

Universidad de las Ciencias Informáticas

"Facultad 4"



Título: "Portal del Departamento Especialidad"

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

Autor(es): "Jacsel Agramonte Osorio y Edgar Pirez Nash"

Tutor(es): "Ailia Morejón Espinosa"

Consultante: "Karina Panucia Tillán"

"Julio 2007"

...en Cuba nunca nadie ha hecho tanto en tan poco tiempo...

Fidel Castro Ruz.

DECLARACIÓN DE AUTORÍA

Declaramos ser autores de la presente tesis y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmo la presente a los 3 días del mes de julio del año 2007.

"Jacsel Agramonte Osorio"

Firma del Autor.

"Ing. Ailia Morejón Espinosa"

Firma del Tutor.

"Edgar Pirez Nash"

Firma del Autor.

AGRADECIMIENTOS

A Karina Panucia por su atención y ayuda.

A nuestra tutora Ailia Morejon por su disposición, su ayuda y confianza.

A la máxima dirección de nuestro país, de esta universidad y a nuestro inigualable Comandante en Jefe por la confianza que deposito en nosotros cuando tuvo la maravillosa y visionaria idea de la creación de este proyecto, del cual hoy nos enorgullecemos de ser los primeros graduados.

Jacsel

En la vida todos tenemos grandes metas, unas más difíciles de alcanzar que otras, pero para todas necesitamos del apoyo de alguien. Para llegar a alcanzar esta gran meta que hoy estoy a punto de tocar con mis manos he necesitado del apoyo de muchos entre ellos mis padres que me han brindado lo mejor de ellos y me han apoyado siempre, mi hermana, mis abuelos que los adoro y me atienden incondicionalmente, mis amigos que han estado junto a mí en los momentos más difíciles, mis compañeros del laboratorio con los que he pasado largas noches de trabajo y todo el que de una forma u otra ha contribuido al logro de este propósito, que le llegue mis más sinceros agradecimientos.

Edgar

A todos aquellos compañeros de aula y del deporte que de una forma u otra me ayudaron, compartieron conmigo y me brindaron su amistad la cual perdurará aunque ya no estemos juntos, en especial a Adolfo, Lannie, Luis Orlando y otros muchos quienes me ayudaron a enriquecer mis conocimientos y mis experiencias para la vida. Y por ultimo a mi profesora Hilda Loisa Garrido Pelegrin por su apoyo, incondicionalidad y ayuda, simplemente mi mamá en la UCI.

DEDICATORIA

El empeño, la dedicación y consagración que he puesto en la realización de este trabajo y el trabajo en si, todo esta dedicado a mis padres y a mi familia, quienes han soñado siempre con que yo llegase a alcanzar la cúspide de esta gran montaña que constituye el ser universitario, me reconforta hoy poderles dar esa satisfacción y llenarlos de gratitud.

Jacsel

Todos estos años de estudio y sacrificio, el desarrollo de este trabajo de diploma es para mi hermano y mi mamá por ser la que me condujo a que yo estuviera y me desarrollara en este proyecto, por creer en mi, por su apoyo, su sacrificio y dedicación. A mi papá por su ayuda. Gracias a ellos y a todas mis amistades que hicieron posible con su apoyo que yo llegara a convertirme en un ingeniero de estos tiempos y por último le dedico también este trabajo a todos los que como yo ansían obtener esta meta.

Edgar

RESUMEN

Desde los comienzos de la UCI se hizo necesaria la organización de la universidad en facultades las cuales fueron 6 al comienzo, se ampliaron a 10 y continuaron creciendo a medida que fue aumentando la matrícula de los nuevos estudiantes que ingresaban en la universidad. En principio todos los departamentos de las asignaturas estaba centralizado debido a que la matrícula de la universidad así lo permitía, después con el objetivo de lograr una mejor organización, planificación y funcionamiento se extendieron a cada facultad para así mejorar la coordinación de las asignaturas por las facultades ya que el aumento de la matrícula de alumnos y profesores así lo requería.

Se organizaron departamentos por asignaturas, agrupados según las especialidades de las mismas por facultades teniendo un departamento central el cual sería el vínculo para la sincronización a la hora de realizar ciertas actividades que dependen de un factor único que las une como es el caso de las pruebas de nivel a través de una plataforma central entre otras.

Los departamentos por facultades son los encargados de organizar todas las pruebas y actividades de las asignaturas de un semestre para cada uno de los años correspondientes.

En estos momentos no existe un sistema que facilite el desempeño del trabajo en la facultad y optimice su funcionamiento, en el momento de publicar, confeccionar, difundir documentos, informaciones, noticias entre otras; por lo que se concibe la creación una aplicación, después de realizar el análisis, diseño e implementación basado en la tecnología web que solucione las problemáticas de este departamento y eleve la calidad del desarrollo del trabajo en el mismo.

TABLA DE CONTENIDOS

AGRADECIMIENTOS	I
DEDICATORIA.....	II
RESUMEN.....	III
INTRODUCCIÓN.....	8
Problema Científico.....	8
Objeto de estudio.....	9
Campo de acción.....	9
Hipótesis.....	9
Objetivo general.....	10
Objetivos específicos.....	10
Tareas desarrolladas para cumplir los objetivos.....	10
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA	11
1.1- La Gestión de la información como base de la creación del sistema.....	11
1.2- Aplicaciones WEB	12
1.3- Arquitectura propuesta.....	14
1.3.1- Las tres capas definidas son.....	15
1.4- Análisis de los lenguajes de programación Web.....	16
1.4.1- Posibles a tener en consideración.....	17
1.5- Introducción al .NET Framework.....	24
1.6- Vista general de ASP.NET 2.0	25
1.7- Argumentación de la tecnología web escogida.....	27
1.3- Argumentación del Gestor de Base de Datos escogido.....	29

