia ha sido otorgada.
Prioridad	Media
Flujos Alternos	

Línea 3	Si los directivos deciden no otorgar la licencia docente al estudiante, se le informa al mismo el rechazo y así culmina este caso de uso.
Mejoras	El proceso no se hará manual, actualizándose automáticamente el estado del estudiante y los cambios que este proceso cause en los datos de los estudiantes, se obtendrá como salida una planilla de licencia con los datos que requiere.

Caso de uso del negocio	Solicitar Baja
Actores	Estudiante(inicia)
Propósito	Solicitar baja del centro.
Resumen El caso de uso se inicia cuando el estudiante decide pedir baja del centro, esta solicitud la analizan los directivos, culminando el caso de uso cuando la baja es otorgada.	
Acción del actor	Respuesta del proceso de negocio
1. El estudiante hace una solicitud de baja a la secretaria 5. El estudiante recibe la notificación de la baja.	2. La secretaria entrega datos a los directivos. 3. Los directivos analizan la solicitud del estudiante. 4. Los directivos Informan al estudiante el otorgamiento de la baja.
Prioridad	Media
Mejoras	El proceso no se hará manual, disminuirá el tiempo para realizar la acción y los cambios en los datos del estudiante se efectuarán automáticamente. De este proceso se obtendrá un modelo de baja con los

	datos requeridos referentes al estudiante que solicita la baja.
--	---

Caso de uso del negocio	Solicitar Traslado
Actores	Estudiante(inicia)
Propósito	Pedir un traslado a otro centro, ya sea por un período de tiempo o definitivo.
Resumen El caso de uso se inicia cuando el estudiante decide pedir traslado, este entrega por escrito una solicitud, la secretaria la recoge, le hace entrega de la solicitud a los directivos decidiendo estos si se otorga o no el traslado. El caso de uso culmina cuando el estudiante recibe respuesta.	
Acción del actor	Respuesta del proceso de negocio
1. El estudiante hace una solicitud de traslado a la secretaria. 5. El estudiante recibe notificación de traslado.	2. La secretaria entrega datos a los directivos. 3. Los directivos analizan la solicitud de traslado. 4. Informa al estudiante notificación de traslado aceptado.
Prioridad	Media
Cursos Alternos	
Línea 5	Si los directivos no aceptan las causas del traslado informan al estudiante que el traslado es rechazado culminando así el caso de uso.
Mejoras	El proceso no se hará manual, actualizándose automáticamente el estado del estudiante y los cambios que este proceso cause en los datos de los estudiantes.

Caso de uso del negocio	Solicitar Cambio de Carrera
Actores	Estudiante(inicia)
Propósito	Realizar un cambio de carrera que puede ser dentro de la misma facultad, a facultades distintas o a otro centro.
Resumen	
El caso de uso se inicia cuando el estudiante llega a la secretaría de su facultad para solicitar cambio de carrera, la secretaria recibe por escrito la solicitud, esta es entregada a los directivos para decidir el cambio de carrera. El caso de uso termina cuando se le da respuesta al estudiante.	
Acción del actor	Respuesta del proceso de negocio
1. El estudiante entrega una solicitud de cambio de carrera a la secretaria.	2. La secretaria recibe la solicitud, para entregarla a los directivos
	3. Los directivos reciben la solicitud y analizan si cumple con los requerimientos necesarios para realizar cambio de carrera.
5. Recibe notificación de cambio de carrera aceptado.	4. Informa al estudiante cambio de carrera aceptado.
Prioridad	Media
Cursos Alternativos	
Línea 4	Si la solicitud no es aceptada se le informa al estudiante culminando así el caso de uso.
Mejoras	El proceso será automatizado, permitiendo actualizar los datos del estudiante, posibilitando

	imprimir los documentos referentes al cambio de carrera.
--	--

Caso de uso del negocio	Solicitar Re-Matrícula
Actores	Estudiante (inicia)
Propósito	Ratificar la matrícula ya existente
Resumen El caso de uso se inicia cuando el estudiante solicita ser re-matricularse, para lo cual tiene que acudir a la secretaría para actualizar sus datos. El caso de uso culmina cuando quedan actualizados los datos de los estudiantes en la matrícula.	
Acción del actor	Respuesta del proceso de negocio
1. El estudiante al iniciar el curso docente, se dirige a la secretaría a actualizar sus datos. 2. Entrega sus datos personales 6. El estudiante recibe la notificación de ratificación de matrícula aceptada.	3. La secretaria compara los datos con los de la matrícula y en caso de que hayan modificaciones actualiza la planilla. 4. La secretaria actualiza el curso docente en la planilla de re-matrícula. 5. Le informa al estudiante que sus datos han sido actualizados.
Prioridad	Alta
Mejoras	Este proceso al automatizarlo permitirá, al comenzar cada año, actualizar los datos del estudiante

	cambiando a la vez todos los documentos que reflejen alguna información que pueda cambiar.
--	--

Caso de uso del negocio	Actualizar evaluaciones
Actores	Profesor (Inicia)
Propósito	Registrar las notas de las evaluaciones de los estudiantes.
Resumen El caso de uso se inicia cuando el profesor recibe de la secretaria el modelo de acta de los estudiantes, una vez realizado el examen se plasma en el acta la evaluación del estudiante y este firma, luego se entrega el acta a la secretaria, culminando así el caso de uso.	
Acción del actor	Respuesta del proceso de negocio
1. El profesor solicita el modelo de acta a la secretaria. 3. El profesor recibe el modelo de acta. 4. El profesor llena el acta con la nota obtenida por el estudiante en el examen y la entrega a la secretaria.	2. La secretaria recibe solicitud de acta y entrega dicho modelo. 5. La secretaria recibe el acta. 6. La secretaria revisa que todos los datos del acta estén correctamente llenados.
Prioridad	Alta
Mejoras	Se podrán generar las actas para reflejar las evaluaciones teniendo en cuenta el grupo a evaluar

	y en caso de que algún estudiante no tenga que ser evaluado saldrá con esta información actualizada.
--	--

Caso de uso del negocio	Solicitar Reportes
Actores	Consejo de Dirección (inicia)
Propósito	Solicitar un reporte a la secretaría
Resumen El caso de uso se inicia cuando un estudiante, un profesor o un directivo necesita un reporte y lo solicita a la secretaria, esta analizando el acceso a la información de quien solicita el reporte lo conforma y lo entrega, terminando así el caso de uso.	
Acción del actor	Respuesta del proceso de negocio
1. La persona solicita a la secretaria un reporte. 5. Recibe el reporte solicitado.	2. La secretaria analiza la solicitud del reporte. 3. Verifica si la persona tiene acceso a la información solicitada. 4. Conformo el reporte y lo entrega a la persona que lo solicita.
Prioridad	Media
Cursos Alternos	
Línea 3	Si la persona no tiene acceso a la información solicitada la secretaria notifica a la persona rechazo de la solicitud culminando así el caso de uso.
Mejoras	Los reportes se harán de manera automática, utilizando los documentos previamente almacenados por el sistema.

II.3. Diagramas de Actividades

El diagrama de actividad es un grafo que contiene los estados en que puede hallarse la actividad a analizar. Cada estado de la actividad representa la ejecución de una sentencia de un procedimiento, o el funcionamiento de una actividad en un flujo de trabajo. En resumen describe un proceso que explora el orden de las actividades que logran los objetivos del negocio.[19]