1.9-	Argumentación de la metodología de desarrollo de software escogida.....	31
1.10-	Otros programas utilizados en la propuesta realizada.....	31
1.11-	Sistemas Actuales (Estado del Arte).	32
1.11.1-	Internacionales.	32
1.11.2-	Nacionales.....	33
1.12-	Conclusiones.....	34
CAPÍTULO 2:	CARACTERÍSTICAS DEL SISTEMA	35
2.3-	Problema y Situación Problémica.....	35
2.3.1-	Flujo actual de los procesos involucrados en el campo de acción.	35
2.3.2-	Análisis crítico de cómo se ejecutan actualmente esos procesos.	36
2.3.3-	Descripción del sistema propuesto.....	38
2.3.4-	Modelo Conceptual (M. de Dominio).	41
2.3.5-	Glosario de Términos.	41
2.4-	Especificación de los requisitos del software.	43
2.4.1-	Requerimientos Funcionales.	43
2.4.2-	Requerimientos no Funcionales.	46
2.5-	Definición de los casos de uso.	49
2.5.1-	Definición de los actores.	49
2.5.2-	Listado de casos de uso del sistema.....	50
2.5.3-	Diagrama de casos de uso del sistema.....	56
CAPÍTULO 3:	ANÁLISIS Y DISEÑO DEL SISTEMA.....	57
3.1-	Análisis.....	57
3.1.1-	Diagrama de clases del Análisis.....	57
3.2-	Diseño.	58
3.2.1-	Diagramas de interacción.....	58
3.2.2-	Diagrama de clases web.	59
3.2.3-	Descripción de las clases.....	59

3.3-	Diseño de la Base de Datos.....	61
3.3.1-	Diagrama Entidad Relacional de la Base de Datos.....	61
3.3.2-	Descripción de las tablas.....	62
CAPÍTULO 4: IMPLEMENTACIÓN Y PRUEBA	66	
IMPLEMENTACION.....	66	
4.1-	Diagrama de Despliegue.....	66
4.2-	Diagrama de Componentes.....	66
4.2.1-	Paquete de Presentación.....	67
4.2.2-	Paquete de Entidades y Lógica del Negocio.....	68
4.2.3-	Paquete de Acceso a Datos.....	69
PRUEBA.....	70	
	Nombre del Caso de Uso: Autenticar en el Sistema.....	70
	Nombre del Caso de Uso: Realizar Reporte, sección Buscar Información de Profesor.....	72
	Nombre del Caso de Uso. Confeccionar Horario, sección Mostrar Horario de un grupo.....	72
	Nombre del Caso de Uso. Confeccionar Horario, sección Mostrar Horario de Aula.....	72
	Nombre del Caso de Uso. Confeccionar Horario, sección Insertar Horario Docente.....	73
	Nombre del Caso de Uso. Insertar Plan de Trabajo.....	74
CONCLUSIONES GENERALES.....	75	
RECOMENDACIONES.....	76	
BIBLIOGRAFÍA.....	77	
ANEXOS.....	79	
ANEXOS I. Diagrama de clases del Análisis.....	79	
	Figura 1. DCA. CUS Publicar Plan de Actividades.....	79
	Figura 2. DCA. CUS Administrar.....	80

Figura 3. DCA. CUS Confeccionar Horario.....	81
Figura 4. DCA. CUS Publicar Información.....	82
Figura 5. DCA. CUS Publicar Noticia.....	82
Figura 6. DCA. CUS Publicar Plan de Trabajo.	83
Figura 7. DCA. CUS Publicar Plan del Departamento.....	83
Figura 8. DCA. CUS. Realizar Reporte.....	84
ANEXOS II Diagramas de Secuencia del Diseño.....	85
Figura 9. DSD. CUS Mostrar Horario.....	85
Figura 10. DSD. CUS Mostrar Información de Alumno Ayudante.	86
Figura 11. DSD. CUS Mostar Reporte.....	87
ANEXOS II Diagramas de Clases del Diseño Web.....	88
Figura 12. DSD. CUS Publicar Noticia.....	88
Figura 13. DCD. CUS Mostar Horario del Profesor.	89
Figura 14. DCD. CUS Mostrar Horario de Grupo.	89
Figura 15. DCD. CUS. Mostar Horario de Local.	90
Figura 16. DCD. CUS. Mostrar Información de Alumno Ayudante.	90
Figura 17. DCD. CUS. Mostrar Información del Profesor.	91
Figura 18. DCD. CUS. Mostrar Reporte.	92
Figura 19. DCD. CUS. Publicar Información.....	93
Figura 20. DCD. CUS. Publicar Noticia.	93
ANEXOS III Descripción de las Clases del Diseño Web.	94
GLOSARIO.....	105

INTRODUCCIÓN

Actualmente, en la facultad 4 de la UCI (Universidad de las Ciencias Informáticas), se hace difícil gestionar las distintas tareas del Departamento de Ciencias Informáticas de manera eficiente. El Jefe de Departamentos de la facultad, entre otras responsabilidades mensuales debe, para cada Profesor, controlar su Plan de Trabajo y darle una evaluación, controlar las actividades realizadas, controlar sus responsabilidades como guía de grupo, como militante y en su proyecto, sus incidencias, su categoría docente, su superación, además publicar un Plan de Trabajo del Departamento. Los profesores deben, para cada AA (Alumno Ayudante), chequear su plan de trabajo, sus actividades y darles una evaluación.

Para el desarrollo de los procesos que tienen lugar en el departamento en cuestión, todos los modelos son elaborados prácticamente de forma manual y con la ayuda de tablas en Microsoft Excel, Word y los datos de las entidades involucradas en los mismos también (Plan de Trabajo, Evaluaciones, Actividades etc.), lo cual conduce a grandes esfuerzos y en consecuencia a demoras en las entregas de los documentos debido a la información manejada en los mismos de una forma no muy cómoda. Además muchos de estos parten de la información de otros, lo que significa reescribir los datos en cada caso, y también es necesario organizar esta información dependiendo de las necesidades, para la emisión de reportes.

Debido a que no existe una herramienta que facilite las actividades realizadas en del departamento y a todas las dificultades plantea anteriormente es que se hace necesario investigar sobre una posible solución que sea capaz de resolver y automatizar toda la dinámica de esta organización a través de la implementación de un sistema informático que permita elevar la eficiencia y calidad del trabajo en el departamento.

Problema Científico.

La no existencia de un sistema automatizado que permita aprovechar las ventajas de las tecnologías de la informática y las comunicaciones para la captación y transformación de la información manejada en un Departamento Docente.

Objeto de estudio.

Con vista a dar solución al problema antes planteado se define que el objeto de estudio es la organización y el acceso a la información relacionada con la actividad de un Departamento Docente.

Campo de acción.

Sistema automatizado para la gestión de la información de un Departamento.

Hipótesis.