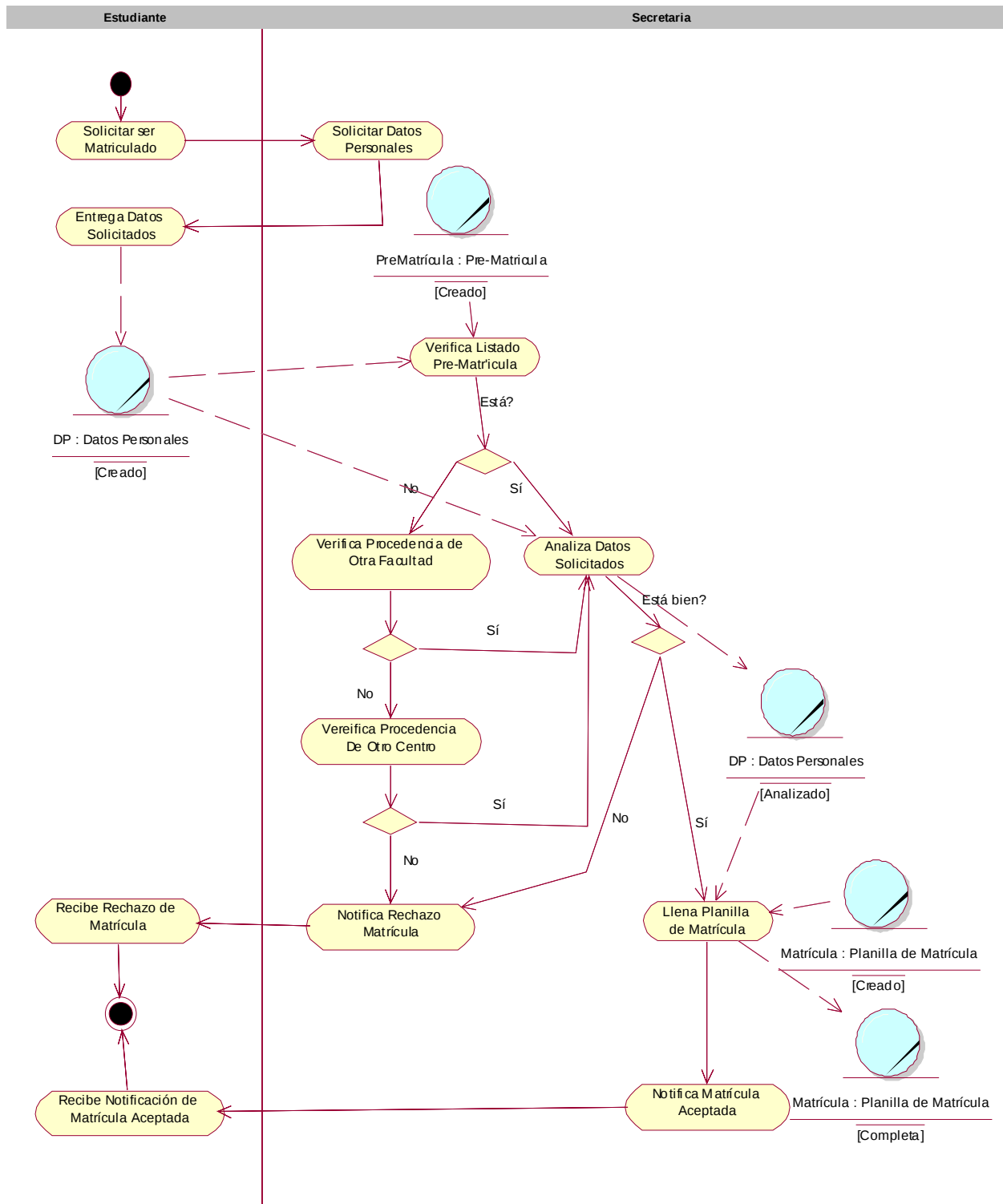


Fig. 2.- Diagrama de Actividad. Caso de Uso Solicitar Matrícula

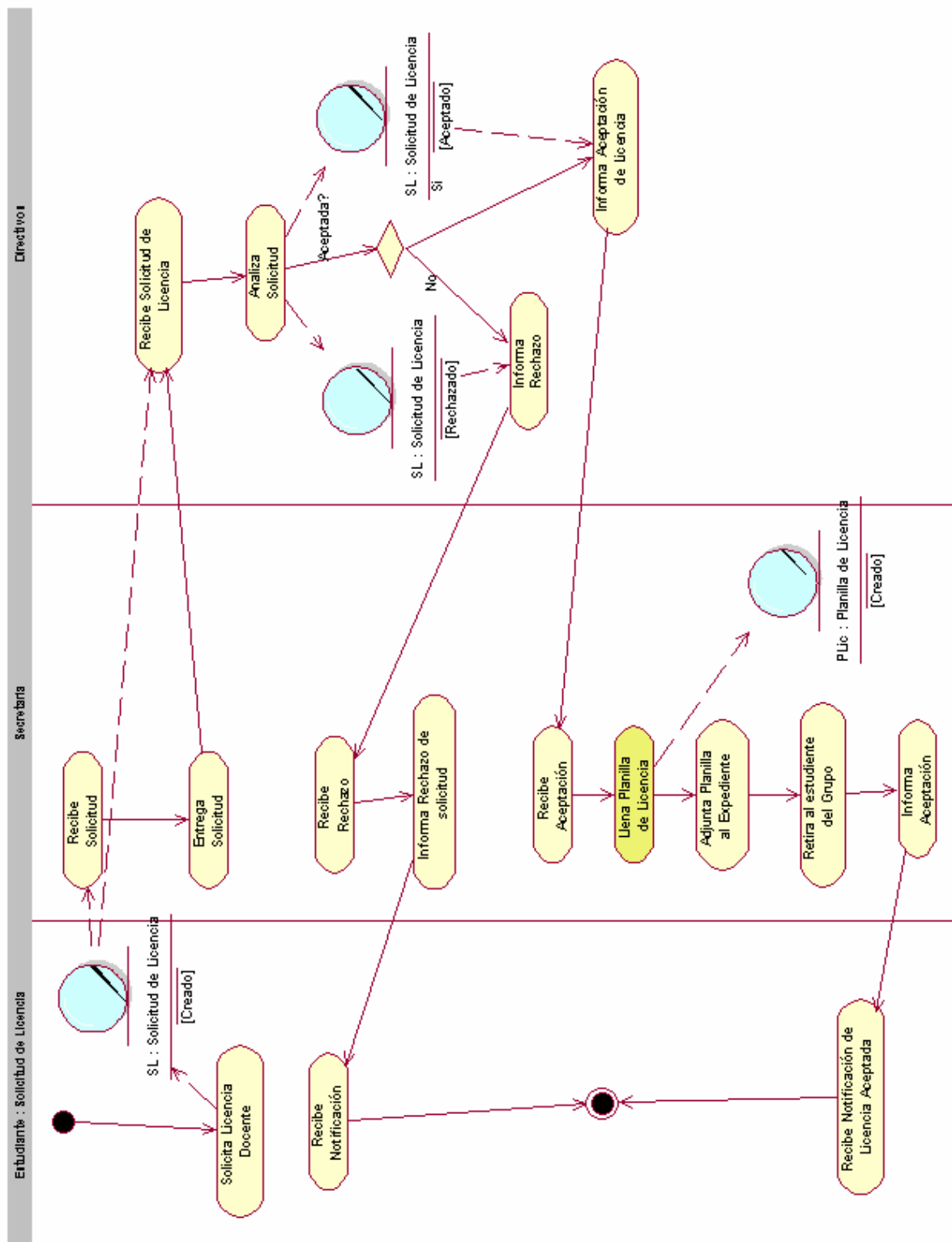


Fig. 3.- Diagrama de Actividad. Caso de Uso Solicitar Licencia

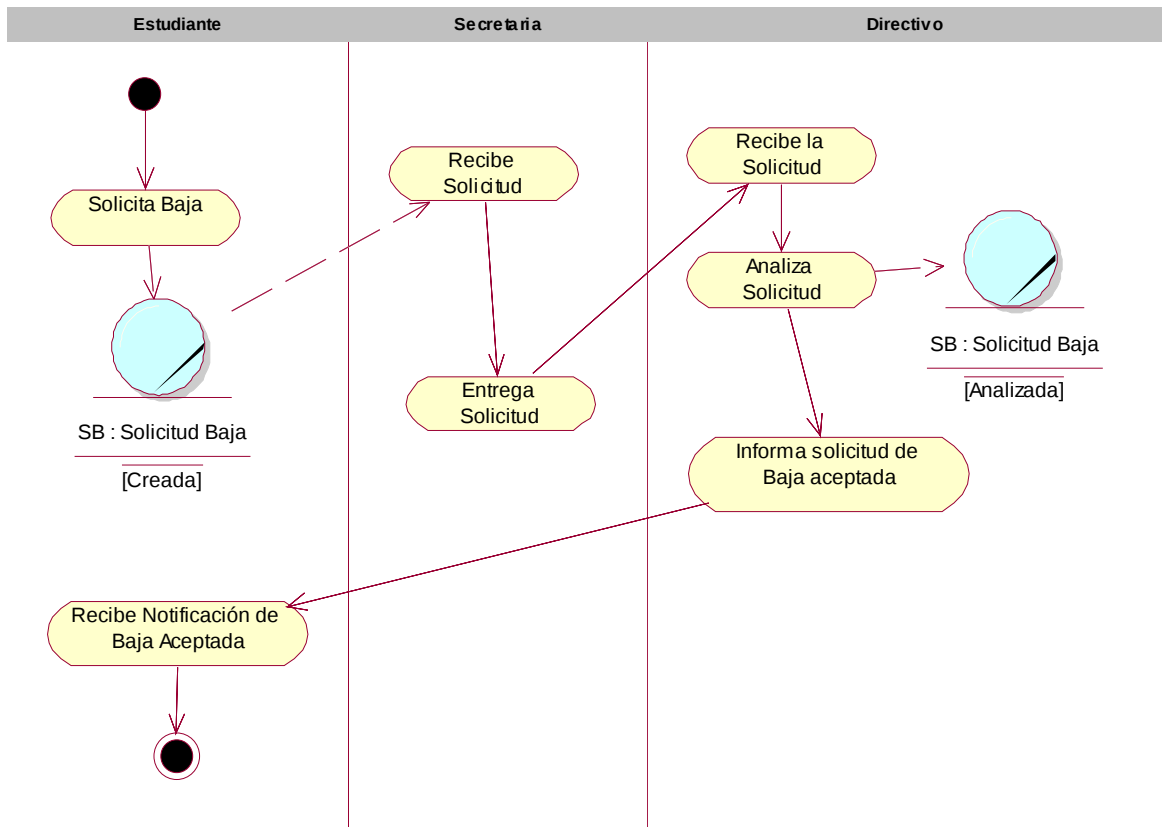


Fig. 4.- Diagrama de Actividad. Caso de Uso Solicitar Baja

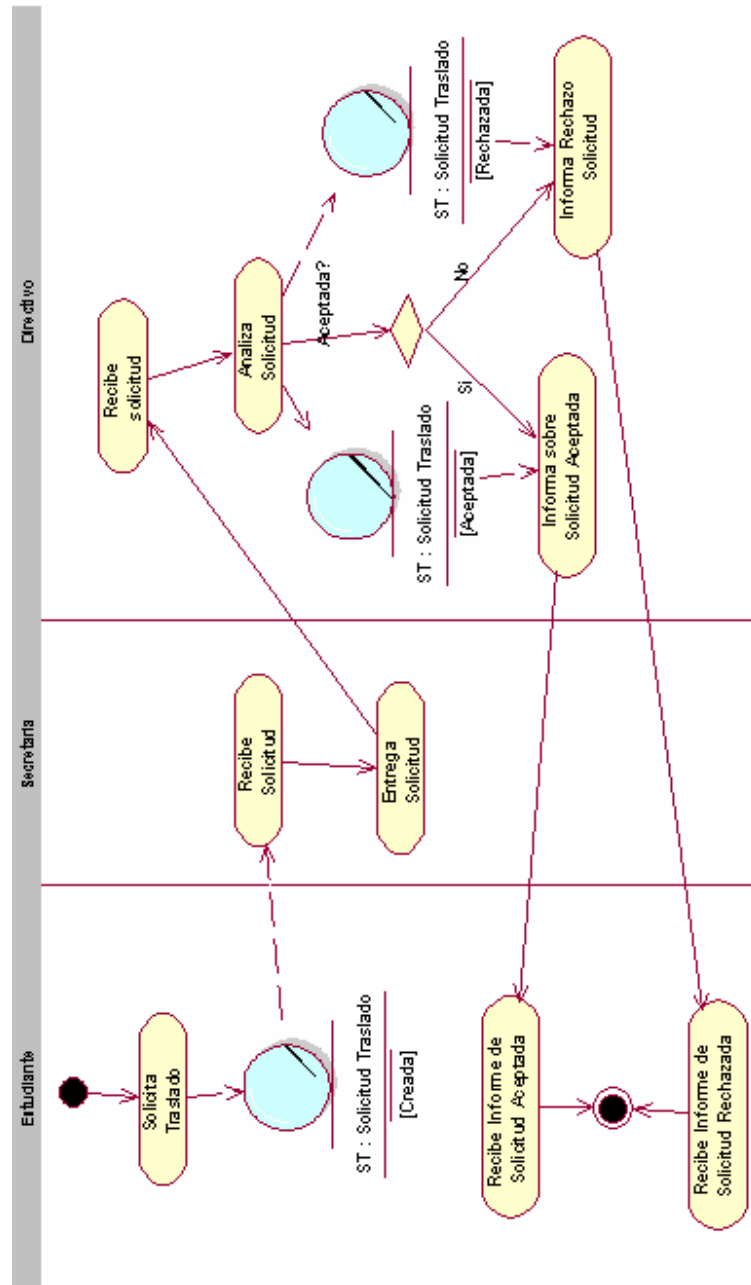


Fig. 5.- Diagrama de Actividad Caso de uso Solicitar Traslado

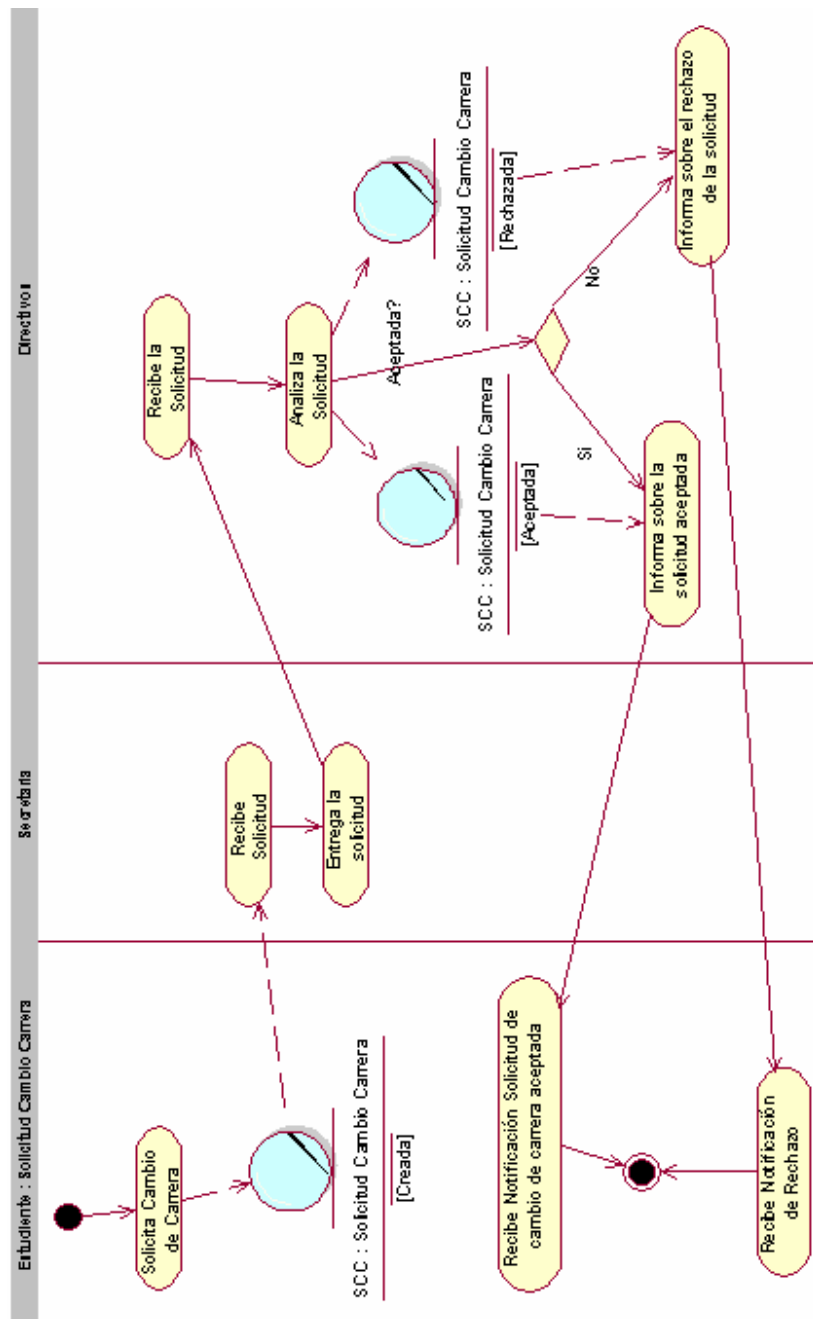


Fig. 6.- Diagrama de Actividad. Caso de Uso Solicitar Cambio de Carrera

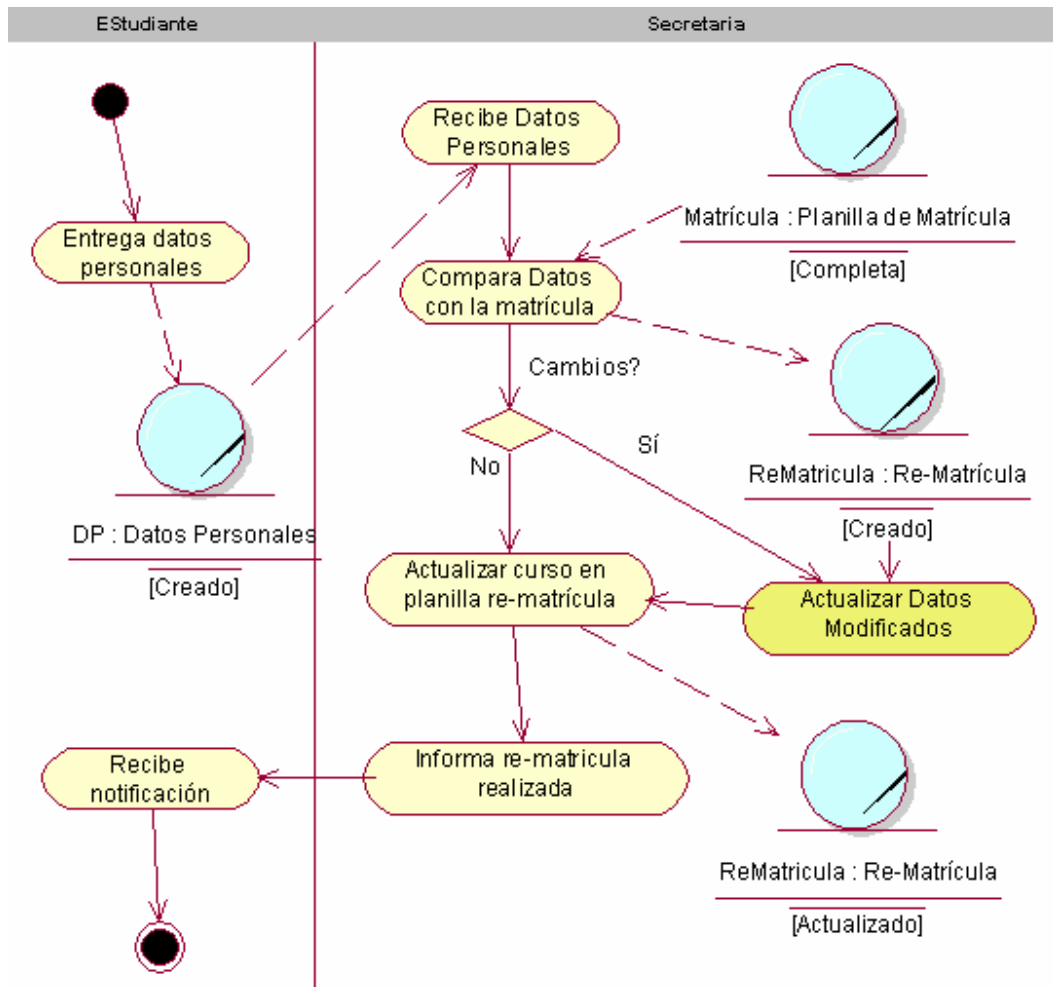


Fig. 7.- Diagrama de Actividad. Caso de Uso Solicitar Re-Matrícula

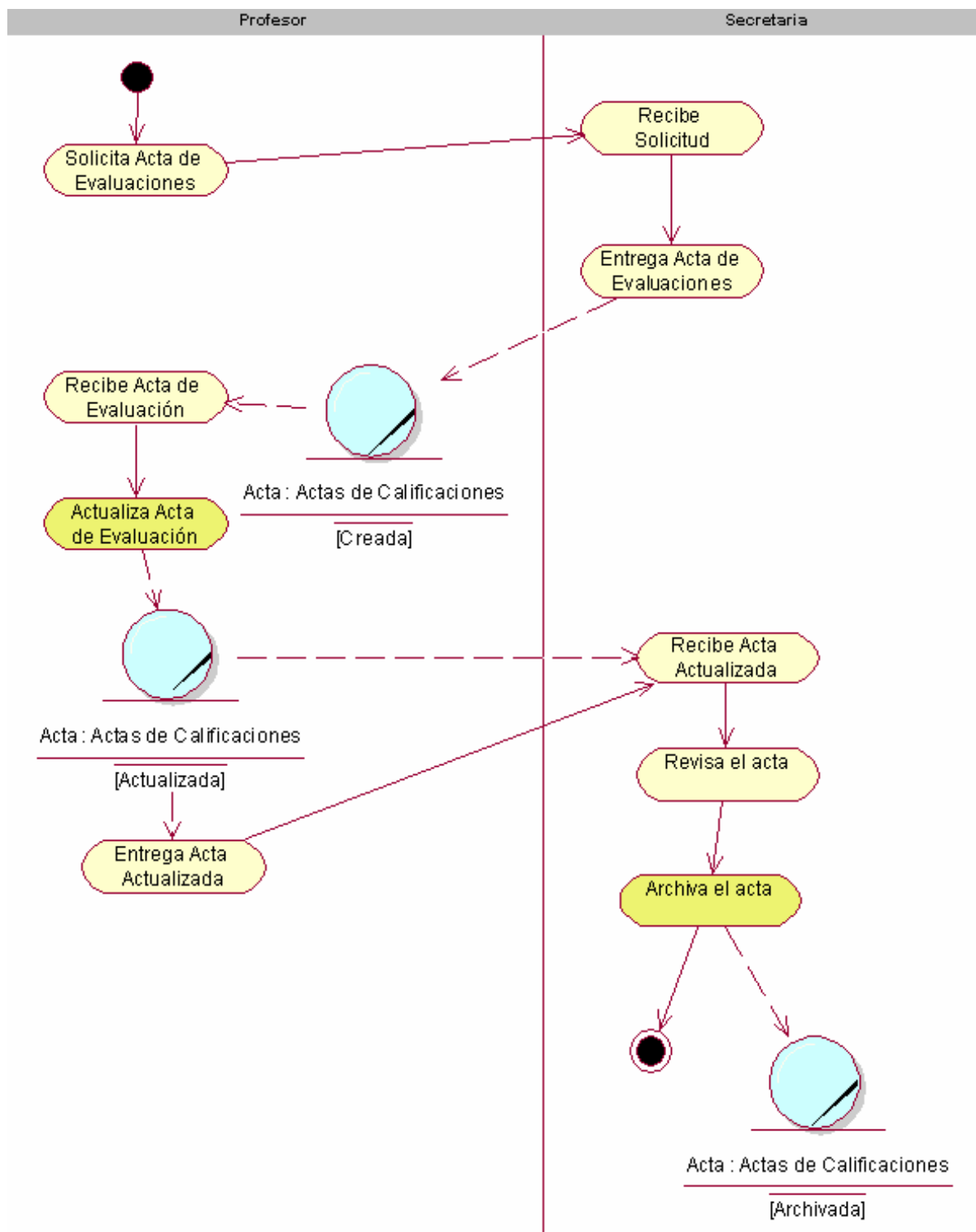


Fig. 8.- Diagrama de Actividad. Caso de Uso Actualizar Evaluaciones

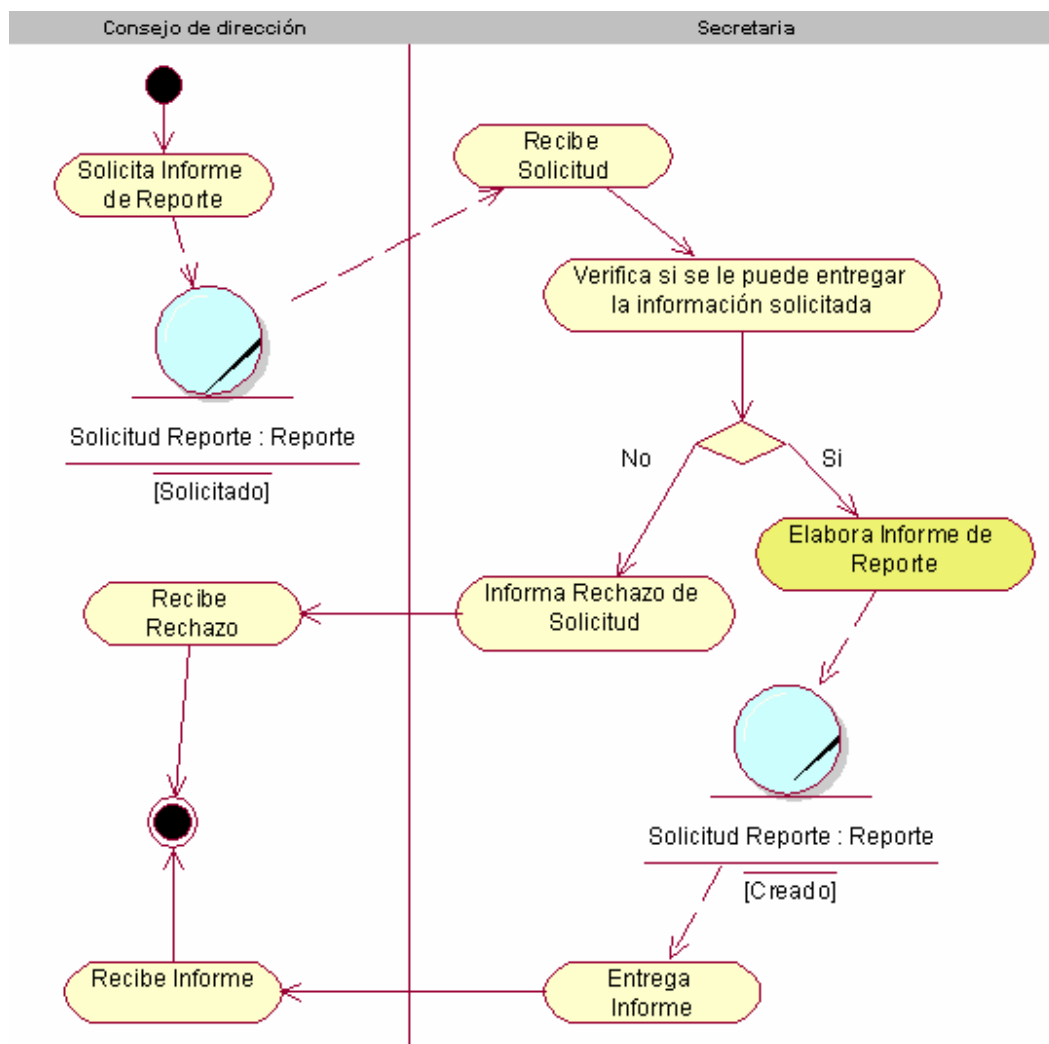


Fig. 9.- Diagrama de Actividad. Caso de Uso Solicitar Reportes

II.4. Diagrama de Clases del Modelo de Objetos

Un modelo de objetos del negocio es un modelo interno a un negocio. Describe como cada caso de uso del negocio es llevado a cabo por parte de un conjunto de trabajadores que utilizan un conjunto de entidades del negocio y unidades de trabajo [20].

Una entidad del negocio representa algo, que los trabajadores toman, inspeccionan, manipulan, producen o utilizan en un caso de uso del negocio. El diagrama de clases del modelo de objeto, es un artefacto que se construye para describir el modelo de

objetos del negocio. En la figura 10 se muestra el modelo de objetos del negocio estudiado.

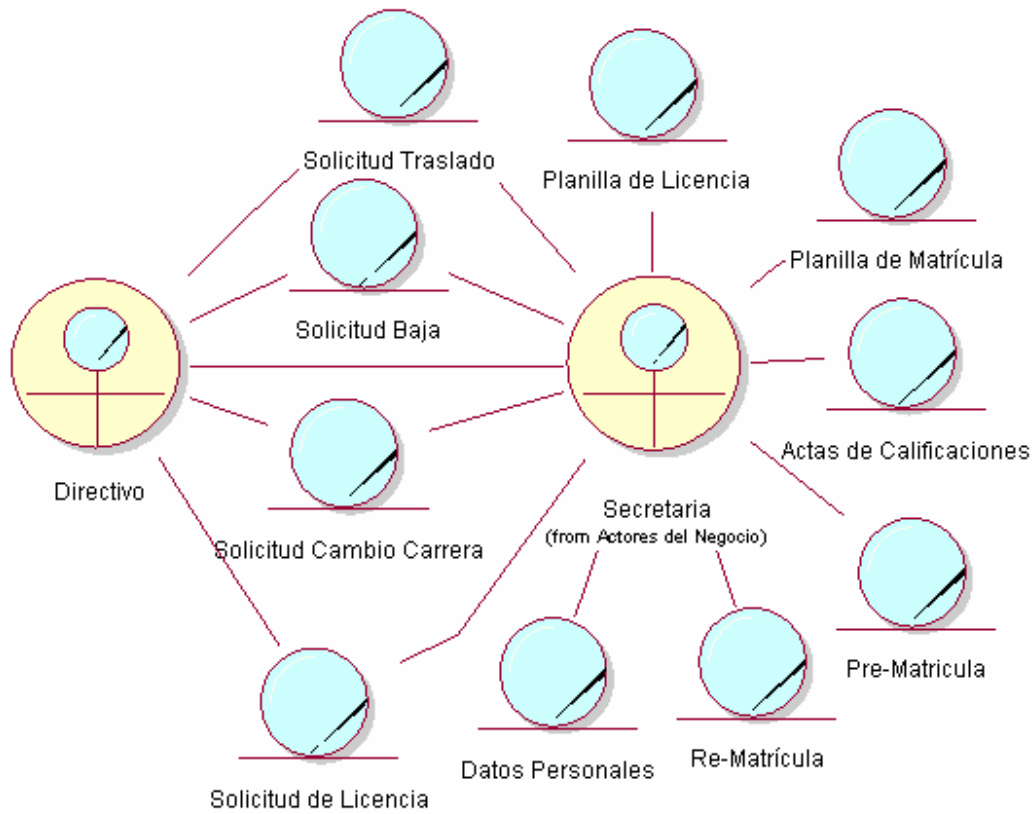


Fig. 10.- Diagrama de Clases del Modelo de Objetos

II.5. Conclusiones del Capítulo.