Se debe realizar una investigación de los procesos involucrados en la entidad así como su funcionamiento con el objetivo de logra implementar un software que unifique y estandarice los datos de las principales entidades que comprende dicho negocio de manera que permita manipular eficazmente la información del departamento y los reportes que se requieren del mismo, de forma que se cumplan los objetivos trazados. Debido a las características de las que se quiere prescindir como son; que se quiere disponer, procesar los datos y la información en cualquier momento y desde cualquier puesto en la red es que se piensa construir una aplicación web que se adapta muy bien para implementar la lógica de este negocio por la abstracción que brinda en cuanto a componentes adicionales y del lugar físico en el que se encuentre el cliente con respecto al sistema.

Objetivo general.

Lograr la automatización del proceso de gestión de información un departamento mediante el desarrollo de un sistema informático que permita organizar y mostrar adecuadamente dicha información.

Objetivos específicos.

- ❖ Controlar y disponer de los planes de trabajo de los profesores, alumnos ayudantes y del propio Departamento, para que puedan ser consultados en cualquier momento que se necesite y desde cualquier lugar en la red.
- ❖ Mantener un control sobre las actividades realizadas por los integrantes del departamento.
- ❖ Ofrecer una solución que permita a los integrantes del departamento mantenerse informado a través del acceso rápido y eficiente a la información, datos, reuniones, etc del Dpto.
- ❖ Visualizar reportes con la información sobre el estado del departamento, de los profesores, alumnos ayudantes, asignaturas y horarios.
- ❖ Plantear una solución general que en futuras versiones pueda ser aplicada a cualquier departamento que presente una problemática similar a la del Departamento de Ciencias Informáticas de la Facultad 4.

Tareas desarrolladas para cumplir los objetivos.

Con el objetivo de entender la lógica del negocio se realizaron un grupo de acciones:

- ❖ Entrevistas con los clientes y usuarios finales.
- ❖ Cita de documentos en vistas de solidificar la solución ofrecida.
- ❖ Intercambio constante con el jefe y algunos integrantes del departamento.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

En este capítulo se materializarán aspectos que serán de ayuda y guía para que los lectores comprendan las definiciones actuales existentes y las que se establecen en el trabajo. Así como el porque de la selección de la tecnología con que se desarrollará el sistema, basado en sus cualidades y prestaciones para la gestión de la información, cuya necesidad generó la realización de este proyecto que puede ser utilizado como punto de partida para el desarrollo de nuevos software en el campo tratado en nuestro trabajo.

1.1- La Gestión de la información como base de la creación del sistema.

Desde el mismo momento en que el hombre tomó conciencia sintió la necesidad de gestionar la información, sintió la necesidad de almacenarla, organizarla, centralizarla de alguna manera para materializar su existencia y reflejar su paso por las distintas épocas que en las que fue dejando su huella con los diferentes descubrimientos que iba haciendo.

La información a medida que pasa el tiempo a ido aumentando porque cada día son mas los descubrimientos, innovaciones, la creatividad del hombre nunca tiene un fin y siempre tiene un gran reto por delante; hoy día los volúmenes son mucho mas grandes cada día por lo que se hace cada vez mas inevitable una gestión eficiente de la misma con el objetivo de optimizar funciones, tareas, procesos en fin de lograr una mayor calidad en su procesamiento.

En los tiempos actuales y con todo el desarrollo del mundo digital se ha hecho más fácil la manipulación de toda la información ya sea a través de sistemas basados en desarrollo web o en aplicaciones de escritorio, situándose en el caso específico al que se pretende dar solución fue escogido un sistema web debido al sin número de prestaciones que brinda, actualmente la tecnología mas usada junto al correo electrónico y la mensajería instantánea en Internet con un crecimiento a un ritmo vertiginoso.

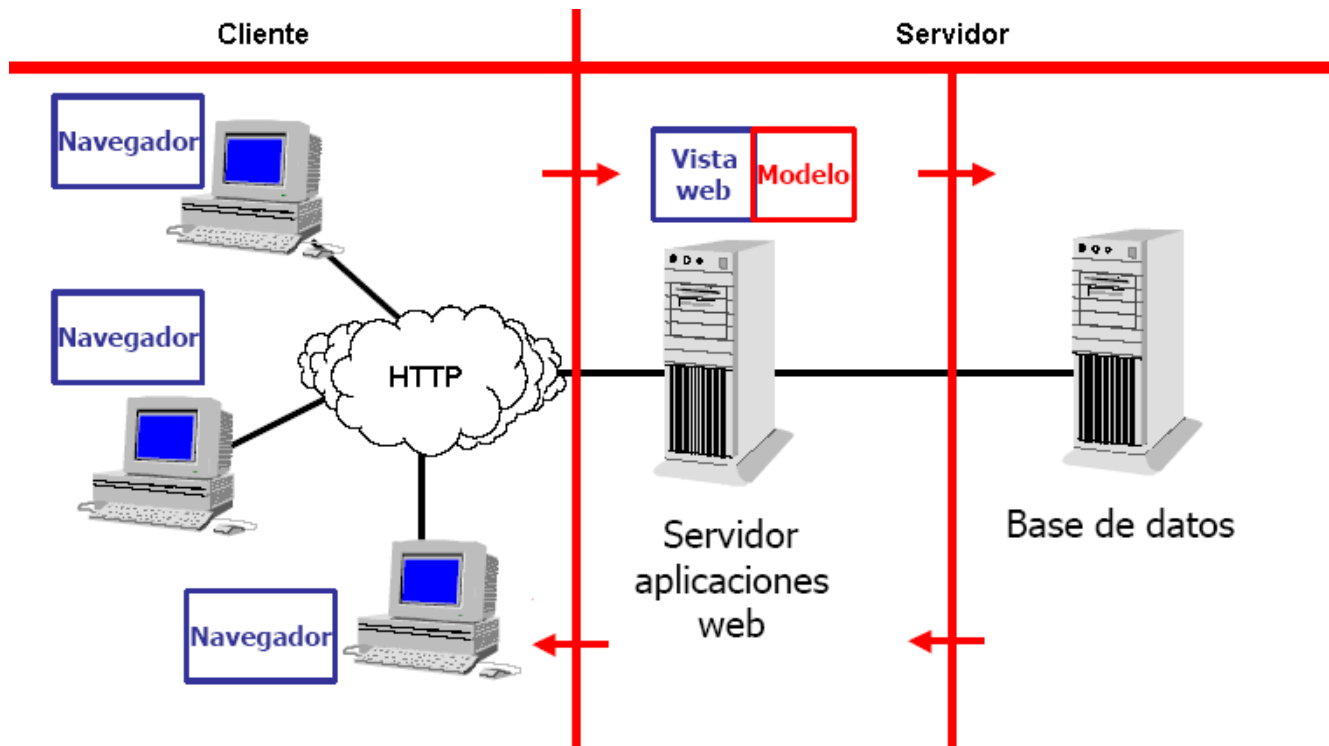
La Web, se puede decir que son páginas confeccionadas en un lenguaje especial llamado HTML, en inglés (Hyper Text Markup Language), que le permite a los desarrolladores web visualizar múltiples contenidos como son textos, imágenes, videos, documentos, entre otros. Las páginas se

organizan para formar parte de un sitio, estas pueden tener cualquier cantidad de información sobre cualquier tema. La gran mayoría de las empresas de estos tiempos usan este servicio para extender su campo de acción, para expandir su mercado, y a través de la web brindan innumerables servicios, haciendo los negocios abstrayéndose de la posición geográfica de sus clientes.

Por lo general el protocolo mas usado es el HTML aunque existen otros dependiendo del tipo de transferencia que se vaya a realizar, como son el HTTPS (Secure HTML) para transferencias bancarias, servicios de correo vía web, FTP (File Transfer Protocol) para la transferencia de archivos, entre otros.

1.2- Aplicaciones WEB

La arquitectura cliente/servidor fue una de las razones por la cual se eligió hacer un sistema basado en la web, un usuario cualquier usando un navegador escribiendo básicamente la URL (Uniform Resource Locator) de la página puede hacer una petición a través del protocolo HTTP (Hypertext Transfer Protocol) a un servidor web, este atiende la petición, procesa la misma y envía una respuesta al usuario a través del navegador que muestra la respuesta en formato HTML. En la siguiente figura # se muestra una imagen con al arquitectura web básica.



El surgimiento de Internet ha impulsado inmensamente esta tecnología, el surgimiento diario de miles de sitios, los múltiples accesos de millones de personas a los mismos sin importar su posición geográfica y las cada vez mayores solicitudes e intercambio de datos e información constituyen para los desarrolladores web un desafío y un reto, esto muy ligado a la constante evolución de las tecnologías que requiere aplicaciones mas rápidas, robustas, confiables que permitan hacer mas eficiente y seguro el acceso a la web mundial.

En el caso que se centra este trabajo, les permitirá a los usuarios el acceso a las informaciones del departamento, donde se encontrarán publicados planes de trabajo, horarios, noticias entre otros que se encontraran disponibles en la web con fácil acceso a través de la red local.

Las aplicaciones de los sistemas basados en la web son muchos desde la publicación de videos como Youtube y GoogleVideo hasta los populares servicios de correo electrónico Yahoo Mail, Hotmail y Gmail de Google, pero no solo son estas las únicas funciones que tienen también son utilizados en le manejo de la lógica de negocios y en la optimización de las funciones de una empresa, organizando, gestionando, almacenando, etc esto unidos a otras herramientas como lo son los gestores de base de datos, sistemas gestores de conocimiento, entre muchas otras.

El sistema web que se prevé implementar está orientado a facilitar y perfeccionar el trabajo del Departamento de Ciencias Informáticas de La Facultad 4 apoyado en un gestor de base de datos para almacenar toda la información que se quiere publicar y consultar.

Debido a las altas prestaciones que brindan las aplicaciones web actualmente, se ha generalizado su uso, generando variedades de propuestas ya que el usuario o cliente solo debe disponer de un navegador web como interprete del código HTML y los recursos incorporados, evitando así la instalación de componentes de software adicionales y posibles configuraciones que pueda dificultar el uso de los mismos.

Las páginas web permiten un mayor aprovechamiento de las potencialidades de Internet como son sus estándares que hacen que los desarrolladores tengan que construir las mismas siguiendo un patrón como modelo para implementarlas y hacen que sean universales por lo que cualquier persona sin importar el lugar del mundo donde se encuentre puede acceder a ellas tan solo con un navegador web que por lo general no requieren de un conocimiento informático desarrollado para usarlas.

Para disponer de las mismas solo se necesita de una conexión de red para consultar la información contenida en ellas sin importar en el lugar donde se encuentre alojado el sitio que las agrupe, dependiendo de los permisos que se tengan para restringir el tipo de acceso a la red. Por estas razones que han sido argumentadas y otras es que se ha llegado a la conclusión de que un sistema pasado en aplicaciones web es el más idóneo para automatizar el Departamento de Ciencias Informáticas de la Facultad 4.

1.3- Arquitectura propuesta.

Existen varias arquitecturas para desarrollar aplicaciones web pero de todas, la más usada debido a las altas prestaciones y ventajas que brinda para la implementación de sistemas web es la arquitectura de 3 capas, este constituye una generalización de la arquitectura cliente-servidor donde la carga se divide en tres partes con un reparto claro de funciones: una capa para la presentación, otra para el negocio y la otra para el acceso a datos, se propone organizar los elementos de la aplicación en componentes independientes en vistas de buscar una mejor eficacia en su administración. Una capa solamente tiene relación con la siguiente.

1.3.1- Las tres capas definidas son.

Presentación

En esta capa se diseña todo lo que constituye la interfaz gráfica y la interacción del usuario con el sistema.

Negocio

Contiene todas las subrutinas creadas con el propósito de regular alguna acción del usuario.

Acceso a Datos

En esta capa se programa todo lo que tiene que ver con el acceso a la base de datos. Esta capa queda encargada de tomar la información de la base de datos dada una petición de la capa de Reglas del Negocio, que a su vez es generada por la capa de presentación.

1.3.2- Aplicaciones del modelo tres capas.

Ventajas

- Implementación en paralelos de cada capa.
- Las aplicaciones son más sólidas debido al encapsulamiento, ya que oculta la complejidad de la implementación sin importa como son construidos los componentes de los módulos sino saber como interactuar con ellos, con el fin de que si alguno es modificado no afecte su relaciones con los demás componentes.
- Como los módulos son independientes uno de otro darle mantenimiento y soporte no implicaría tener que cambiar los demás.
- Una mejor flexibilidad, es posible en un futuro adicionarle otros módulos a la aplicación que contengan nuevas funcionalidades sin que se vean afectados los otros módulos.
- Una mayor escalabilidad, el sistema puede atender mas peticiones con el mismos rendimiento con tan solo adicionarle mas hardware.