En este capítulo fueron descritos los procesos que tiene lugar en la Facultad de Informática de la Universidad de Cienfuegos, identificando a su vez los roles y objetos del negocio, así como su relación en esos procesos. Esta descripción fue realizada mediante el modelo del negocio, para lo cual se elaboraron los modelos de casos de uso y de actividad.

Se logró una mejor comprensión del negocio que se trata, dando paso al modelado del sistema.

Capítulo III: Modelo de Sistema.

En este capítulo se hace una descripción pormenorizada de los elementos necesarios para realizar un análisis de los requisitos funcionales del Módulo Docente de la Intranet de una facultad.

III.1. Descripción General del Modelo de Sistema

III.1.1. Requisitos Funcionales

Los requerimientos¹ funcionales permiten expresar una especificación más detallada de las responsabilidades del sistema que se propone. Ellos permiten determinar, de una manera clara, lo que debe hacer el mismo. [21].

Los requerimientos funcionales del sistema propuestos son los siguientes:

1. Insertar Noticias
2. Actualizar Noticias
3. Eliminar Noticias
4. Insertar Efeméride
5. Actualizar Efeméride.
6. Eliminar Efeméride.
7. Registrar Matrícula
8. Iniciar Curso Escolar
9. Actualizar Matrícula
10. Visualizar Datos Personales de los estudiantes
11. Registrar datos de baja.
12. Registrar datos de licencia.
13. Re-Matricular Estudiantes.
14. Conformar Listados de Grupos
15. Insertar Codificador de Sexo

¹ En el presente documento requisitos y requerimientos se refiere al mismo concepto.

16. Actualizar Codificador de Sexo
17. Insertar Codificador Raza
18. Actualizar Codificador Raza
19. Insertar Codificador País
20. Actualizar Codificador País
21. Insertar Codificador Provincia
22. Actualizar Codificador Provincia
23. Insertar Codificador Municipio
24. Actualizar Codificador Municipio
25. Insertar Codificador Nivel Educativo Padres
26. Actualizar Codificador Nivel Educativo Padres
27. Insertar Codificador Clase de Estudiante
28. Actualizar Codificador Clase de Estudiante
29. Insertar Codificador Tipo de Curso
30. Actualizar Codificador Tipo de Curso
31. Insertar Codificador Situación Escolar
32. Actualizar Codificador Situación Escolar
33. Insertar Procedencia Escolar
34. Actualizar Procedencia Escolar
35. Insertar Codificador Vía de Ingreso
36. Actualizar Codificador Vía de Ingreso
37. Insertar Nivel Educativo
38. Actualizar Nivel Educativo
39. Insertar Codificador Concepto de Ingreso
40. Actualizar Codificador Concepto de Ingreso
41. Insertar Codificador Actividad Laboral
42. Actualizar Codificador Actividad Laboral
43. Insertar Codificador Cargo Ocupacional
44. Actualizar Codificador Cargo Ocupacional
45. Actualizar Registro de Calificaciones
46. Conformar Actas de Asignaturas

47. Registrar Calificaciones
48. Modificar Calificaciones.
49. Visualizar Calificaciones
50. Visualizar Planilla de Matrícula
51. Visualizar Planilla de Baja
52. Visualizar Planilla de Licencia
53. Visualizar Actas de Asignaturas
54. Visualizar Calificaciones
55. Visualizar Registro de Calificaciones
56. Visualizar Listado Oficial de Estudiantes
57. Visualizar Listados de Estudiantes por Grupos
58. Consultar Ayuda
59. Consultar Efemérides
60. Consultar Noticias
61. Iniciar Sesión
62. Crear Usuario
63. Cambiar Contraseña Usuario
64. Modificar Usuario
65. Asignar Permisos a Usuarios
66. Eliminar Usuario
67. Habilitar Usuario
68. Inhabilitar Usuario
69. Consultar Información General

III.1.2. Requisitos no funcionales

Los requerimientos no funcionales especifican cualidades, propiedades del sistema; como restricciones del entorno o de la implementación, rendimiento, dependencias de la plataforma, etc. [22].

Los requerimientos no funcionales del sistema propuesto son los siguientes.

Requisitos de interfaz.

- La interfaz debe ser diseñada de modo tal que el usuario pueda tener en todo momento el control de la aplicación, lo que le permitirá ir de un punto a otro dentro de ella con gran facilidad. Se cuidará porque la aplicación sea lo más interactiva posible.
- La ejecución de los comandos debe ser posible por el uso del teclado u otros dispositivos como el *Mouse*.
- Los mensajes de error deben ser reportados por la propia aplicación en la medida de las posibilidades y no por el Sistema Operativo.
- Los mensajes de las aplicaciones deben estar en español.
- La entrada de datos debe ser posible por varias vías, ya sea por el teclado, mouse u otros dispositivos.

Requisitos de Usabilidad.

- El sistema será utilizado solo por personas registradas, estos pueden ser estudiantes, profesores y trabajadores, a las cuales se les asignan privilegios, es decir solo pueden trabajar con la información a la que tienen acceso.

Requisitos de Rendimientos

- La eficiencia del sistema estará determinada por el aprovechamiento de los recursos en el modelo a tres capas.

Requisitos de Soporte

- Los servicios de instalación y mantenimiento del sistema será responsabilidad del administrador de la red de la facultad.

Requisitos de Seguridad

- Debe garantizar la conectividad e integridad de los datos almacenados a través de la red. Esto está garantizado por Sistema Operativo.
- Debe garantizar la confidencialidad para proteger la información de acceso no autorizado. Esto estará garantizado por el Sistema Gestor de Base de Datos
- El sistema impondrá un estricto control de acceso que permitirá a cada usuario tener disponible solamente las opciones relacionadas con su actividad.
- El sistema no permitirá el acceso a informaciones a partir de puntos no autorizados.
- En el diseño de la aplicación debe tenerse en cuenta la existencia de regulaciones y/o restricciones en la manipulación de la información.
- Las reglas de control de acceso deben ser aplicables a las bases de datos y a los sistemas que trabajan operativamente con los datos.

Requisitos de Confiabilidad

- El sistema debe ser tolerante ante los fallos; y las operaciones a realizar deben ser transaccionales.

Requisitos de Software

- La aplicación debe poderse ejecutar en entornos *Windows* y/o *Linux* (Multiplataforma). Del lado del servidor se utilizará Apache como servidor web, del lado del cliente cualquiera de los exploradores existentes en el mercado.

Requisitos de Hardware

- Se requiere de una máquina que funcione como servidor que los requerimientos específicos estarán en dependencia del sistema Gestor de Base de Datos a utilizar. Las computadoras clientes al menos deben cumplir los requisitos mínimos para poder ejecutar los navegadores de web.

Requisitos Políticos

- La aplicación debe cumplir con lineamientos, políticos y/o regulaciones del Ministerio de Educación Superior.

III.2. Modelo de Casos de Uso del sistema

El modelo de casos de uso permite que los desarrolladores del software y los clientes lleguen a un acuerdo sobre los requisitos, es decir, sobre las condiciones y posibilidades que debe cumplir el sistema. Describe lo que hace el sistema para cada tipo de usuario. [23].

III.2.1. Actores del Sistema

Un actor no es más que un conjunto de roles que los usuarios de Casos de Uso desempeñan cuando interaccionan con estos Casos de Uso. Los actores representan a terceros fuera del sistema que colaboran con el mismo. Una vez que hemos identificado los actores del sistema, tenemos identificado el entorno externo del sistema [24].

Actor	Justificación
Visitante	Toda persona que entre al sitio web en busca de información general como noticias efemérides, etc, que de algún modo debe estar vinculada a la Universidad.

Secretaria	Este actor es el encargado de matricular al estudiante, de actualizar el registro de evaluaciones de los estudiantes, de emitir las actas docentes, reportes estadísticos, y los reportes de evaluaciones, además registra la licencia docente, las bajas y realiza la re-matrícula.
Profesor	Es la persona que actualiza las actas y entrega la información a poner en el registro de calificaciones, además solicita reportes de su interés.
Directivos	Interesados en recibir reportes sobre el funcionamiento de la facultad.
Administrador	Tiene control total sobre el sistema, es quien crea las cuentas de acceso al mismo y le asigna a cada usuario sus permisos en dependencia al rol a desarrollar.
Encargado Administrativo	Este rol lo cumple determinado usuario que tiene privilegio para modificar las informaciones que se muestran en el módulo general del módulo docente de la facultad.

Tabla 4. Actores del Sistema

III.2.2. Casos de Uso del Sistema

La forma en que interactúa cada actor del sistema con el sistema se representa con un Caso de Uso. Los Casos de Uso son “fragmentos” de funcionalidad que el sistema ofrece para aportar un resultado de valor para sus actores. De manera más precisa, un Caso de Uso especifica una secuencia de acciones que el sistema puede llevar a cabo interactuando con sus actores, incluyendo alternativas dentro de la secuencia [15].

Por el número de casos de uso se introducen paquetes al modelo de casos de uso del sistema con el objetivo de disminuir el tamaño y así aumentar en comprensión.

En la (Figura 11) se muestra el diagrama de casos de uso por paquetes (Paquete: Información General, Paquete: Módulo Administrativo y Paquete: Módulo Docente, este último paquete a su vez está dividido en 4 paquetes (Figura 12): Matrícula, Codificadores, Evaluaciones y Reportes.

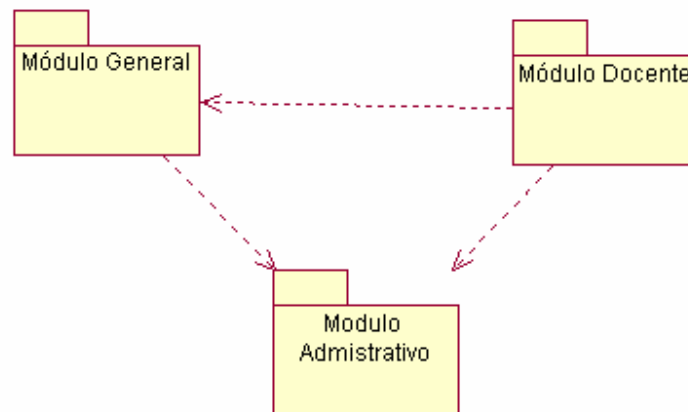


Fig. 11.- Diagrama de Casos de Usos por Paquetes

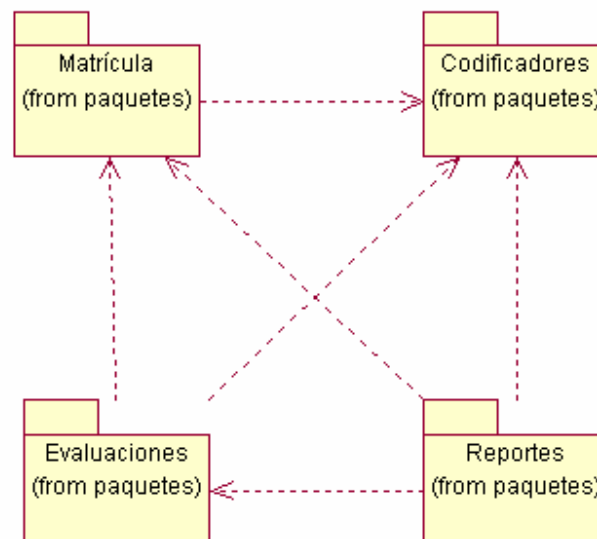


Fig. 12.- Paquete: Módulo Docente

El paquete Módulo General (Figura 13) contiene los casos de uso de información general manejada en el sistema:

1. Consultar Efemérides

2. Consultar Noticias.
3. Consultar Información General

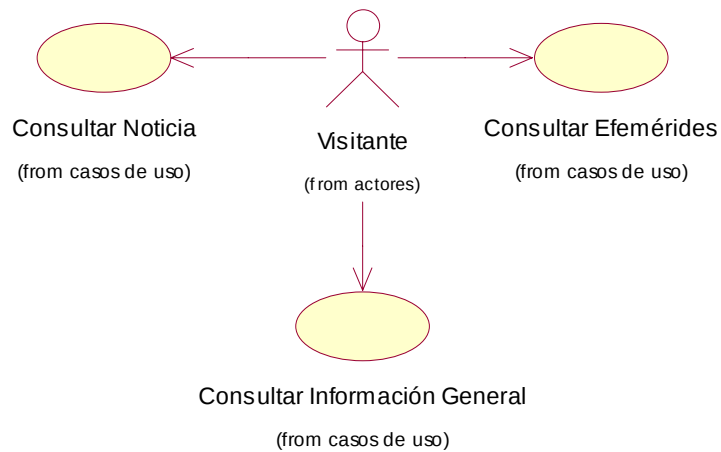


Fig. 13.- Diagrama de caso de Uso del Sistema: Paquete Módulo General

El paquete Matrícula (Figura 14) contiene todos los casos de uso relacionados con la matrícula de los estudiantes:

1. Registrar Matrícula.
2. Actualizar Matrícula.
3. Conformar Listados de Grupos.
4. Re-Matricular Estudiantes.
5. Registrar Datos Licencia.
6. Registrar Datos Baja.
7. Iniciar Curso Escolar.

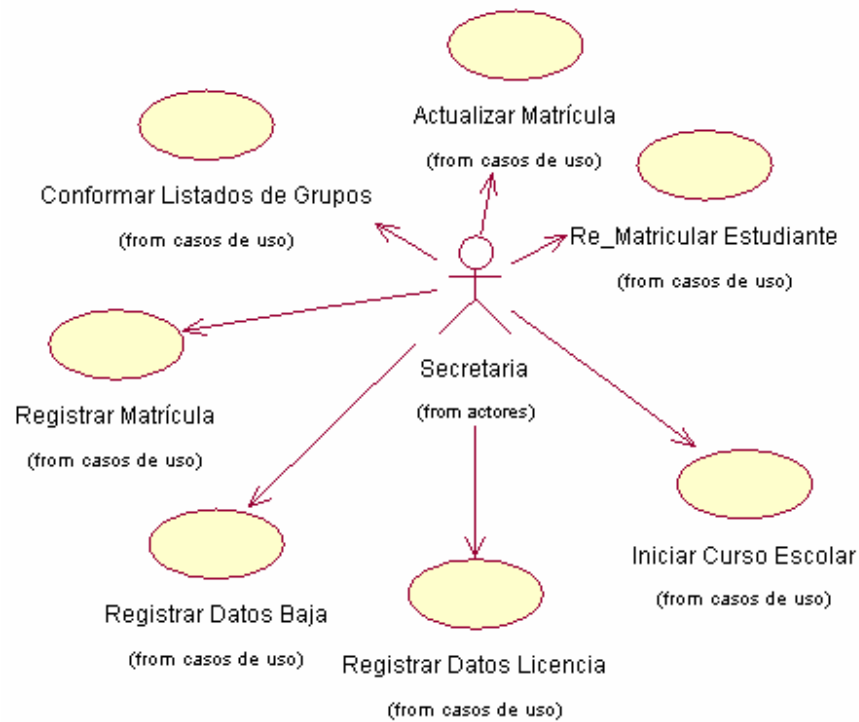


Fig. 14.- Diagrama de caso de Uso del Sistema: Paquete Matrícula

El paquete Codificadores (Figura 15) contiene los casos de uso utilizados para el manejo de los codificadores.

1. Gestionar Codificador

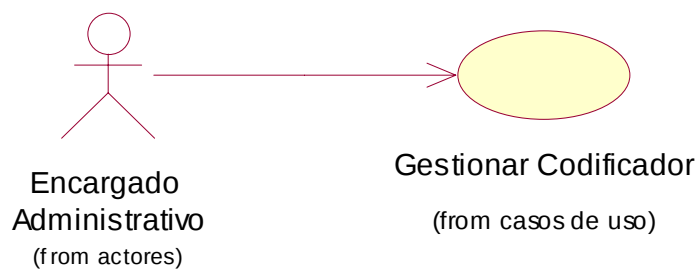


Fig. 15.- Diagrama de caso de Uso del Sistema: Paquete Codificadores

El paquete Evaluaciones (Figura 16) agrupa los casos de uso relacionados con las evaluaciones en sentido general.

1. Registrar Calificaciones
2. Modificar Calificaciones
3. Conformar Actas de Asignaturas

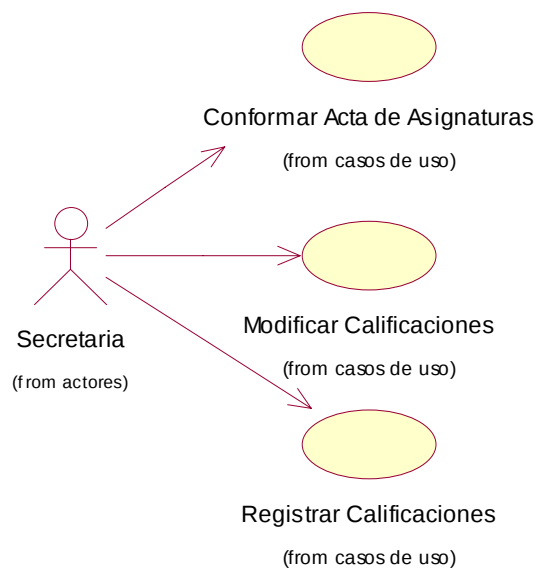


Fig. 16.- Diagrama de caso de Uso del Sistema: Paquete Evaluaciones

El paquete Reportes (Figura 17) contiene los casos de uso relacionados con los reportes.