Este modelo nos ofrece una abstracción del sistema de manera que contempla posibles crecimientos de la entidad en la que es montado, así como también es muy adaptable ya que permite ser adecuado a diferentes entornos sin importar el tamaño del mismo, los principales componentes de la aplicación serán programados por separado y unidos en tiempo de ejecución como plantea esta arquitectura, esto implica que si una de las capas es modificada no se vean

afectadas las demás ni el resultado final. Por lo que debido a la estabilidad y eficiencia que brinda este modelo es que fue escogido para aplicarlo en la implementación de nuestro sistema.

1.4- Análisis de los lenguajes de programación Web.

El mundo de la programación web es cada vez más exigente, debido a los crecientes volúmenes de información que se producen diariamente, es necesario crear aplicaciones que muestren más rápido la información solicitada por los usuarios, que sean más eficientes en el manejo de los datos, aprovechamiento de recursos y esto no se logra si no es con una buena gestión ya sea tanto del lado del cliente que del lado del servidor, evidenciándose una división en dos grupos de los lenguajes para la programación web según la arquitectura Cliente/Servidor, siendo estos grupos los lenguajes de programación del lado del cliente, los lenguajes de programación del lado del servidor y los lenguajes del lado cliente/servidor.

Así pues, podemos hablar de lenguajes de lado servidor que son aquellos lenguajes que son reconocidos, ejecutados e interpretados por el propio servidor y que se envían al cliente (navegador) en un formato comprensible para él, generalmente HTML. El navegador es una especie de aplicación capaz de interpretar las órdenes recibidas en forma de código HTML fundamentalmente y convertirlas en las páginas que son el resultado de dicha orden.

Cuando se accede a un enlace que solicita una página que contiene un script en un lenguaje comprensible únicamente por el servidor, lo que ocurre en realidad es que dicho script es ejecutado por el servidor y el resultado de esa ejecución da lugar a la generación de un archivo HTML que es enviado al cliente.

Por otro lado, los lenguajes de lado cliente entre los cuales no sólo se encuentra el HTML sino también el Java y el JavaScript son simplemente incluidos en el código HTML, son aquellos que pueden ser directamente ejecutados por el navegador y no necesitan un pretratamiento.

Cada uno de estos tipos de lenguajes de programación web tiene por supuesto sus ventajas y sus desventajas. Como son por ejemplo, un lenguaje de lado cliente es totalmente independiente del servidor, lo cual permite que la página pueda ser albergada en cualquier sitio sin necesidad de pagar más, ya que por regla general los servidores que aceptan páginas con scripts de lado servidor son en su mayoría de pago o sus prestaciones son muy limitadas. Inversamente, un lenguaje de lado servidor es independiente del cliente por lo que es mucho menos rígido respecto

al cambio de un navegador a otro o respecto a las versiones del mismo. Por otra parte, los scripts son almacenados en el servidor quien los ejecuta y traduce a HTML por lo que permanecen ocultos para el cliente. Este hecho puede resultar una forma legítima de proteger el trabajo intelectual realizado por un individuo.

Como se pudo apreciar no solo la selección de la arquitectura es importante sino que también lo es la selección de los lenguajes a utilizar para implementar cada capa ya sea del lado del cliente o del lado del servidor, por el significado que tiene en el resultado total. Se hará un estudio de los posibles lenguajes a utilizar para seleccionar el más apropiado por capas según el problema en cuestión, que abordamos en nuestro trabajo.

1.4.1- Posibles a tener en consideración.

1.4.1.1- Del lado del Servidor.

La variedad de lenguajes con características a tener en cuenta en estos momentos es buena debido a la fortaleza y eficiencia con que cuentan muchas de las plataformas de programación que se consideran de alta calidad en estos momentos entre estas podemos mencionar a:

JSP (Java Server Pages).

Es una invención de la SUN que provee de un lenguaje de scripting en el lado del servidor que se comunica con clases Java, objetos RMI, CORBA, etc. La metodología de trabajo esperada es la misma que con Visual Basic con la diferencia de que esta vez se trata de una plataforma mucho más abierta.

El código JSP se puede poner dentro de las páginas HTML, o se puede precompilar en Servlets (programas en Java también pueden ser compilados en Servlets). Un servlet es una subclase de servlet que tiene métodos para atender requerimientos. El servidor pasa las variables hacia y desde el ambiente de ejecución del Servlet.

Para programar en JSP se requiere conocer Java, ser metódico y ordenado. El lenguaje no relaja las condiciones de tipado que le son propias.

ColdFusion

Es un lenguaje 100% de scripting en que la idea es lograr resultados con pocas líneas de código. Explota el hecho de que en la práctica muchos programadores lo que hacen es simplemente programar y no quieren tener que ver con Objetos, Clases ni nada de alto vuelo sino sólo con hacer consultas a la base de datos e imprimir el resultado.

Eso no quiere decir que el lenguaje no sea muy potente. Además cuenta con un IDE y soporte comercial. ColdFusion corre como un add-on a los servidores, o como un CGI aparte. Es similar en concepto y expresividad a W3-Msql.

ColdFusion es la mejor manera de sacar algo bueno de una mala situación, esto es, de tener que desarrollar una aplicación para Web bajo Windows 2000.

PHP

Es un pre-procesador de páginas HTML que está motivado y dirigido expresamente a construir páginas, como ColdFusion. Un ejemplo de ello es que los argumentos CGI son importados directamente al espacio de nombres global de cualquier programa PHP: PHP puede correr como un CGI o como un plug-in del servidor web.

PHP ha crecido en soporte y en funciones y ahora viene "de fábrica" con varias decenas de bibliotecas para funciones matemáticas, de bases de datos, etc. En este momento se encuentra en una fase de consolidación tras unos cuantos años de éxito, y la fase expansiva ha sido más bien dejada atrás para madurar en aspectos más relacionados con la integración de sus partes entre sí.

Perl

Permite resolver los problemas fáciles fácilmente, y resolver también problemas difíciles. Es rápido hacer una pequeña aplicación Web.

Las ventajas principales son el que se trata de un lenguaje muy maduro, que lleva mucho tiempo funcionando, y que tiene cientos de bibliotecas operativas y listas para ser usadas.

En Perl es más fácil que en otros lenguajes hacer algo inentendible y oscuro. Por otra parte, si uno tiene paciencia puede encontrar el 90% del trabajo que tenía que hacer ya hecho por otra persona, y el código que tiene que escribir es sólo el 10% más trivial.

Perl debe ser usado precompilado vía ModPerl o el impacto en procesador es muy alto.