- 1 Visualizar Planilla de Matrícula
- 2 Visualizar Datos de Baja
- 3 Visualizar Datos de Licencia
- 4 Visualizar Actas de Asignaturas
- 5 Visualizar Calificaciones
- 6 Visualizar Registro de Calificaciones
- 7 Visualizar Listados de Estudiantes por Grupos
- 8 Visualizar Listado Oficial de Estudiantes

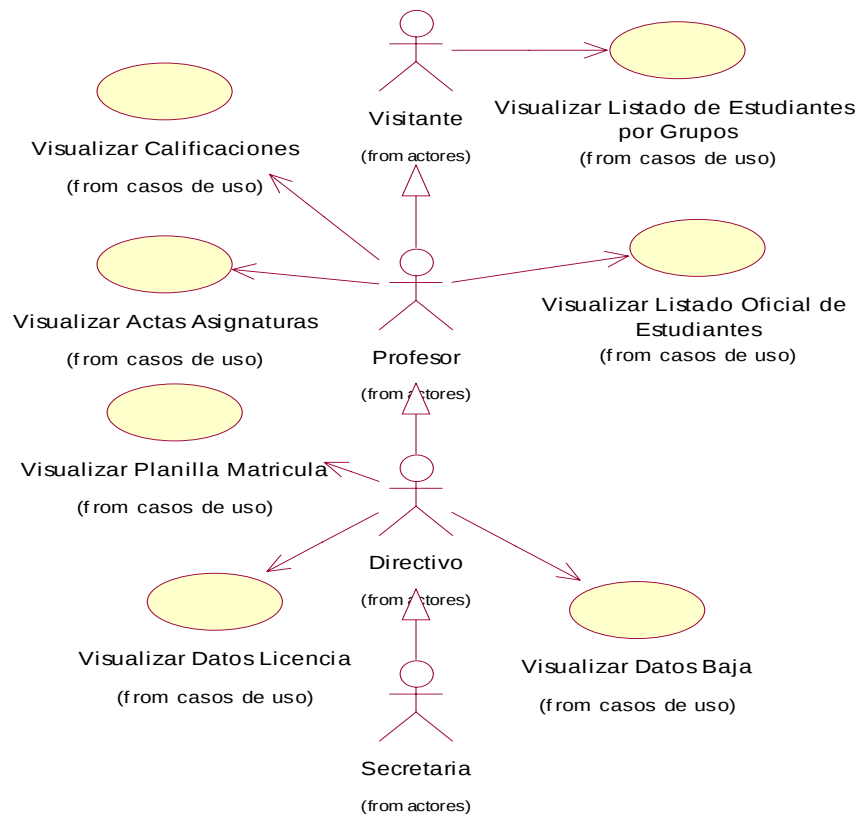


Fig. 17.- Diagrama de caso de Uso del Sistema: Paquete Reportes

El Paquete Administrativo (Figura 18) contiene los casos de usos referentes a la administración del sistema:

1. Crear Usuario.
2. Eliminar Usuario
3. Iniciar Sesión
4. Cambiar Contraseña.
5. Habilitar Usuario
6. Inhabilitar Usuario
7. Gestionar Efemérides.
8. Gestionar Noticias

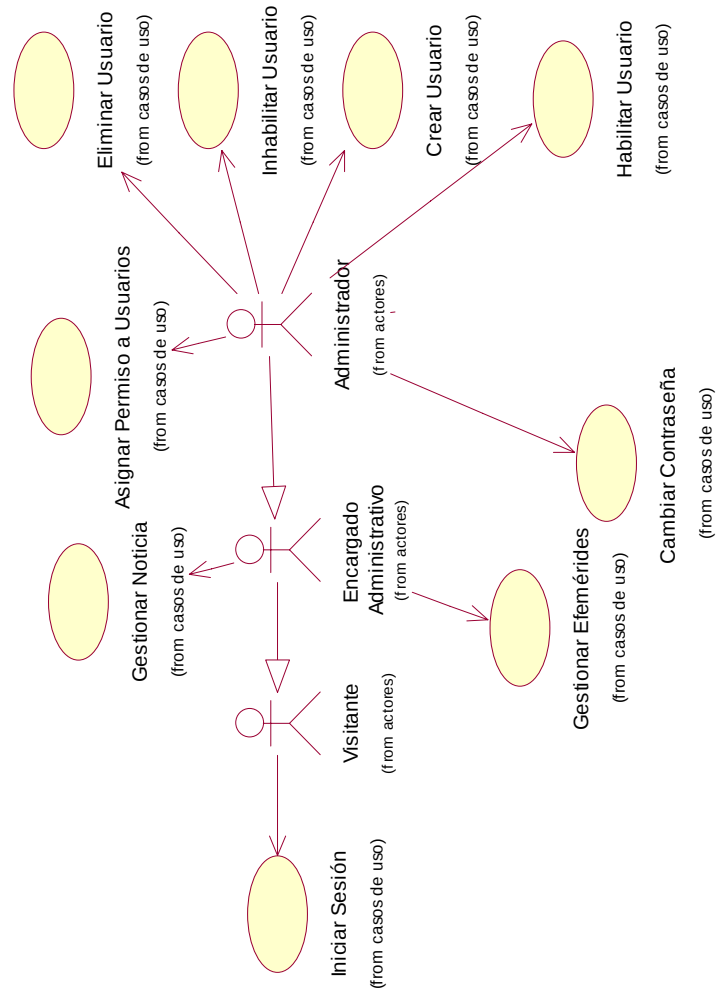


Fig. 18.- Diagrama de caso de Uso del Sistema: Paquete Administrativo

III.2.3. Descripción de los Casos de Uso

Caso de uso: Consultar Efemérides
Actores: Visitante (inicia).
Propósito: Consultar información sobre las efemérides
Resumen: El caso de uso de inicia cuando un visitante desea consultar las efemérides que están disponibles en los archivos del sistema. En la primera pantalla que consulta el usuario se mostrará un resumen de las efemérides del día, si la desea ver en detalles puede seleccionarla y el sistema muestra una descripción más ampliada de la misma. El caso de uso finaliza con la visualización de las efemérides.
Precondiciones:- Que exista al menos una efeméride almacenada en la base de datos.
Referencias: R 58
Poscondiciones:-
Prototipo: Ver anexo 3.1

Caso de uso: Consultar Noticias
Actores: Visitante (inicia).
Propósito: Mostrar noticias.
Resumen: El caso de uso inicia cuando un visitante desea conocer noticias referentes a la facultad o Universidad. En la primera pantalla se muestra el resumen de la noticia y si desea ver información más detallada el sistema lo muestra. El caso de uso finaliza cuando se visualiza la noticia.
Precondiciones:- Deben estar las noticias a visualizar en la base de datos.
Referencias: R59
Poscondiciones:-

Prototipo: Ver anexo 3.2
Caso de uso: Gestionar Efemérides
Actores: Encargado Administrativo (inicia).
Propósito: Permite mantener actualizado el archivo de las efemérides.
Resumen: El caso de uso se inicia cuando el encargado administrativo desea actualizar las efemérides, el sistema permite que pueda insertar, actualizar o eliminar efemérides. El caso de uso termina con la actualización de las efemérides.
Precondiciones: -Si lo que se va a realizar es actualizar o eliminar deben estar las efemérides en la base de datos.
Referencias: R4, R5, R6
Poscondiciones: Se actualiza las efemérides a mostrar. Si acción: insertar, se insertan efemérides. Si acción: actualizar, se actualizan efemérides. Si acción: eliminar, se eliminan efemérides
Prototipo: Ver anexo 3.3

Caso de uso: Gestionar Noticias
Actores: Encargado Administrativo (inicia).
Propósito: Permite mantener actualizado el registro de las noticias.
Resumen El caso de uso se inicia cuando el encargado administrativo desea actualizar las noticias, el sistema permite que pueda insertar, actualizar o eliminar noticias. El caso de uso termina con la actualización de las Noticias.

Precondiciones:
Las noticias a eliminar o modificar deben estar en la base de datos.
Referencias: R1, R2, R3
Poscondiciones:
Se actualiza las noticias a mostrar de la facultad.
Si acción: insertar, se insertan noticias.
Si acción: actualizar, se actualizan las noticias
Si acción: eliminar, se eliminan las noticias
Prototipo: Ver anexo 3.4

Caso de uso: Registrar Matrícula
Actores: Secretaria (inicia)
Propósito: Registrar los datos de los estudiantes que matriculan en una carrera determinada.
Resumen:
El caso de uso se inicia cuando la secretaria desea matricular un estudiante, para esto registra todos sus datos y finalmente imprimir una planilla para ser firmada y archivada.
Precondiciones:
El estudiante a matricular debe estar en el listado de pre-matrícula, proceder de otra facultad u otro CES. Tiene que haberse iniciado el curso escolar.
Referencias: R7
Poscondiciones: -Quedan registrados los datos de la persona en los archivos de matrícula.
Prototipo: Ver anexo 3.5

Caso de uso: Actualizar Matrícula
Actores: Secretaria (inicia)
Propósito: Actualizar datos pendientes o erróneos de los estudiantes

matriculados en la facultad.
Resumen: El caso de uso se inicia cuando la secretaria desea actualizar matrícula, esto puede estar ocasionado por omisión de datos o errores de escritura, el sistema muestra la planilla de matrícula dando la posibilidad de modificar los datos. El caso de uso termina quedando actualizada la matrícula del estudiante.
Precondiciones: Debe existir el registro del estudiante en los archivos de la facultad.
Referencias: R9,
Poscondiciones: Quedan actualizados los datos de la matrícula.
Prototipo: Ver anexo 3.5

Caso de uso: Conformar Listados de Grupos.
Actores: Secretaria (inicia).
Propósito: Conformar los listados de grupos por carreras en la facultad.
Resumen: El caso de uso se inicia una vez que estén matriculados los estudiantes, los estudiantes son divididos en tantos grupos como la secretaria proponga y ordenados alfabéticamente. El caso de uso finaliza con la confección de los listados de grupos.
Precondiciones: Los estudiantes a incorporar en los grupos deben estar matriculados previamente.
Referencias: R14
Poscondiciones: Quedan ubicados los estudiantes en sus grupos.
Prototipo: Ver anexo 3.6

Caso de uso: Re-Matricular Estudiantes.
Actores: Secretaria (inicia).

Propósito: Ratificar la matrícula de los estudiantes antes de comenzar el nuevo curso.
Resumen: El caso de uso inicia cuando al comenzar el curso escolar la secretaria actualiza los datos personales de los estudiantes y el curso escolar. El caso de uso termina con la re-matriculación del estudiante.
Precondiciones: El estudiante debe estar matriculado.
Referencias: R13
Poscondiciones: Quedan actualizados los datos de los estudiantes.
Prototipo: Ver anexo 3.7

Caso de uso: Registrar Datos Licencia
Actores: Secretaria (inicia).
Propósito: Registrar los datos de un estudiante que se le ha aprobado una licencia docente.
Resumen: El caso de uso inicia cuando a un estudiante se le aprueba una licencia docente, este proceso provoca cambios en los documentos del estudiante. El sistema da la posibilidad de llenar una planilla referente a la licencia que se imprime para ser adjuntada al expediente del estudiante. El caso de uso termina con el registro de la licencia.
Precondiciones: El estudiante debe ser matrícula de la facultad.
Referencias: R12
Poscondiciones: Queda actualizado el registro de licencia del estudiante.
Prototipo: Ver anexo 3.8

Caso de uso: Registrar Datos Baja
Actores: Secretaria (inicia).

Propósito:
Registrar la información de un estudiante al cual se le da baja de la facultad.
Resumen:
El caso de uso se inicia cuando un estudiante recibe baja de la facultad, este cambio provoca transformaciones en los documentos del estudiante, el sistema brinda la posibilidad de llenar una planilla de baja para imprimirla y adjuntarla al expediente. El caso de uso termina con el registro de la baja.
Precondiciones: El estudiante debe estar previamente matriculado en la facultad.
Referencias: R11
Poscondiciones: -Quedan actualizados los datos de baja del estudiante.
Prototipo: Ver anexo 3.9

Caso de uso: Iniciar Curso Escolar
Actores: Secretaria (inicia)
Propósito: Actualizar la configuración general del módulo docente.
Resumen:
La secretaria antes de iniciar el período de matrícula debe actualizar la configuración del módulo docente para esto debe indicar el inicio del curso escolar, el sistema actualiza los datos que se modifican con este proceso.
Precondiciones: Para ejecutar esta acción la fecha debe coincidir con el mes de agosto.
Referencias: R8
Poscondiciones: Queda iniciado el curso escolar y se puede comenzar con el proceso de matrícula.
Prototipo: Ver anexo 3.10

Caso de uso: Gestionar Codificador
Actores: Encargado Administrativo (inicia).
Propósito: Actualizar codificador, que puede ser codificador de sexo, raza, país, provincia, municipio, Nivel Educativo Padres, Clase de Estudiante, Tipo de Curso, Situación Escolar, Procedencia Escolar, Vía de Ingreso, Nivel Educativo, Concepto de Ingreso, Actividad Laboral, Cargo Ocupacional.
Resumen: El caso de uso se inicia cuando el encargado administrativo desea actualizar los codificadores, para esto se puede insertar o actualizar codificador. El caso de uso termina cuando se actualizan los codificadores.
Precondiciones:- Si se va a actualizar debe existir el codificador en la base de datos.
Referencias: R15-R44
Poscondiciones: Se actualizan los codificadores. Si acción: insertar, se inserta un nuevo codificador. Si acción actualizar, se actualiza la información de un codificador.
Prototipo: Ver Anexo 3.11

Caso de uso: Registrar Calificaciones.
Actores: Secretaria (inicia).
Propósito: Registrar las calificaciones de los estudiantes en las asignaturas.
Resumen: El caso de uso se inicia cuando la secretaria, desea registrar las calificaciones de los estudiantes, el sistema genera el acta preguntando el semestre, el grupo y la asignatura perteneciente al acta a generar. El caso de uso termina cuando se registran las calificaciones en el acta

generada por el sistema.
Precondiciones: Debe existir en la base de datos el grupo, la asignatura y el semestre.
Referencias: R47
Poscondiciones: Se registran las calificaciones del estudiante.
Prototipo: Ver anexo 3.12

Caso de uso: Modificar Calificaciones.
Actores: Secretaria (inicia).
Propósito: Modificar las calificaciones de los estudiantes en las actas de asignaturas, en caso de que los datos registrados sean incorrectos.
Resumen: El caso de uso se inicia cuando la secretaria, luego de tener todas las calificaciones registradas en las actas necesita cambiar los datos por error en la introducción de la información u otra causa que requiera de actualizar las actas. El caso de uso culmina cuando se actualizan las calificaciones del estudiante.
Precondiciones: La calificación a modificar debe estar registrada.
Referencias: R48
Poscondiciones: Quedan actualizadas las calificaciones de los estudiantes.
Prototipo: Ver anexo 3.12

Caso de uso: Conformar Actas de Asignaturas
Actores: Secretaria (inicia).
Propósito: Conformar Actas de Calificaciones de los Grupos
Resumen: El caso de uso se inicia cuando la secretaria desea conformar las actas de calificaciones de los estudiantes, esta selecciona el grupo, el semestre y la asignatura para la cual construirá el acta,

culminando el caso de uso con su confección.
Precondiciones: La asignatura a realizar el acta debe pertenecer al semestre seleccionado.
Referencias: R48
Poscondiciones: Queda conformada el acta de calificaciones.
Prototipo: Ver anexo 3.12

Caso de uso: Visualizar Listado Oficial de Estudiantes.
Actores: Profesor (inicia)
Propósito: Ofrecer un listado de los estudiante con los datos personales.
Resumen: El caso de uso se inicia cuando los profesores desean visualizar el listado oficial de los estudiantes. El sistema muestra resumido por grupos los datos personales del estudiante, culminando así el caso de uso.
Precondiciones:- El grupo a visualizar el listado debe existir en la facultad
Referencias: R56
Poscondiciones:-
Prototipo: Ver anexo 3.13

Caso de uso: Visualizar Planilla de Matrícula
Actores: Directivo (inicia).
Propósito: Mostrar la planilla de matrícula de los estudiantes con los datos referentes al documento.
Resumen: El caso de uso de inicia cuando una vez tomados todos los datos necesarios para llenar la planilla de matrícula de los estudiantes se

muestra el modelo, a partir de este se puede revisar la planilla para imprimirla. El caso de uso culmina visualizando la planilla de matrícula.
Precondiciones: La planilla a mostrar debe estar ya creada.
Referencias: R50
Poscondiciones:-
Prototipo: Ver anexo 3.14

Caso de uso: Visualizar Datos de Baja
Actores: Directivo (inicia).
Propósito: Mostrar la planilla de baja de un estudiante.
Resumen: La secretaria, después de crear la planilla de baja del estudiante, la muestra con los datos correspondientes al estudiante que se le procesa la baja, luego puede imprimirla. El caso de uso culmina visualizando la planilla de baja.
Precondiciones: La planilla a mostrar debe ser previamente creada.
Referencias: R51
Poscondiciones:
Prototipo: Ver anexo 3.15

Caso de uso: Visualizar Datos de Licencia
Actores: Directivo (inicia).
Propósito: Mostrar la planilla de licencia de un estudiante
Resumen: El caso de uso se inicia cuando la secretaria, después de crear la planilla de licencia del estudiante, la muestra con los datos correspondientes al estudiante que se le procesa la licencia, luego puede imprimirla. El caso de uso termina visualizando la planilla de

licencia del estudiante.
Precondiciones: La planilla a visualizar debe ser previamente creada
Referencias: R52
Poscondiciones:
Prototipo: Ver anexo 3.16

Caso de uso: Visualizar Acta de Asignatura
Actores: Profesor (inicia).
Propósito: Mostrar el acta de evaluaciones de un grupo en una asignatura
Resumen: El caso de uso de inicia cuando la secretaria después de haber creado el acta de asignatura puede mostrarla para imprimir sus datos. Culmina el caso de uso cuando se visualiza el acta de evaluación.
Precondiciones: El acta a mostrar debe ser previamente creada.
Referencias: R53
Poscondiciones:
Prototipo: Ver anexo 3.17

Caso de uso: Visualizar Calificaciones
Actores: Profesor (inicia).
Propósito: Mostrar las calificaciones de los estudiantes.
Resumen: El caso de uso se inicia cuando una vez actualizadas las actas de las asignaturas se puede mostrar las calificaciones de los estudiantes, culminando así el caso de uso.
Precondiciones: Las calificaciones deben estar previamente registradas.
Referencias: R54
Poscondiciones:

Prototipo: Ver anexo 3.18

Caso de uso: Visualizar Listados de Estudiantes por Grupos
Actores: Visitante (inicia).
Propósito: Mostrar los listados de estudiantes por grupos.
Resumen: El caso de uso se inicia cuando un visitante solicita visualizar el listado de estudiantes por grupos, culminando el caso de uso mostrando dicho listado.
Precondiciones: Debe haberse hecho la distribución de los grupos con anterioridad.
Referencias: R57
Poscondiciones:
Prototipo: Ver anexo 3.19

Caso de uso: Visualizar Listados Oficial de Estudiantes.
Actores: Profesor (inicia).
Propósito: Mostrar los listados de estudiantes por grupos.
Resumen: El caso de uso se inicia cuando el profesor, el estudiante, el directivo desean visualizar el listado oficial de estudiantes por grupos. El sistema muestra la distribución de estudiantes por grupos con algunos datos de interés sobre el estudiante, culminando así el caso de uso.
Precondiciones: Debe haberse hecho la distribución de los grupos con anterioridad.
Referencias: R57
Poscondiciones:
Prototipo: Ver anexo 3.20

Caso de uso: Iniciar Sesión
Actores: Visitante (inicia).
Propósito: Registrarse en el sistema
Resumen: El caso de uso se inicia cuando la persona universitaria desea acceder a información o servicios que se brindan a los usuarios que tienen determinados permisos, para esto debe introducir su cuenta y su contraseña. El sistema verifica en la base de datos si los datos introducidos son correctos y pasa a mostrar su perfil, culminando así el caso de uso.
Precondiciones: Para poder registrarse con satisfacción debe existir este usuario con esta contraseña.
Referencias: R61
Poscondiciones: El usuario puede acceder a toda la información a la que tiene permisos.
Prototipo: Ver anexo 3.21

Caso de uso: Crear Usuario
Actores: Administrador (inicia).
Propósito: Añadir un nuevo usuario al sistema.
Resumen: El caso de uso se inicia cuando el administración desea usuarios que necesiten trabajar con la información que requiere de usuarios registrados.
Precondiciones:
Referencias: R62
Poscondiciones: Queda registrado el usuario y listo para asignarle los permisos ante el

sistema.
Prototipo: Ver anexo 3.22

Caso de uso: Habilitar Usuario
Actores: Administrador (inicia).
Propósito: Habilitar la cuenta de un usuario del sistema.
Resumen: El caso de uso se inicia cuando el administrador desea activar la cuenta de un usuario para el uso del sistema.
Precondiciones: El usuario a habilitar debe estar previamente registrado como usuario del sistema.
Referencias: R67
Poscondiciones: Quedan activada la cuenta del usuario.
Prototipo: Ver anexo 3.23

Caso de uso: Inhabilitar Usuario
Actores: Administrador (inicia).
Propósito: Inhabilitar la cuenta de un usuario del sistema.
Resumen: El caso de uso se inicia cuando el administrador desea desactivar la cuenta del usuario para el uso del sistema.
Precondiciones: El usuario a inhabilitar debe estar previamente registrado como usuario del sistema.
Referencias: R68
Poscondiciones: Queda desactivada la cuenta del usuario.
Prototipo: Ver anexo 3.24

Caso de uso: Asignar Permisos a Usuarios
Actores: Administrador (inicia).
Propósito: Dar a los usuario permisos en los módulos.
Resumen: El caso de uso se inicia cuando el administrador desea asignar permisos a los usuarios del sistema, se muestran los módulos y los permisos posibles en cada módulo para seleccionarlos.
Precondiciones: El usuario al que se le van a asignar los permisos debe estar previamente registrado como usuario del sistema.
Referencias: R65
Poscondiciones: Quedan asignados los permisos del usuario.
Prototipo: Ver anexo 3.25

Caso de uso: Cambiar Contraseña
Actores: Administrador (inicia).
Propósito: Permitir que la persona registrado cambie su contraseña
Resumen: El caso de uso se inicia cuando un usuario registrado desea cambiar su contraseña, el sistema le da la posibilidad de poner su cuenta, la contraseña antigua y la nueva, si los datos son correctos la contraseña queda cambiada, de lo contrario se muestra un mensaje de error, finalizando así el caso de uso.
Precondiciones: Debe existir información almacenada de este usuario.
Referencias: R63
Poscondiciones: Queda actualizada la contraseña del usuario
Prototipo: Ver anexo 3.26

Caso de uso: Eliminar Usuario
Actores: Administrador (inicia).

Propósito: Elimina la cuenta de un usuario en el sistema.
Resumen: El caso de uso se inicia cuando el administrador desea eliminar un usuario del sistema, quedando a su vez impedido el acceso a cualquier información que requiera del logueo previo.
Precondiciones: Debe existir el usuario a eliminar.
Referencias: R19
Poscondiciones: Se elimina el usuario del sistema.
Prototipo: Ver anexo 3.27

Caso de uso: Consultar Información General
Actores: Visitante (inicia).
Propósito: Consultar Información General mostrada en el sistema.
Resumen: El caso de uso se inicia cuando un visitante accede al sitio y consulta información general mostrada para todos los usuarios que accedan al sitio aunque no sean usuarios registrados del sistema.
Precondiciones:
Referencias: R69
Poscondiciones:
Prototipo: Ver anexo 3.32

III.3. Conclusiones del Capítulo

En este capítulo han sido descritos los casos de uso del sistema, así como los actores que están asociados con ellos, utilizando el formato de alto nivel. Se hace mención de las precondiciones y poscondiciones del caso de uso.

Capítulo IV: Diseño e Implementación del Sistema

En este capítulo se presenta el diseño de clases propuesto para desarrollar el sistema y los diagramas necesarios para la implementación de la solución propuesta.

IV.1. Principios de Diseño del Sistema

El **diseño de sistemas** se define como el proceso de aplicar ciertas técnicas y principios con el propósito de definir un dispositivo, un proceso o un sistema, con suficientes detalles como para permitir su interpretación y realización física.

IV.1.1. Estándares en la Interfaz de la Aplicación

La primera impresión del usuario cuando visita una aplicación web la brinda el diseño de la interfaz. Es por ello que, para lograr la apariencia adecuada y que el usuario se sienta confortable, se tienen en cuenta varios aspectos, sobre todo relacionados con tipografía, colores, gráficos, navegación, composición del sitio, etc., que a continuación se detallan. En el sistema, el diseño de la interfaz está basado en páginas Web, se utilizan las tonalidades de azul por ser suaves y refrescantes en concordancia con los colores del logo de la universidad y facultad. El vocabulario manejado es lo menos técnico posible, acercándose al utilizado por los usuarios.

Se utilizan imágenes identificativas como vínculos para la navegación dentro del sitio web. La letra utilizada en todo el sistema es Times New Roman (12, 16) lográndose un diseño estándar en todo el sitio. Los mensajes de error son pequeños y en Español

Se utilizan pequeños íconos para una mayor comprensión de las acciones, aunque se seleccionaron imágenes consecuentes con el significado que se quiere transmitir, cada imagen tiene asociado su ayuda rápida con el nombre de la acción que realiza.

El fondo de las páginas es de color blanco para mayor frescura de la vista. Todo esto se ha hecho con el objetivo de que el uso del sitio brinde comodidad y confort al usuario.

IV.1.2. Formatos de Reportes

Los reportes presentan un diseño similar a los reportes que se llevan en la facultad de forma manual pero a través de la web con el objetivo de que los cambios para quienes operan con los reportes sean mínimos.

IV.1.3. Concepción General de la Ayuda

La ayuda constituye una parte imprescindible en todo sistema. En el menú principal aparece una opción *Ayuda* que explicara de forma detallada como funciona el sistema, tratando de aclarar los puntos que podría causar duda al usuario. Este menú aparece en todas las páginas para mayor comodidad del usuario.

Cada una de las opciones del sistema, así como las consideraciones que se asumen en la ejecución de ellas están propiamente documentadas para evitar cualquier tipo de confusión por parte del usuario. Cada aspecto de la ayuda ha sido diseñado con el objetivo de expresar explícitamente cómo y en qué orden debe operar el usuario.

Al inicio de cada módulo se muestra una explicación del objetivo del módulo en sentido general.

IV.1.4. Tratamiento de Excepciones

El diseño de la interfaz ha estado dirigido a evitar errores, teniendo en cuenta paralelamente la creación de interfaces útiles y amigables. Se ha buscado simplificar la validación de los datos garantizando una validación intrínseca de los mismos, procurando facilitar la corrección de errores lógicos tanto en la introducción de la información como en cualquier otro momento del tratamiento de la misma.

La técnica para el manejo de los errores en el sistema se concebirá de manera que cuando ocurra un error se genere una excepción; es decir, la ejecución normal se detenga y se transfiera el control a la zona de tratamiento de excepciones. Las excepciones internas se generan automáticamente por el sistema.

Los mensajes de error que emita el sistema ya sea de la base de datos o de la aplicación cliente se captarán y se traducirán a un lenguaje comprensible para el usuario.

Los formularios manejan los datos en memoria y sólo se actualiza en la base de datos cuando se indique salvarlos.

IV.2. Modelo de Clases Web

Un diagrama de clases Web representa las colaboraciones que ocurren entre las páginas, donde cada página lógica puede ser representada como una clase. Al tratar de utilizar el diagrama de clases tradicional para modelar aplicaciones Web surgen varios problemas, por lo cual los especialistas del Rational plantearon la creación de una extensión al modelo de análisis y diseño que permitiera representar el nivel de abstracción adecuado y la relación con los restantes artefactos de UML.

Caso de Uso	Diagrama de Clases Web
Consultar Efemérides	Anexo 4.1
Consultar Noticias	Anexo 4.2
Gestionar Efemérides.	Anexo 4.3
Gestionar Noticias	Idem 4.3 (Noticias)

Tabla 5. Diagrama de Clases. Paquete Módulo General

Caso de Uso	Diagrama de Clases Web
Registrar Matrícula.	Anexo 4.4
Actualizar Matrícula.	Anexo 4.5
Conformar Listados de Grupos.	Anexo 4.6
Re-Matricular Estudiantes.	Anexo 4.7
Registrar Datos Licencia.	Anexo 4.8
Registrar Datos Baja.	Idem 4.8(Baja)
Iniciar Curso Escolar	Anexo 4.9

Tabla 6. Diagrama de Clases. Paquete de Matrícula

Caso de Uso	Diagrama de Clases Web
Gestionar Codificador	Anexo 4.10

Tabla 7. Diagrama de Clases Web: Paquete Codificadores

Caso de Uso	Diagrama de Clases Web
Conformar Actas de Asignaturas	Anexo 4.11
Modificar Calificaciones	Anexo 4.11
Registrar Calificaciones	Anexo 4.11

Tabla 8. Diagrama de Clases Web: Paquete Evaluaciones

Caso de Uso	Diagrama de Clases
Visualizar Planilla de Matrícula	Anexo 4.12
Visualizar Planilla de Baja	Idem 4.12(Baja)
Visualizar Planilla de Licencia	Idem 4.12(Licencia)
Visualizar Actas de Asignaturas	Anexo 4.13
Visualizar Calificaciones	Anexo 4.13
Visualizar Listados de Estudiantes por Grupos	Anexo 4.14

Tabla 9. Diagrama de Clases Web: Paquete Reportes

Caso de Uso	Diagrama de Clases
Crear Usuario.	Anexo 4.15
Iniciar Sesión	Anexo 4.16
Cambiar Contraseña.	Anexo 4.17
Dar Permisos Usuario	Anexo 4.18
Eliminar Usuario	Anexo 4.19
Habilitar (Inhabilitar) Usuario	Anexo 4.20

Tabla 10. Diagrama de Clases Web: Paquete Administrativo

IV.3. Diseño de la Base de datos

Por la importancia de los datos manejados en el módulo docente de la Intranet de una Facultad es necesario lograr un buen diseño de la información almacenada.

En este epígrafe se muestra el diseño de la base de datos del sistema propuesto a través de los diagramas de clases persistente y el esquema de la base de datos generados a partir de este, con el modelo de datos.

IV.4. Diagrama del modelo lógico de datos.

El diagrama de clases persistentes muestra todas las clases capaces de mantener su valor en el espacio y en el tiempo. Se muestra en el Anexo 4.25.

IV.5. Diagramas del modelo físico de datos.

El modelo de datos que muestra la estructura física de las tablas de la base de datos, obtenido a partir del diagrama de clases persistentes, es mostrado en el Anexo 4.24.

IV.6. Diagramas de Implementación

El modelo de implementación describe cómo los elementos del modelo de diseño se implementan en términos de componentes. Describe también cómo se organizan los componentes de acuerdo con los mecanismos de estructuración y modularización disponibles en el entorno de implementación y en el lenguaje o lenguajes de programación utilizados y cómo dependen los componentes unos de otros. **[15]**

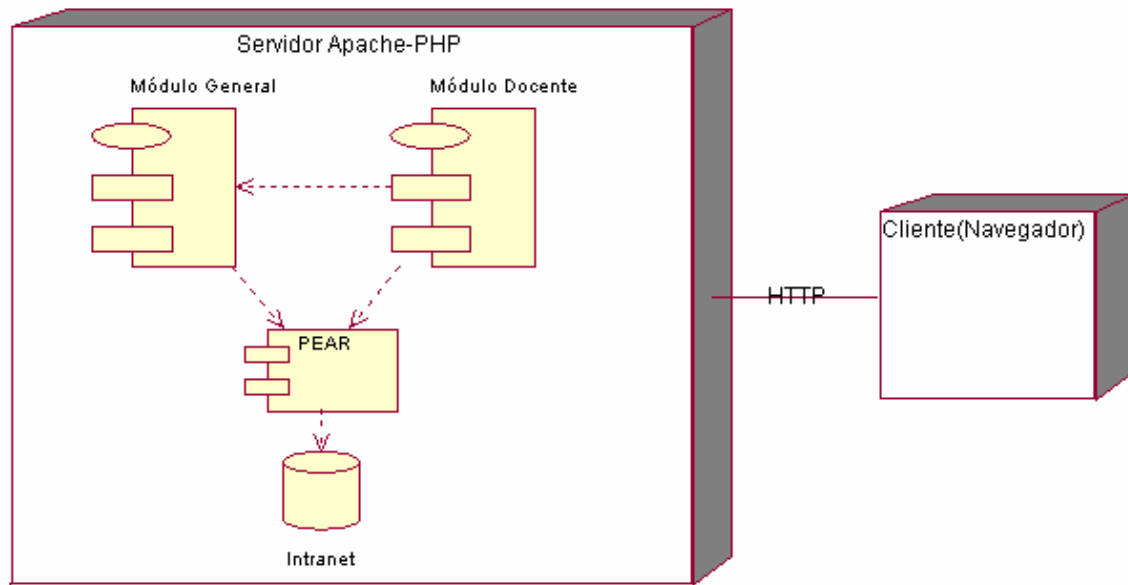


Fig. 19.- Diagrama de Implementación

IV.7. Conclusiones del Capítulo

En el presente capítulo se desarrollaron los diagramas de clases de la aplicación, el diseño de la base de datos y el modelo de despliegue del sistema. Se describieron, además, los principios de diseño seguidos, específicamente, los temas de estándares de la interfaz, concepción del tratamiento de errores, sistema de ayuda y principios de codificación. Se muestran los resultados de las etapas de diseño e implementación del sistema propuesto.

Conclusiones

La gestión de la información en una facultad de un CES lleva consigo varios procesos, que de forma manual resultan engorrosos y lentos, por lo que la realización de un sistema informático basado en el modelo de una Intranet resultaría beneficioso y agilizaría de forma extraordinaria el manejo de los datos que la entidad necesita.

Después de haber realizado un estudio de los flujos de trabajos de la facultad de Informática de la Universidad de Cienfuegos se presenta una propuesta de sistema que basa su funcionamiento en los estándares de la W3C para la web. Esta propuesta permite a los usuarios de la red (personal de la Facultad) producir y consumir información, con la cual se logra un mayor aprovechamiento de la jornada laboral del personal docente y no docente, así como una reducción del tiempo necesario para la visualización de reportes una vez que han sido realizados los registros correspondientes en los archivos ahora digitales.

El estudio de los flujos de trabajos, de información, así como los roles que juega cada personal de la facultad en estos flujos permitió:

- diseñar y establecer niveles de acceso a la información
- diseñar un modelo de datos que abarca el proceso docente-educativo, así como otros generales que tienen relación con el primero.

La aplicación está provista de un ambiente cómodo, fácil de entender, que cumple los estándares de diseño para web y utiliza técnicas de programación orientada a objetos. Se utilizó como servidor de aplicaciones Apache, PHP como lenguaje de programación del lado del servidor con las bibliotecas de objetos PEAR lo que nos permitió lograr un sistema multiplataforma en cuanto a Sistema Operativo y Gestor de Base de Datos.

Se cumplió con el objetivo general del trabajo siguiendo la metodología RUP, y utilizando el lenguaje UML para la modelación del sistema.

Recomendaciones

A pesar de que los objetivos trazados con la realización de este trabajo fueron cumplidos, los desarrolladores del mismo sugieren tomar esta propuesta solo como la primera fase de un proyecto mucho más ambicioso y continuar esta investigación.

Se recomienda entonces:

- Continuar con el estudio de los flujos de trabajos y de información de la facultad para abarcar todos los aspectos referidos a esta, así como profundizar en los aspectos tratados con mayor detalle.
- Extender su uso a otras facultades del centro para la validación de su funcionamiento.
- Mantener la concepción base utilizada de los servicios web para garantizar la interoperabilidad con otros módulos de otros sistemas, ya sean propios o desarrollados por terceros.
- Conservar la idea de que el sistema debe ser desarrollado sobre la base de la multiplataforma lo que permitirá adaptarse a las nuevas tecnologías de la informática moderna.

Bibliografía

Acosta Duarte, Dionel A. Registros Médicos Pediátricos. / Dionel Acosta Duarte. __ __
[s.l. : s.n.], 2002.__ __. 127 p.

Aronow, D.B. *Information technology applications in quality assurance and quality improvement*, The Joint Commision Journal on Quality Improvement, (E.U) 19, (9): 465-477, 1993

Bennatan, E.M. *Software Project Management: A Practitioner's Approach* / E.M Bennatan, __ __E.U: Editorial McGraw Hill, 1992.__ __89 p.

Boehm, B. *Software Engineering Economics* / B. Bohem.__ __E.U: Prentice Hall, 1981. __ __101 p.

Booch, Grady. *Análisis y Diseño Orientado a Objetos.* / Grady Booch, Díaz de Santos __ __ E. U: Ed. Addison-Wesley ,1996. __ __300p.

Date, C.J. *An Introduction to Database Systems.* /C. J. Date. __ __ [s.l.: s.n.], 1997.__ __24p.

Díaz Ferrera, Carlos N. *Sistema de Almacenamiento de la Información.* / Carlos Newton Díaz Ferrera. __ __ Ciudad de la Habana: Editorial ENPES, 1982.__ __ 98p

Fairley, Richard. *Ingienería del Software* /. Richar Fairley.__ __Mexico: Mc Graw-Hill, 1996.__ __ 300p.

Frost. Bases de datos y sistemas expertos. / Frost . __ [s.l.: s.l.], [198?].__ T2.

García Hernández, L. Bases de datos. / Lucina García Hernández, Anatoli Iosifovich .
__ C. Habana: Editorial “Evelio Rodríguez Curbelo”, 1989.__ 400p

Jones, C. Programming Productivity / C.Jones. __ EE UU : McGraw Hill, 1986. __
200p.

Kerr, J. Inside RAD./ J. Kerr, R Junter __ EE UU: Editorial Mc Graw Hill, 1994. __
200p.

Pressman, Roger. Ingeniería de Software, un enfoque práctico./ Roger Pressman. __
_E.U: Mc Graw-Hill, 1998__ 450p.

Ruble, David A. “Análisis y Diseño Práctico para Sistemas Cliente/Servidor con GUI”/
David A. Rouble.__[sl: sn], 1997. __ 600p.

Rumbaugh, J. El Lenguaje Unificado de Modelado. Manual de Referencia. / J.
Rumbaugh, Jacobson, G Booch.__ *Madrid: Pearson Education, 2000.* __ 500p .

SQL Server Books Online. / Microsoft Co. __ EE. UU: Microsoft Press, 1998.__
236p.

Taylor, E.S. An Interim Report on Engineering Design/ E.S Taylor __ E.U:
Cambridge, Massachuett Institute of Technology, 1989. __ 234p.

Teory TJ. JP. "A logical design methodology for relational database using the extended entity-relationship method". ACM Computing Survey (E.U) 18, (2): 8, Junio 1986.

Ullman JD. "Principles of Database Systems". / J.D Ullman . _ _ E U: Computer Science Press, 1983. _ _ 150p.

Capítulo V: Referencias Bibliográficas

- [1] Barberá Heredia, José. [Manual]. "Intranet un nuevo concepto para el manejo de la información corporativa" tomado de : <http://www.internet.com/manuales.htm> , (20/1/006
- [2] Letelier Torres Patricio "Desarrollo de Software Orientado a Objeto usando UML", Departamento Sistemas Informáticos y Computación, Universidad Politécnica de Valencia, España. URL: <http://www.creangel.com/uml/intro.php> , Mayo, 2004
- [3] Ferrá Grau Xavier "Desarrollo orientado a objetos con UML" Universidad Politécnica de Madrid, 2004 URL: <http://www.clikear.com/manuales/uml/introduccion.asp>, abril 2006
- [4] Díaz Antón Maria Gabriela, Angélica Pérez Maria "Propuesta de una metodología de desarrollo de software educativo bajo un enfoque de calidad sistemática" 2004. URL: <http://www.academia-interactiva.com/ise.pdf>, Mayo, 2006.
- [5] Teruel, Alejandro. Introducción a la arquitectura de capas. Tomado de: <http://www ldc.usb.ve/~teruel/ci3715/clases/argCapas.html> (26/012/06)
- [6] Macromedia Dreamweaver MX. Getting Started. Ayuda. (11/04/06)
- [7] Munz, Stefan. Tecnologías Web/ ASP. Tomado de:

<http://es.selfhtml.org/introduccion/tecnologias/asp.htm> (3/02/06)

[8] Idem. Tomado de:

<http://es.selfhtml.org/introduccion/tecnologias/java.htm> (31/02/06)

[9] [PHP, 2004] - Introducción a php. www.ciberteca.net/webmaster/php.