CGI

Es el sistema más antiguo que existe para la programación de las páginas dinámicas de servidor. Se escriben habitualmente en el lenguaje Perl, sin embargo, otros lenguajes como C, C++ o Visual Basic pueden ser también empleados para construirlos. El programa CGI va escribiendo en la salida estándar el resultado de la ejecución del CGI, que incluye etiquetas HTML, ya que lo que se escribe es una página Web.

ASP.NET

Precedentes.

Microsoft introdujo esta tecnología llamada ASP (Active Server Pages) en diciembre de 1996, por lo que no es nada nueva. Es parte del Internet Information Server (IIS) desde la versión 3.0 y es una tecnología de páginas activas que permite el uso de diferentes scripts y componentes en conjunto con el tradicional HTML para mostrar páginas generadas dinámicamente. La definición contextual de Microsoft es que *"Las Active Server Pages son un ambiente de aplicación abierto y gratuito en el que se puede combinar código HTML, scripts y componentes ActiveX del servidor para crear soluciones dinámicas y poderosas para el web"*.

Microsoft desarrolló una nueva tecnología denominada ASP.NET como parte de su estrategia .NET para el desarrollo Web, con el objetivo de resolver las limitaciones de ASP y posibilitar la creación de software como servicio. ASP.NET es la plataforma unificada de desarrollo web que proporciona a los desarrolladores los servicios necesarios para crear aplicaciones web empresariales.

ASP.NET introduce un nuevo concepto, los "server controls", que permiten a modo de etiquetas HTML tener controles manejados por el servidor que identifican el navegador usado adaptándose para cada navegador. Tareas tediosas como la validación de datos se convierten en fáciles y sencillas.

Lo mejor de todo es la rica biblioteca de clases que lleva incorporada, ya no necesitarás obtener componentes de otras empresas para por ejemplo enviar un email, hacer "upload" de un fichero o generar gráficos en tiempo de ejecución.

1.4.1.2- Del lado del cliente.

Varios son los lenguajes que se encuentran del lado del cliente y que ayudan a realizar operaciones básicas como son validaciones, operaciones de información o estética como tiempo de carga de una página web, mensajes de bienvenida, animaciones, entre otras, estos lenguajes son:

HTML

El lenguaje llamado HTML indica al navegador donde colocar cada texto, cada imagen o cada video y la forma que tendrán estos al ser colocados en la página. El lenguaje consta de etiquetas que tienen esta forma o <P>. Cada etiqueta significa una cosa, por ejemplo significa que se escriba en negrita (bold) o <P> significa un párrafo, <A> es un enlace, etc.

Casi todas las etiquetas tienen su correspondiente etiqueta de cierre, que indica que a partir de ese punto no debe de afectar la etiqueta. Por ejemplo se utiliza para indicar que se deje de escribir en negrita. Así que el HTML no es más que una serie de etiquetas que se utilizan para definir la forma o estilo que queremos aplicar a nuestro documento. Esto está en negrita.

JAVASCRIPT

Javascript es un lenguaje de programación utilizado para crear pequeños programitas encargados de realizar acciones dentro del ámbito de una página web. Se trata de un lenguaje de programación del lado del cliente, porque es el navegador el que soporta la carga de

procesamiento. Su uso se basa fundamentalmente en la creación de efectos especiales en las páginas y la definición de interactividades con el usuario.

Las sentencias escritas en javascript se encapsulan entre las etiquetas `<script>` y `</script>`. por ejemplo, si en el código de una página Web incluimos la sentencia

```
<script>  
window.alert("Bienvenido a mi sitio web. Gracias...")  
</script>
```

FLASH

Flash es una tecnología, y un programa, para crear efectos especiales en páginas web. Con Flash también conseguimos hacer páginas dinámicas del lado del cliente. Flash en realidad no es un lenguaje; Sin embargo, si tuviera que catalogarse en algún sitio quedaría dentro del ámbito de las páginas dinámicas de cliente.

Para visualizar las "películas" Flash, nuestro navegador debe tener instalado un programita (plugin) que le permita visualizarlas.

VBScript (Visual Basic SCRIPT)

Es un lenguaje de programación de scripts del lado del cliente, pero sólo compatible con Internet Explorer. Es por ello que su utilización está desaconsejada a favor de Javascript.

Está basado en Visual Basic, un popular lenguaje para crear aplicaciones Windows. Tanto su sintaxis como la manera de trabajar están muy inspirados en él. Sin embargo, no todo lo que se puede hacer en Visual Basic lo podremos hacer en Visual Basic Script, pues este último es una versión reducida del primero.

El modo de funcionamiento de Visual Basic Script para construir efectos especiales en páginas web es muy similar al utilizado en Javascript y los recursos a los que se puede acceder también son los mismos: el navegador.

CSS (Cascading Style Sheets)

CSS, es una tecnología que permite crear páginas web de una manera más exacta. Gracias a las CSS los desarrolladores son más dueños de los resultados finales de la página, pudiendo hacer muchas cosas que no se podía hacer utilizando solamente HTML, como incluir márgenes, tipos de letra, fondos, colores... Incluso podemos definir nuestros propios estilos en un archivo externo a nuestras páginas, así si en algún momento queremos cambiar alguno de ellos, automáticamente se nos actualizarán todas las páginas vinculadas de nuestro sitio. CSS son las siglas en español Hojas de estilo en Cascada.

AJAX (Asynchronous JavaScript And XML)

AJAX, (JavaScript y XML asíncronos), es una técnica de desarrollo web para crear aplicaciones interactivas. Éstas se ejecutan en el cliente, es decir, en el navegador del usuario, y mantiene comunicación asíncrona con el servidor en segundo plano. De esta forma es posible realizar cambios sobre la misma página sin necesidad de recargarla. Esto significa aumentar la interactividad, velocidad y usabilidad en la misma.

Es una combinación de tres tecnologías ya existentes:

- *XHTML* (o *HTML*) y hojas de estilos en cascada (*CSS*) para el diseño que acompaña a la información.
- *Document Object Model (DOM)* accedido con un lenguaje de scripting por parte del usuario, especialmente implementaciones ECMAScript como JavaScript y JScript, para mostrar e interactuar dinámicamente con la información presentada.
- El objeto *XMLHttpRequest* para intercambiar datos asincrónicamente con el servidor web. En algunos frameworks y en algunas situaciones concretas, se usa un objeto *iframe* en lugar del *XMLHttpRequest* para realizar dichos intercambios.
- *XML* es el formato usado comúnmente para la transferencia de vuelta al servidor, aunque cualquier formato puede funcionar, incluyendo HTML preformateado, texto plano, JSON y hasta EBML.