(20/01/2006)

[10] Rodríguez Luis [Manual] “Conoce a Pear: El Repositorio de Aplicaciones

y Extensione de PHP Tomadp de:

http://programacion.com/blogs/60_php_extensiones/php.htm .(4/01/06)

[11] Munz, Stefan. *XML y derivados de XML*. Tomado de:

<http://es.selfhtml.org/introduccion/tecnologias/webservices.htm> (10/01/06)

[12] Idem. Tomado de:

<http://es.selfhtml.org/introduccion/tecnologias/soap.htm> (10/01/06)

[13] Idem. Tomado de:

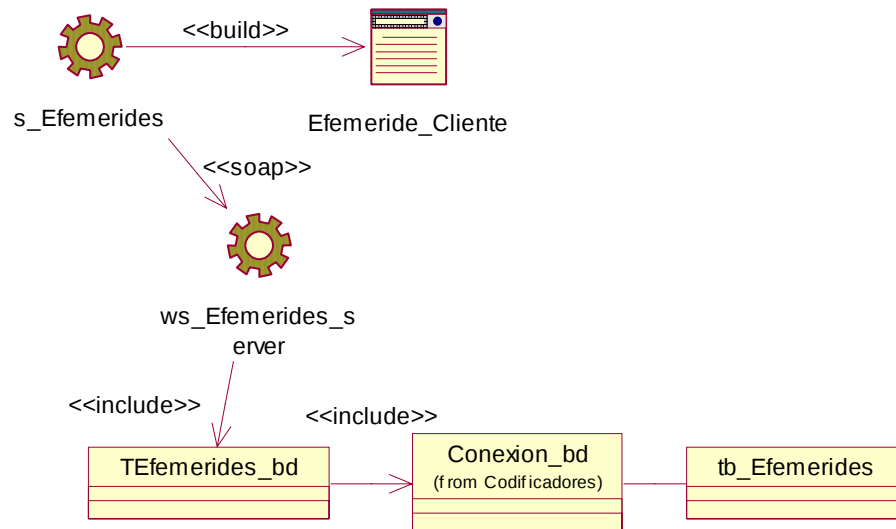
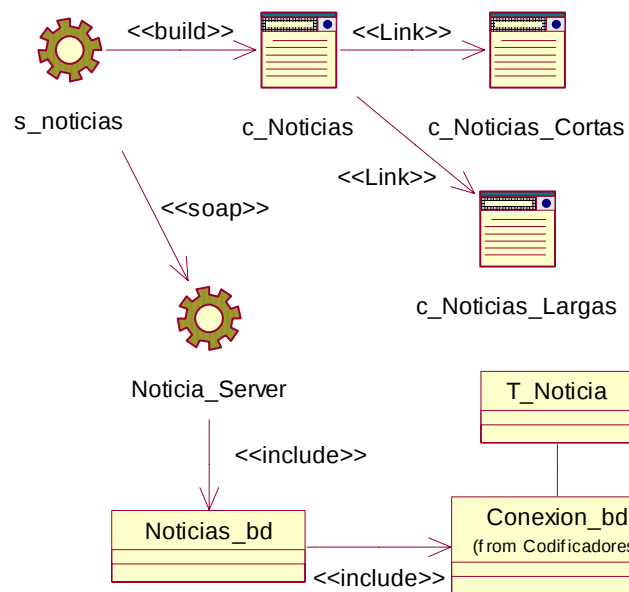
<http://es.selfhtml.org/introduccion/tecnologias/xml.htm> (10/01/06)

[14] Matos, Rosa María. *Introducción al trabajo con Base de Datos*.

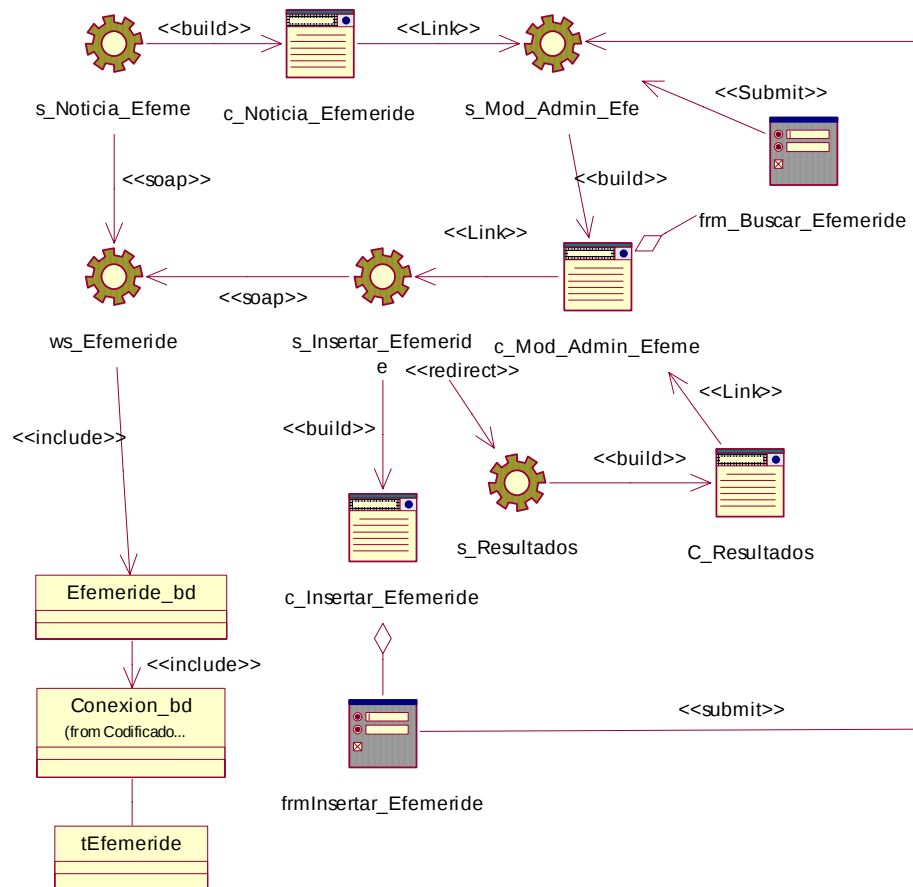
Asignatura de Sistemas de Gestión de Base de Datos.-- Ciudad Habana: CUJAE, 2004.-- p.10.

[15] Jacobson, I.; Booch, G. y Rumbaugh, J.; “El Proceso Unificado de Desarrollo de software”. 2000. Addison-Wesley. 115 p.

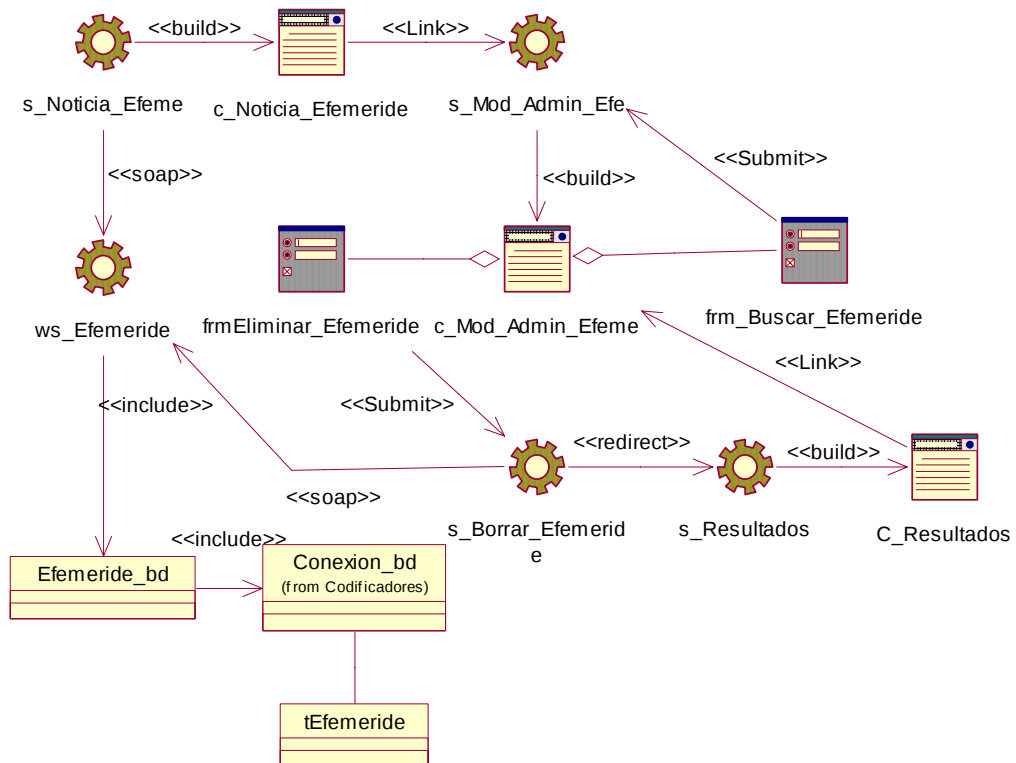
- [16] Idem, pp. 115-118p.
- [17] Idem,. 116 p.
- [18] Idem, 117p.
- [19] Idem, 121p.
- [20] Idem, 125p.
- [21] Idem, 127p
- [22] Idem, 130p
- [23] Idem, 127p

Anexo 4.1 Diagrama de Clases Web Consultar Efemérides.**Anexo 4.2 Diagrama de Clases Web Consultar Noticias**

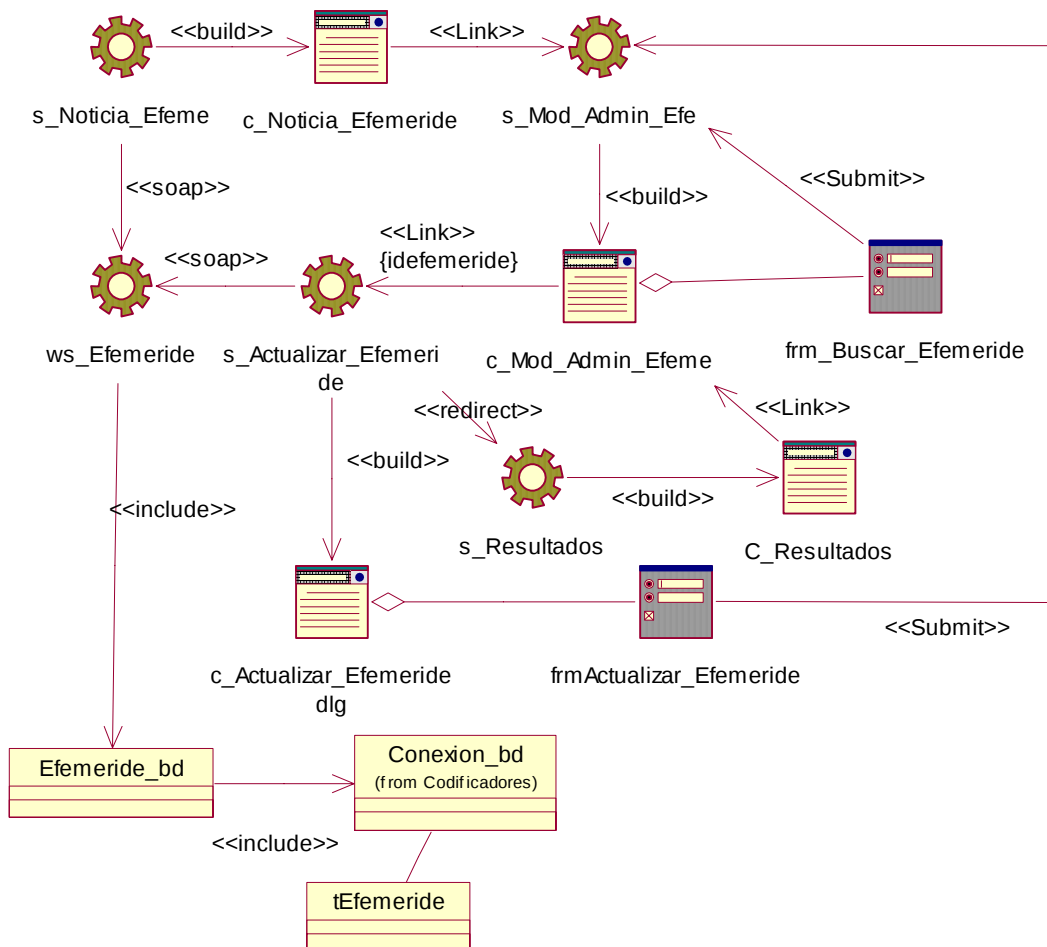
Anexo 4.3 Diagrama de Clases Web Gestionar Efemérides.



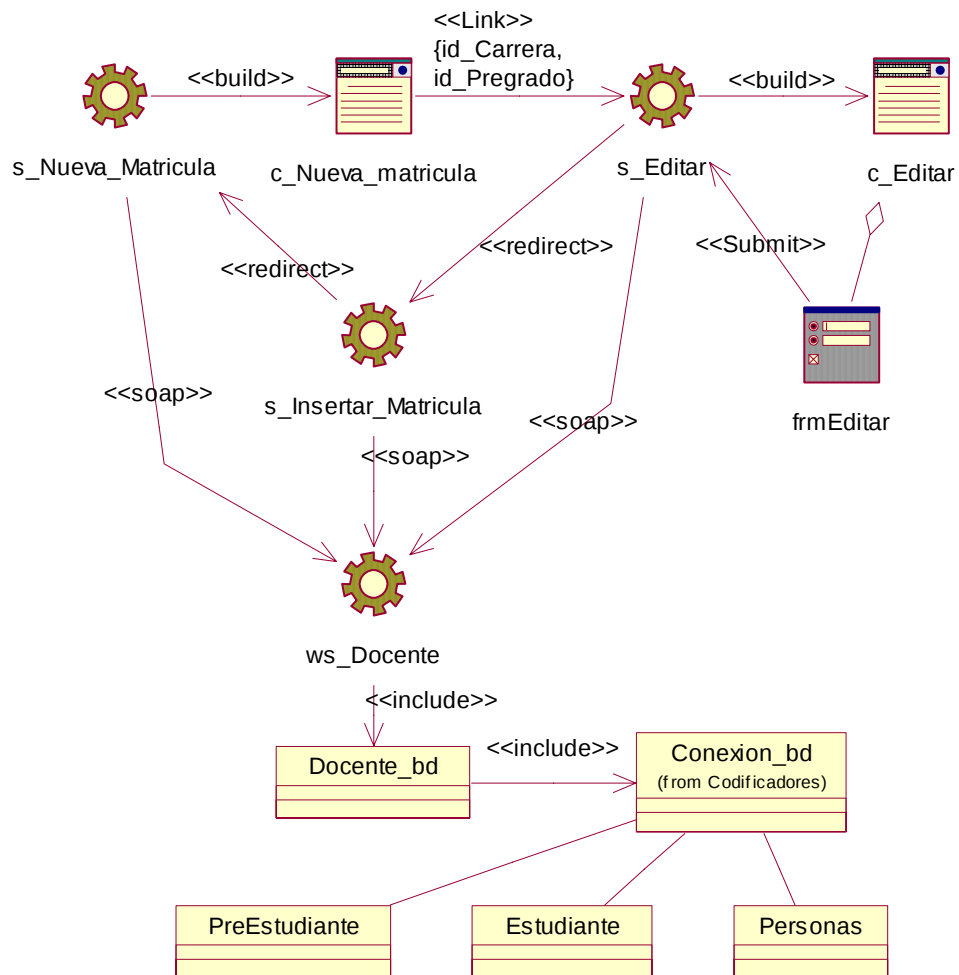
Insertar Efeméride

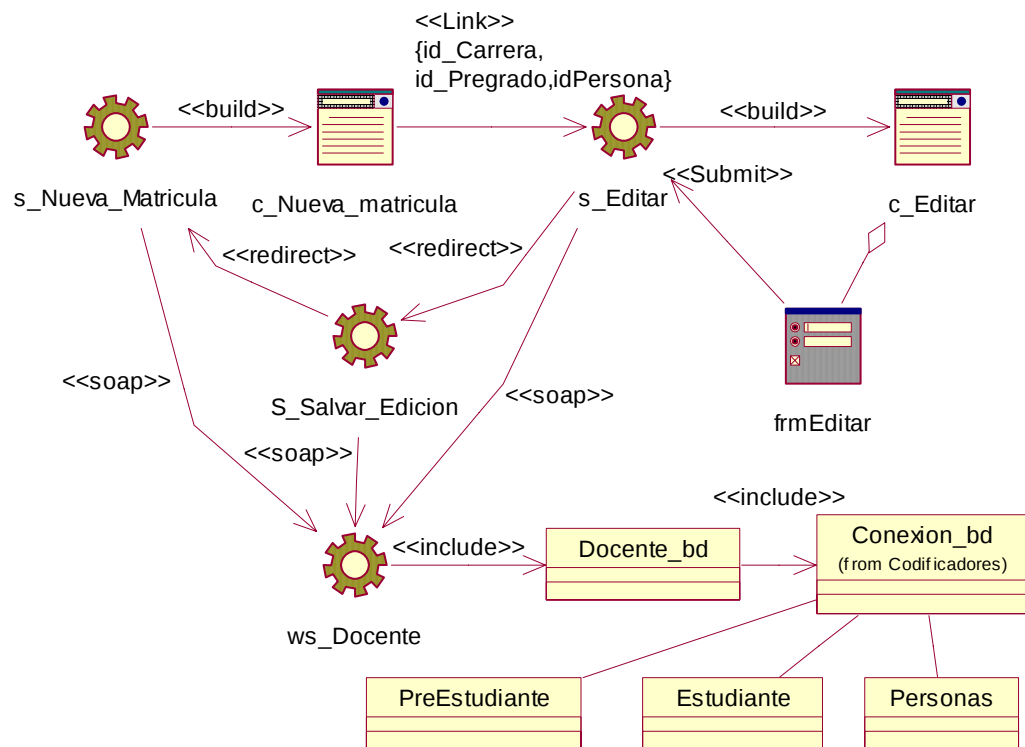


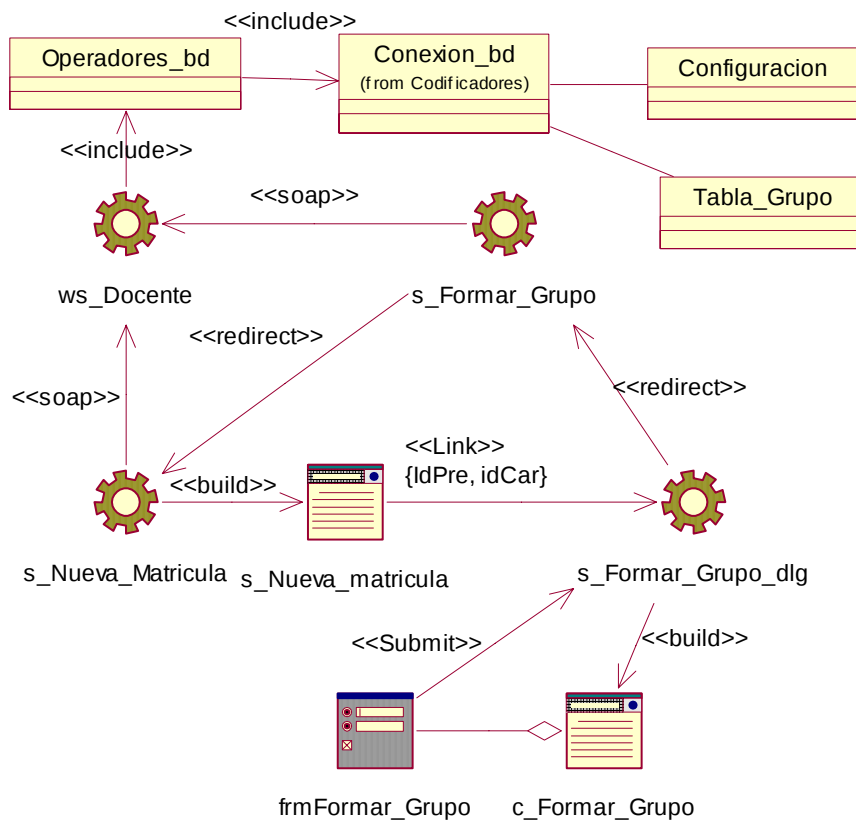
Borrar Efeméride

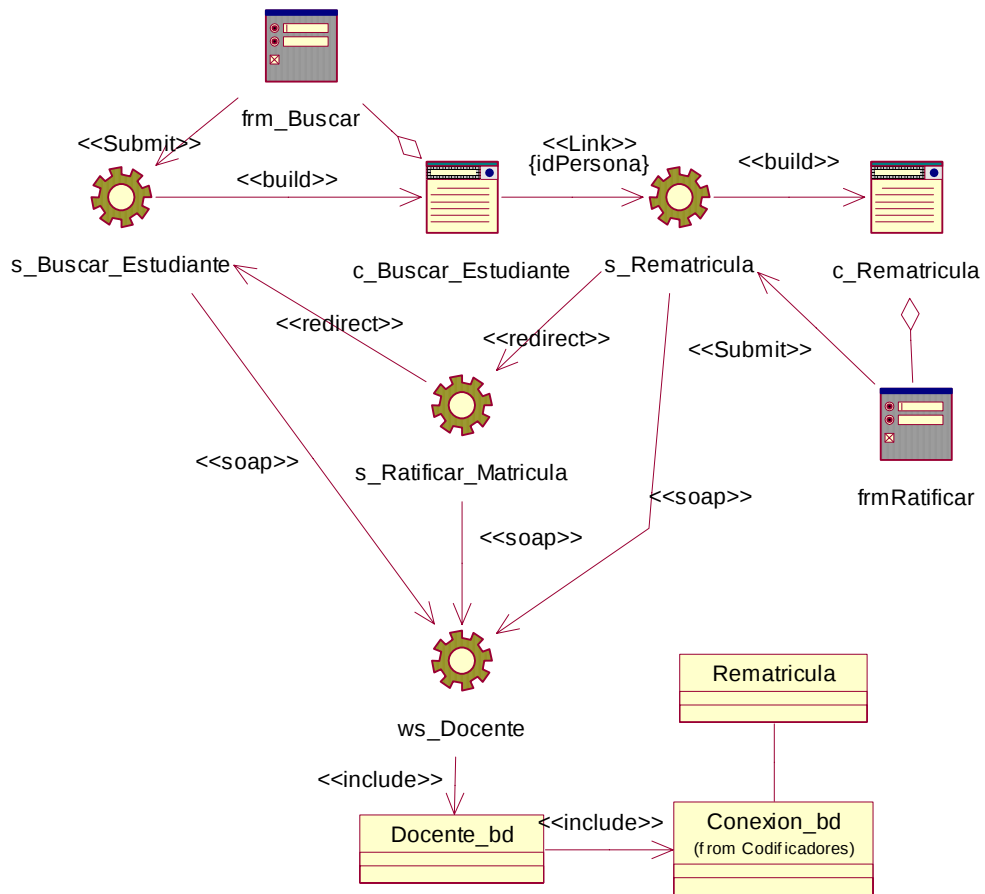


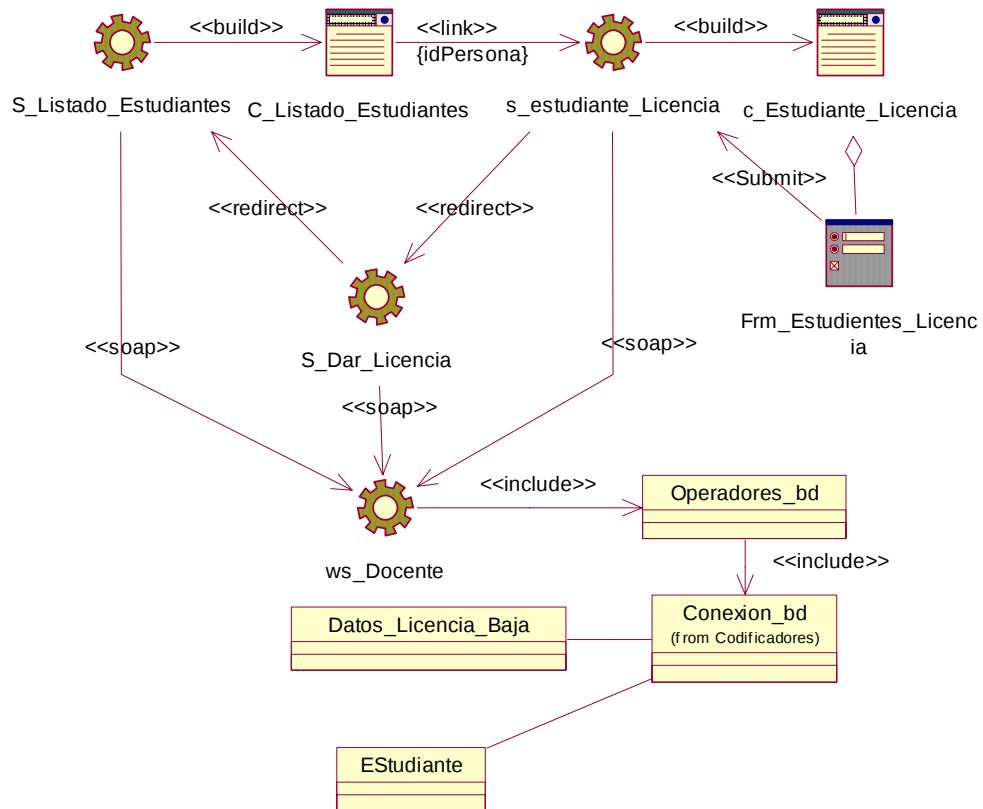
Actualizar Efeméride

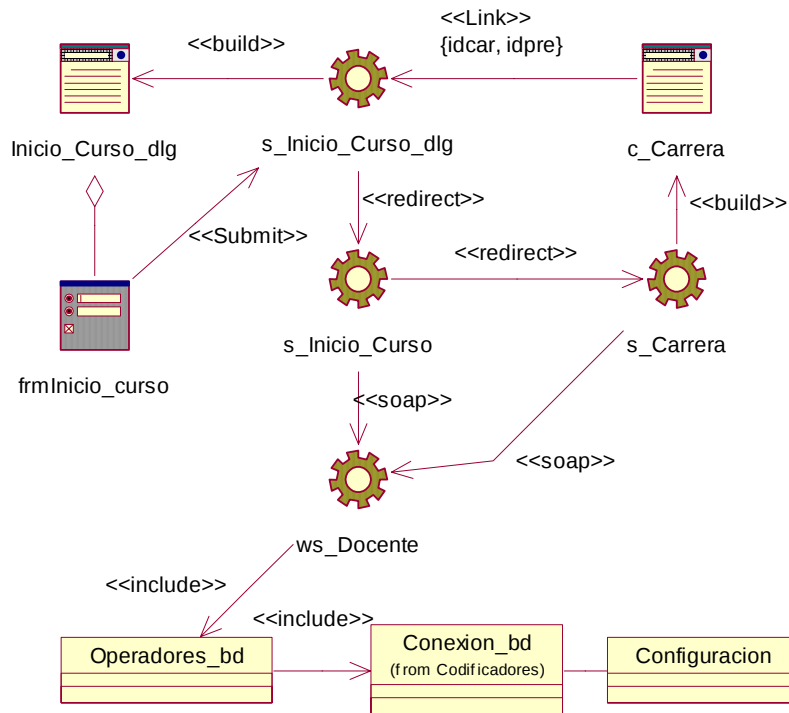
Anexo 4.4 Diagrama de Clases Web Registrar Matrícula.

Anexo 4.5 Diagrama de Clases Web Actualizar Matrícula.

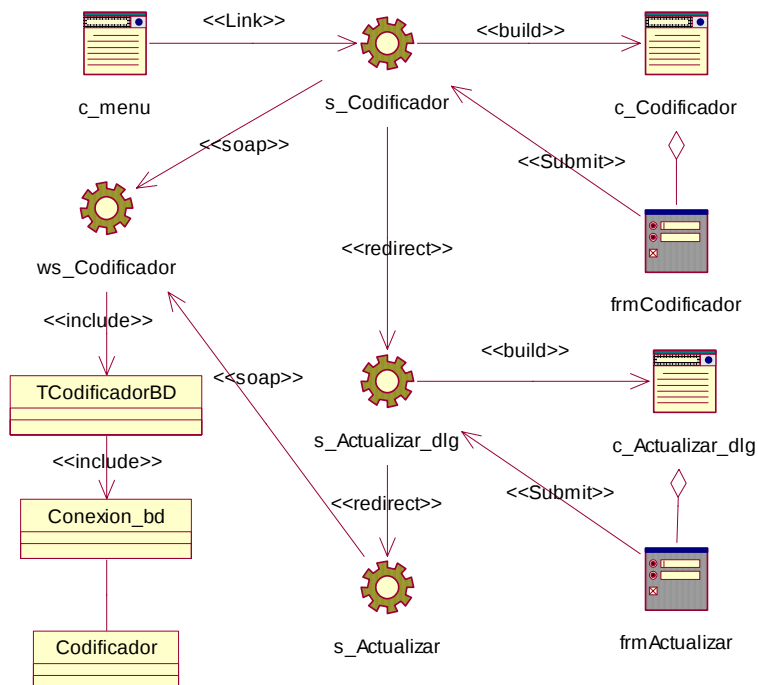
Anexo 4.6 Diagrama de Clases Web Conformar Listados de Grupos

Anexo 4.7 Diagrama de Clases Web Re-Matricular Estudiantes.

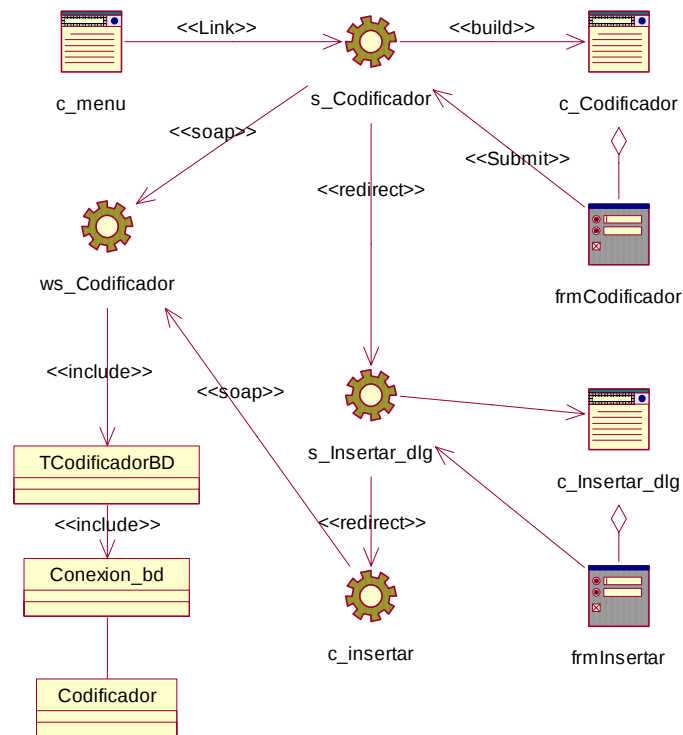
Anexo 4.8 Diagrama de Clases Web Registrar Datos Licencia.**Anexo 4.9 Diagrama de Clases Web Iniciar Curso Escolar**