No constituye una tecnología en sí, sino que es un término que abarca a un grupo de éstas que trabajan conjuntamente.

1.4.1.3- Del lado Cliente/Servidor.

Como bien lo dice su nombre es el que engloba a los dos antes mencionados, permite ejecutar acciones en el lado del servidor y también en el del cliente, esto se debe a que se encuentra compuesto por ambos lenguajes integrándolos en un solo, comenzaremos mencionando a:

DHTML (Dynamic HTML)

No es precisamente un lenguaje de programación. Más bien se trata de una nueva capacidad de la que disponen los navegadores modernos, por la cual se puede tener un mayor control sobre la página que antes.

Cualquier página que responde a las actividades del usuario y realiza efectos y funcionalidades se puede englobar dentro del DHTML, pero en este caso se refiere más a efectos en el navegador por los cuales se pueden mostrar y ocultar elementos de la página, se puede modificar su posición, dimensiones, color, etc.

Proporciona más control sobre la página, gracias a que los navegadores modernos incluyen una nueva estructura para visualizar en páginas web denominada capa. Las capas se pueden ocultar, mostrar, desplazar, etc.

Para realizar las acciones sobre la página, como modificar la apariencia de una capa, se continúa necesitando un lenguaje de programación del lado del cliente como Javascript o VBScript.

En la actualidad, DHTML también puede englobar la programación en el servidor.

Dentro del concepto de DHTML se generalizan también las Hojas de Estilo en Cascada o CSS (Cascade Style Sheets).

XML (eXtensible Markup Language)

Es una tecnología en realidad muy sencilla que tiene a su alrededor otras tecnologías que la complementan y la hacen mucho más grande y con unas posibilidades mucho mayores.

XML, con todas las tecnologías relacionadas, representa una manera distinta de hacer las cosas, más avanzada, cuya principal novedad consiste en permitir compartir los datos con los que se trabaja a todos los niveles, por todas las aplicaciones y soportes.

- Es extensible, lo que quiere decir que una vez diseñado un lenguaje y puesto en producción, igual es posible extenderlo con la adición de nuevas etiquetas de manera de que los antiguos consumidores de la vieja versión todavía puedan entender el nuevo formato.

- El analizador es un componente estándar, no es necesario crear un analizador específico para cada lenguaje. Esto posibilita el empleo de uno de los tantos disponibles. De esta manera se evitan bugs y se acelera el desarrollo de la aplicación.

Si un tercero decide usar un documento creado en XML, es sencillo entender su estructura y procesarlo. Mejora la compatibilidad entre aplicaciones.

1.5- Introducción al .NET Framework.

Microsoft lanza esta nueva tecnología como respuesta a la tecnología Java de Sun. El Framework .NET tiene grandes similitudes con la plataforma Java, por eso todos aquellos que estén familiarizados con Java comprenderán en seguida el funcionamiento de .NET.

Centrándose en el concepto Framework .NET, como el término en inglés dice (Framework = Armazón) es un marco en que las aplicaciones correrán. Las aplicaciones ya no corren directamente bajo el sistema operativo si no que corren bajo este armazón o marco.

Elementos principales .NET Framework:

-CLR (*Common Language Runtime*)

-El conjunto de clases del .NET Framework

-ASP.NET

-Los servicios Web

-Remoting

-Windows Forms

El **CLR** es el motor de ejecución de las aplicaciones .NET, lo que en Java sería la máquina virtual de Java, este motor se encarga de ejecutar todo el código .NET para ello a de ser en dicho lenguaje. El CLR es el encargado de convertir este lenguaje intermedio en lenguaje máquina del

procesador, esto normalmente se hace en tiempo real por un compilador JIT (Just-In-Time) que lleva incorporado el CLR.

El conjunto de clases del .NET Framework es la piedra angular de cualquier desarrollador de .NET, es un rico conjunto de clases, interfaces, tipos que simplifican y optimizan el desarrollo de aplicaciones .NET además de proporcionar acceso a la funcionalidad del sistema. Como desarrolladores el dominio de este conjunto de clases es vital para un buen desarrollo en .NET.

ASP.NET es la parte del .NET Framework dedicada al desarrollo web. A través del servidor web (IIS) nuestras aplicaciones ASP.NET se ejecutarán bajo el CLR y se podrá usar el conjunto de clases del .NET Framework para desarrollarlas, obteniendo así una versatilidad y una potencia nunca antes conseguida en las aplicaciones ASP.

También son destacables los servicios web, que nos permitirán comunicarnos a través de Internet entre diferentes ordenadores, incluso entre distintos sistemas. Así como **.NET Remoting** que nos permite tener objetos en máquinas remotas e invocarlos desde otras máquinas. Y por último los **Windows Forms**, parte del .NET Framework que permite crear aplicaciones en el más clásico de los sentidos.

1.6- Vista general de ASP.NET 2.0

ASP.NET 2.0 se construye sobre la base de ASP.NET 1.x para permitir operaciones de creación y administración de usuarios mucho más sencillas, y la protección de las páginas de una aplicación Web por medio de contraseñas. El nuevo marco incorpora características novedosas para el trabajo con la autenticación y la autorización que se han diseñado para que resulten igualmente atractivas para administradores de sitios Web y desarrolladores.

Los administradores pueden aprovechar la nueva herramienta de administración de sitios Web (Web Site Administration Tool) para crear nuevos usuarios y funciones, y controlar el acceso a las páginas de una aplicación Web. Esta herramienta es un conjunto de páginas ASP.NET preescritas que pueden emplear para configurar una aplicación Web hasta los usuarios sin experiencia en programación.

Los desarrolladores, por su parte, pueden beneficiarse de los nuevos controles Login, que les permitirán generar con rapidez páginas relacionadas con la seguridad en una aplicación Web. Por ejemplo, un desarrollador puede crear una página de inicio de sesión simplemente arrastrando un

control Login hasta una página .aspx. Los controles Login permiten crear tanto páginas de inicio de sesión como páginas de registro y páginas de recuperación de contraseñas sin necesidad de escribir ningún código.