Anexo 4.10 Diagrama de Clases Web Gestionar Codificador

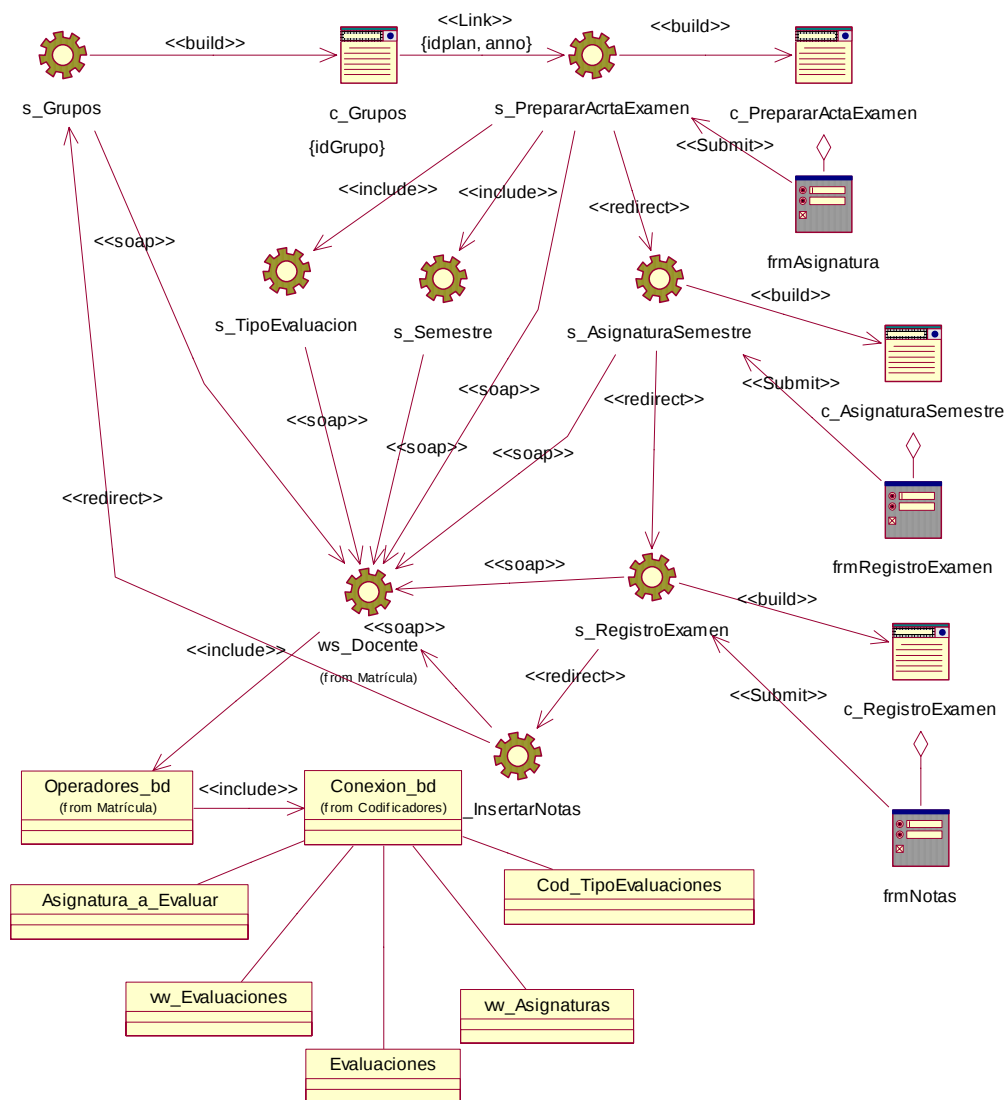


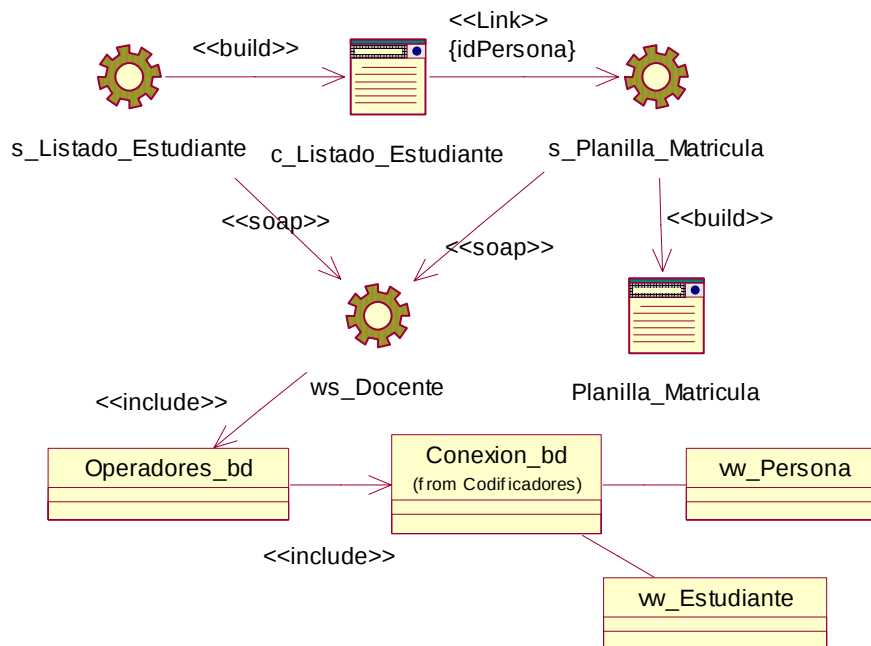
Actualizar Codificador

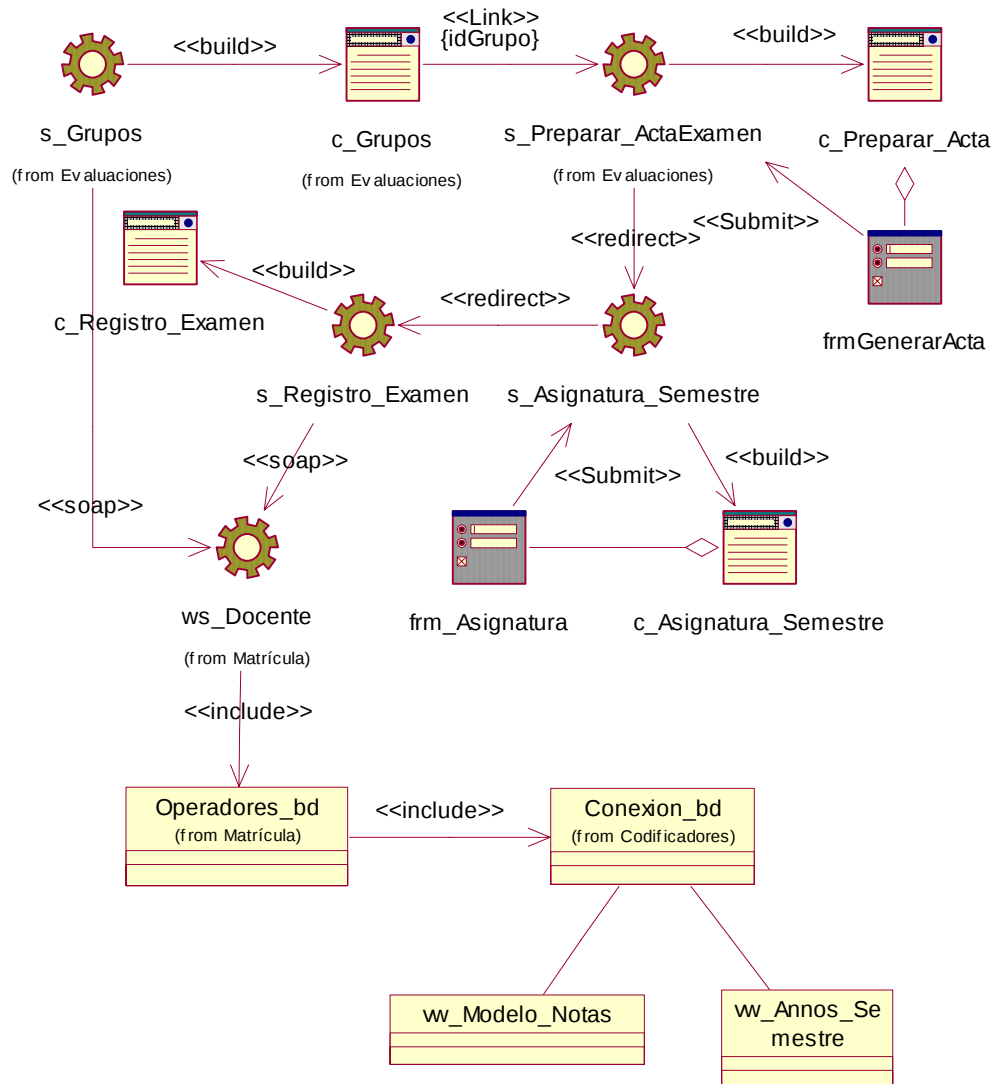


Insertar Codificador

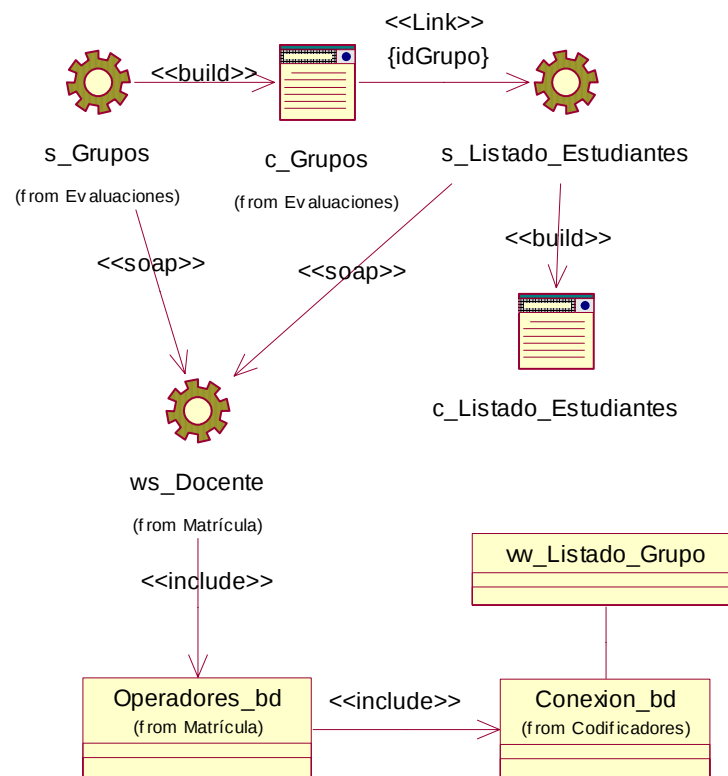
Anexo 4.11 Diagrama de Clases Web Conformar Actas de Asignaturas

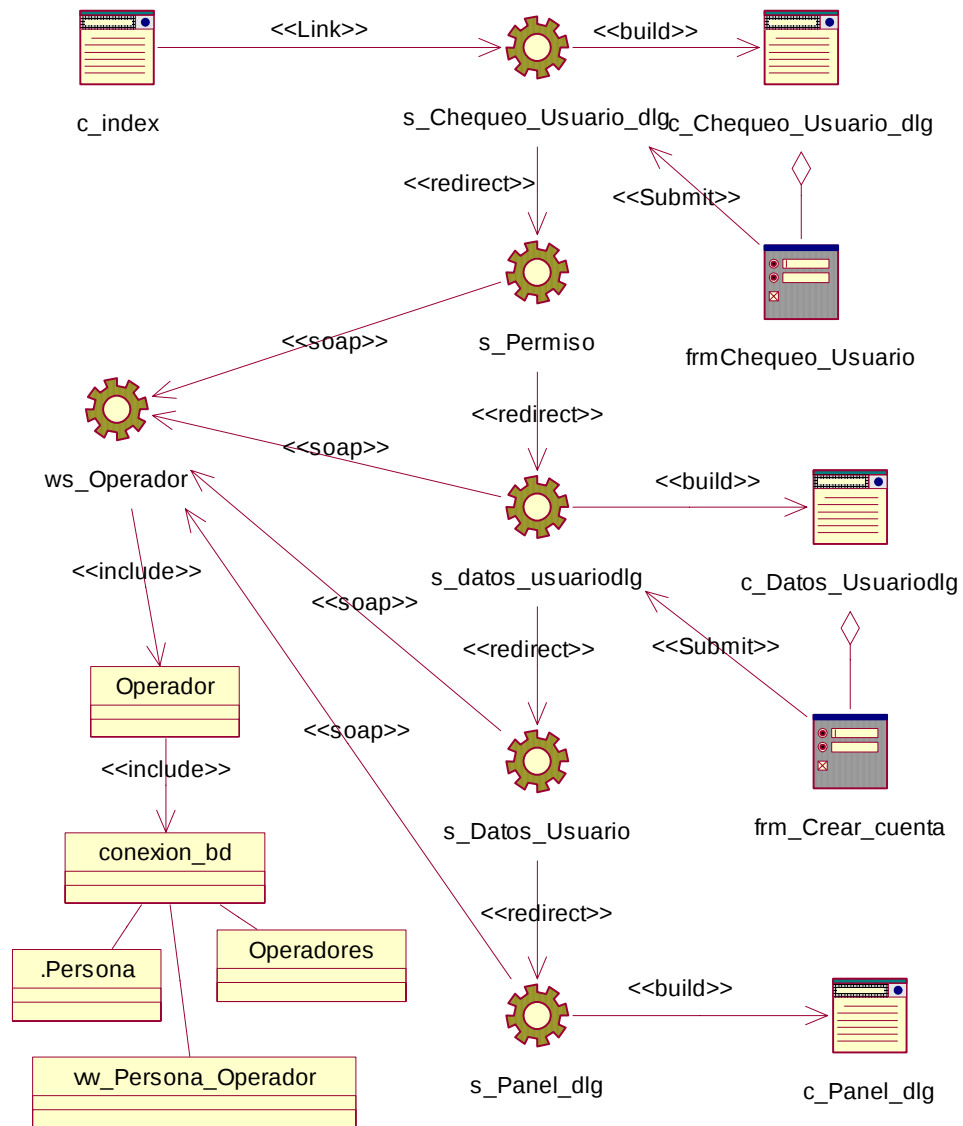


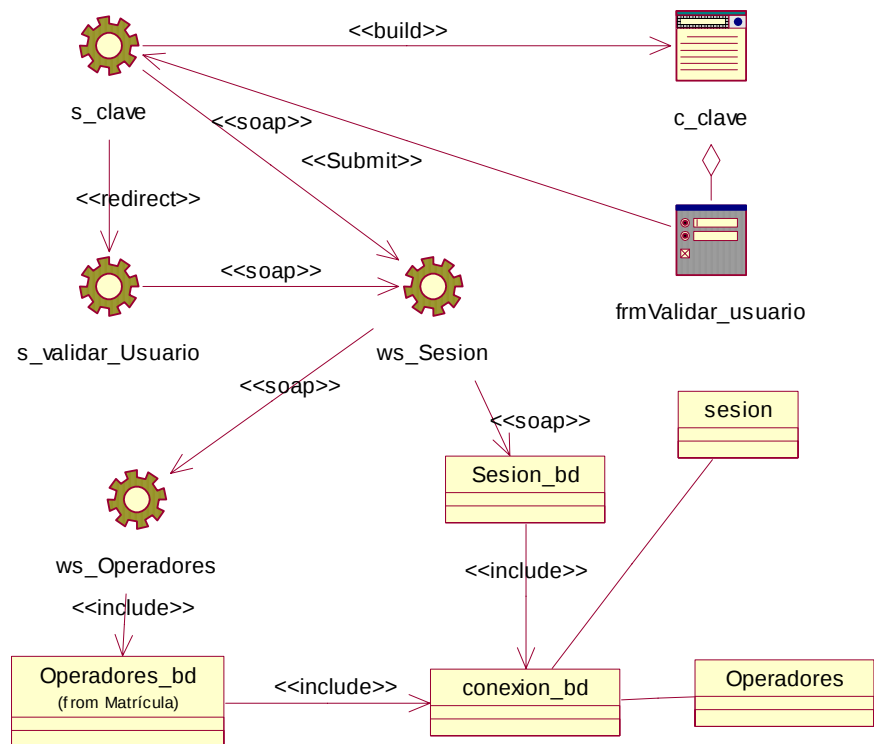
Anexo 4.12 Diagrama de Clases Web Visualizar Planilla de Matrícula

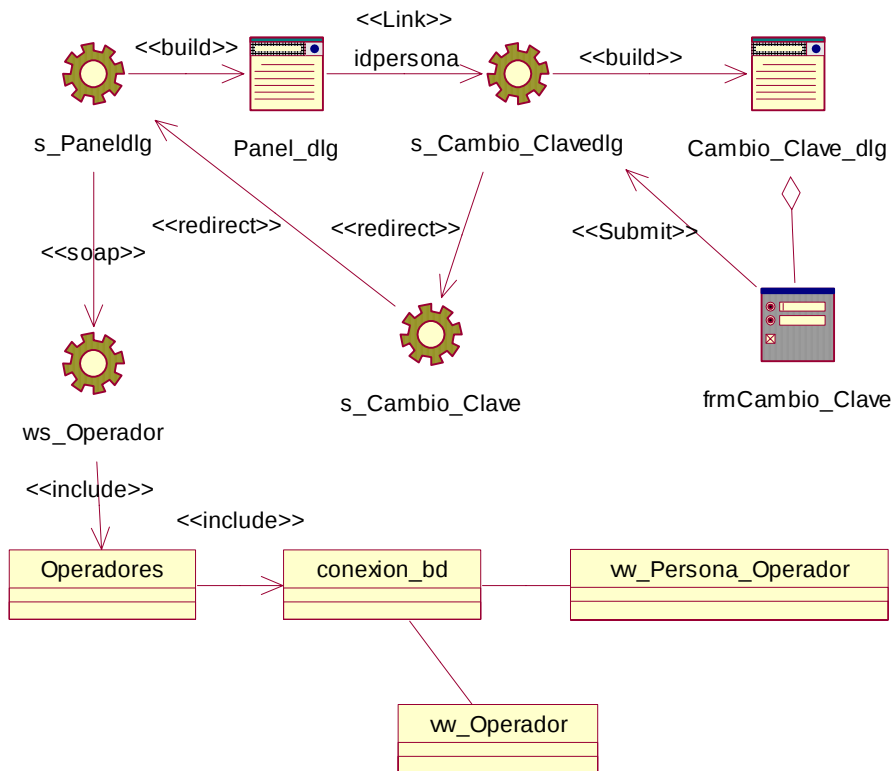
Anexo 4.13 Diagrama de Clases Web Visualizar Actas de Asignaturas

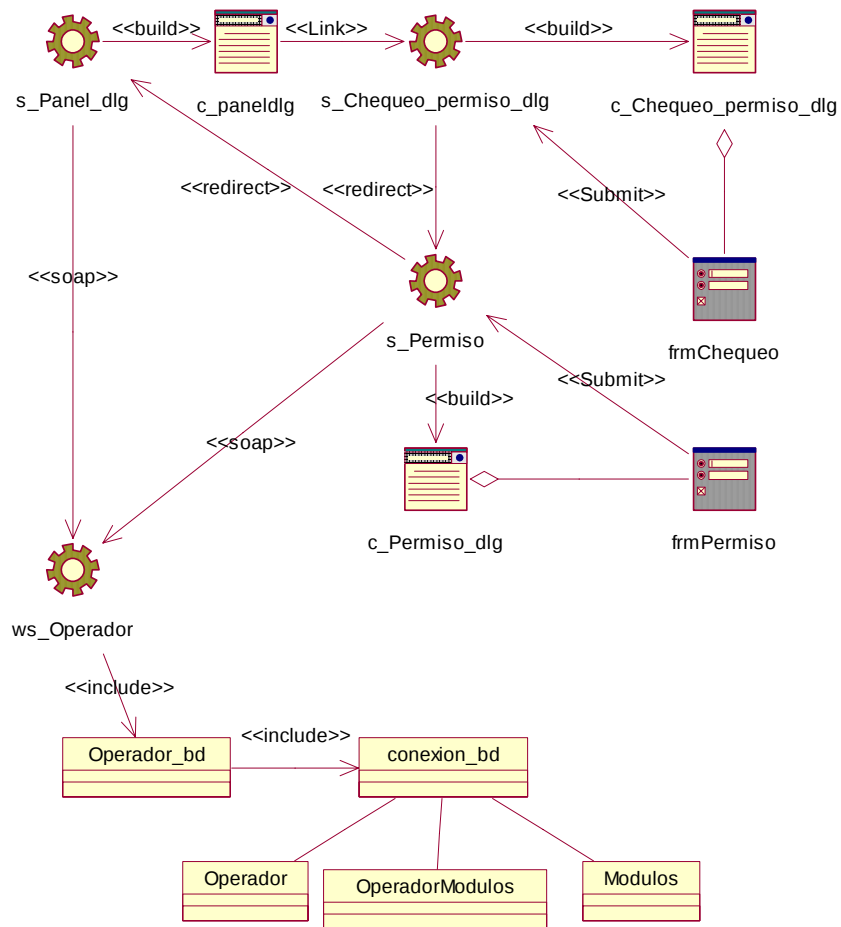
Anexo 4.14 Diagrama de Clases Web Visualizar Listados de Estudiantes por Grupos

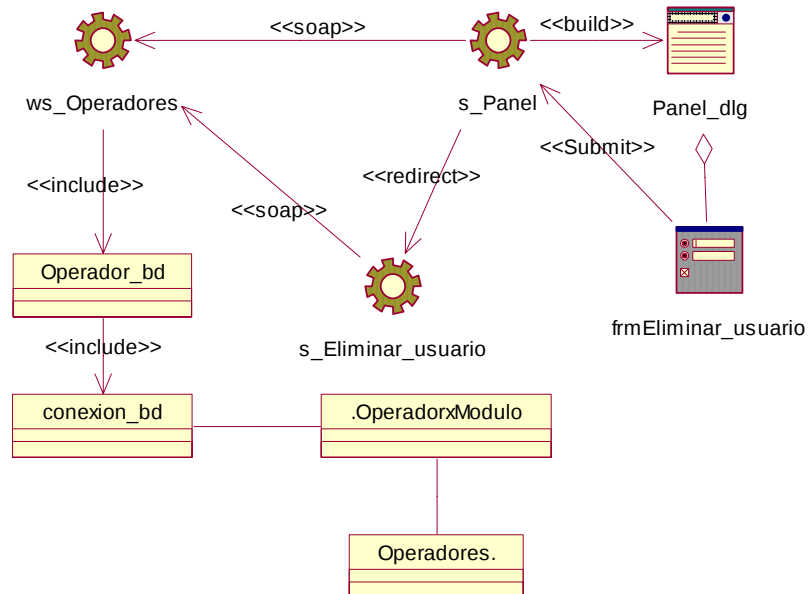


Anexo 4.15 Diagrama de Clases Web Crear Usuario.

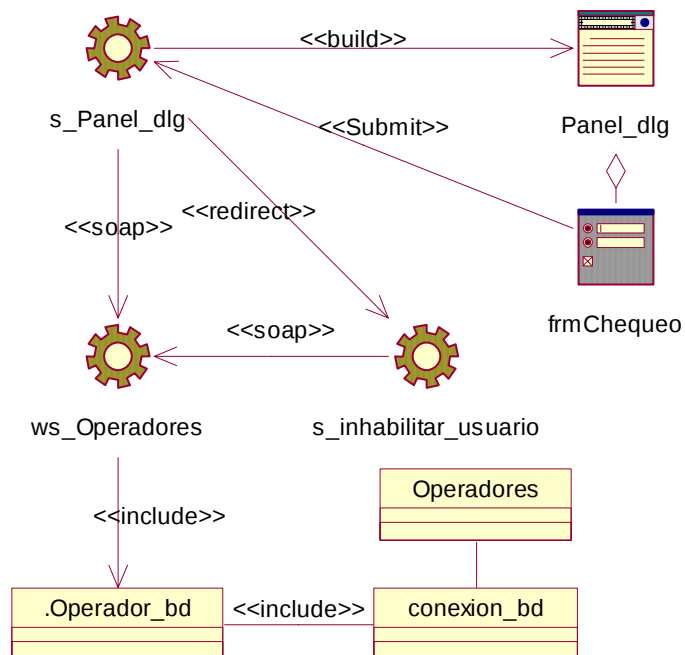
Anexo 4.16 Diagrama de Clases Web Iniciar Sesión**Anexo 4.17 Diagrama de Clases Web Cambiar Contraseña.**

**Anexo 4.18 Diagrama de Clases Web Dar Permisos Usuarios**

**Anexo 4.19 Diagrama de Clases Web Eliminar Usuario**



Anexo 4.20 Diagrama de Clases Web Habilitar (Inhabilitar) Usuario



Anexo 4.21 Consulta a la EBSCO

Lista de carpetas

Elementos de la lista de resultados (2) ▶ Imágenes (0) ▶ Vídeos (0) Vínculos persistentes a la búsqueda (0) ▶ Búsquedas guardadas (0) ▶ Alertas de búsqueda (0) ▶ Alertas de publicación (0) ▶ Páginas Web (0)	Elementos de la lista de resultados	
	1 - 2 de 2 Páginas: 1  Imprimir  Enviar  Guardar en disco Eliminar todos	
	1. SRW/U with OAI: Expected and Unexpected Synergies . By: Sanderson, Robert; Young, Jeffrey; LeVan, Ralph. D-Lib Magazine, Feb2005, Vol. 11 Issue 2, p1-1, 1p; DOI: 10.1045/february2005-sanderson; (AN 16556763)	
	2. Potential of Webservices to enable smart business networks . By: van Hillegersberg, Jos; Boeke, Ruurd; van den Heuvel, Willem-Jan. Journal of Information Technology (Palgrave Macmillan), Dec2004, Vol. 19 Issue 4, p281-287, 7p, 6 diagrams; DOI: 10.1057/palgrave.jit.200027; (AN 15908178)	
	1 - 2 de 2 Páginas: 1	