Por último, el marco ASP.NET 2.0 ofrece una serie de características nuevas relacionadas con la seguridad que les resultarán muy interesantes a los desarrolladores más avanzados. La nueva API de pertenencia a grupo es un conjunto de clases que contiene métodos para crear y recuperar información sobre los usuarios de las aplicaciones. Asimismo, también incorpora clases que simplifican el trabajo con las funciones de usuario personalizadas.

La seguridad y el modelo de proveedores

La novedad más significativa que ofrece el marco ASP.NET 2.0 es básicamente que la seguridad funciona. Se puede comenzar a registrar y autenticar a los usuarios comparando la información de la base de datos inmediatamente después de habilitar la autenticación por formularios, sin necesidad de generar tablas de base de datos ni de escribir ningún código. Esto es posible porque el marco ASP.NET 2.0 utiliza el modelo de proveedores para la seguridad.

Uso de SqlMembershipProvider

Utilizar una base de datos de Access resulta adecuado cuando se está desarrollando una aplicación Web o creando una aplicación destinada a un grupo reducido de personas. Sin embargo, si se precisa crear una aplicación más robusta, resultará recomendable almacenar los nombres de usuario y las contraseñas en una base de datos escalable, por ejemplo, una base de datos de Microsoft SQL Server.

Uso de la herramienta de administración de sitios Web para configurar la seguridad

La herramienta de administración de sitios Web que incluye el marco ASP.NET 2.0 permite configurar una aplicación ASP.NET desde una única interfaz de página Web. Puede utilizarla para crear y administrar usuarios y funciones, y controlar el acceso a las carpetas y las páginas de una aplicación Web (la herramienta de administración de sitios Web también se puede emplear para configurar otros aspectos de la propia aplicación).

Uso del control Login

ASP.NET 2.0 contiene una serie de novedosos controles relacionados con la seguridad que se conocen en su conjunto como controles Login. Mediante estos controles Login, se pueden crear páginas estándar de registro, de inicio de sesión y de recuperación de contraseñas sin necesidad de escribir código.

Uso del control CreateUserWizard

Permite crear una página estándar de registro de usuarios. Simplemente agregando la siguiente etiqueta a la página, los nuevos usuarios podrán registrarse en el sitio Web.

```
<asp:CreateUserWizard ID="CreateUserWizard1" Runat="server" />
```

Uso del control PasswordRecovery

El control PasswordRecovery permite que los usuarios de la aplicación Web puedan solicitar un mensaje de correo electrónico de recordatorio de su contraseña (consulte la figura 5). Al igual que el control CreateUserWizard, las propiedades del mensaje de correo electrónico enviado al usuario se definen mediante la propiedad MailDefinition.

Uso del control ChangePassword

El control ChangePassword, como se puede suponer, permite a los usuarios cambiar sus contraseñas registradas. El control crea cuadros de texto en los que se puede facilitar la contraseña original y la nueva.

1.7- Argumentación de la tecnología web escogida.

Se han analizado todas las propuestas investigadas y se ha llegado al consenso debido a que es la que se me ajusta a la situación examinada por las ventajas que ofrece con respecto a las demás como son ahorro significativo del tiempo de desarrollo del software lo mas relevante ya que por su integración con el gestor de base de datos es automática por lo que no se tiene que configurar ningún tipo de conexión con el mismo, usar driver, etc, además porque contiene una paleta de componentes muy útiles que generan una gran parte del código que debe ser implementado con la inserción de componentes con funcionalidades ya predeterminadas, el lenguaje escogido fue el C# lenguaje perteneciente a la tecnología ASP.NET, el cual por sus funcionalidades, prestaciones, por

la robustez del lenguaje, el cual a continuado enriqueciéndose ahora con la versión ASP.NET 2.0 en el cual incorporan nuevas características que le permiten mejorar aun mas su seguridad, fortaleza y comodidad para los programadores que lo usan. Algunas de las características que mas resaltan son:

Modernidad.

- C# incorpora en el propio lenguaje elementos que a lo largo de los años ha ido demostrándose son muy útiles para el desarrollo de aplicaciones y que en otros lenguajes como Java o C++ hay que simular, como un tipo básico **decimal** que permita realizar operaciones de alta precisión con reales de 128 bits (muy útil en el mundo financiero), la inclusión de una instrucción **foreach** que permita recorrer colecciones con facilidad y es ampliable a tipos definidos por el usuario, la inclusión de un tipo básico **string** para representar cadenas o la distinción de un tipo **bool** específico para representar valores lógicos.

Orientado a componentes.

- La propia sintaxis de C# incluye elementos propios del diseño de componentes que otros lenguajes tienen que simular mediante construcciones más o menos complejas. Es decir, la sintaxis de C# permite definir cómodamente **propiedades** (similares a campos de acceso controlado), **eventos** (asociación controlada de funciones de respuesta a notificaciones) o **atributos** (información sobre un tipo o sus miembros).

Instrucciones seguras.

- Para evitar errores muy comunes, en C# se han impuesto una serie de restricciones en el uso de las instrucciones de control más comunes. Por ejemplo, la guarda de toda condición ha de ser una expresión condicional y no aritmética, con lo que se evitan errores por confusión del operador de igualdad (==) con el de asignación (=); y todo caso de un **switch** ha de terminar en un **break** o **goto** que indique cuál es la siguiente acción a realizar, lo que evita la ejecución accidental de casos y facilita su reordenación.

Eficiencia.

- En principio, en C# todo el código incluye numerosas restricciones para asegurar su seguridad y no permite el uso de punteros. Sin embargo, y a diferencia de Java, en C# es posible saltarse dichas restricciones manipulando objetos a través de punteros. Para ello basta marcar regiones de código como inseguras (modificador **unsafe**) y podrán usarse en ellas punteros de forma similar a cómo se hace en C++, lo que puede resultar vital para situaciones donde se necesite una eficiencia y velocidad procesamiento muy grandes.

Compatibilidad.

- Para facilitar la migración de programadores, C# no sólo mantiene una sintaxis muy similar a C, C++ o Java que permite incluir directamente en código escrito en C# fragmentos de código escrito en estos lenguajes, sino que el CLR también ofrece, a través de los llamados **Platform Invocation Services (PInvoke)**, la posibilidad de acceder a código nativo escrito como funciones sueltas no orientadas a objetos tales como las DLLs de la API Win32. Nótese que la capacidad de usar punteros en código inseguro permite que se pueda acceder con facilidad a este tipo de funciones, ya que éstas muchas veces esperan recibir o devuelven punteros.

1.3- Argumentación del Gestor de Base de Datos escogido.

Son varios los gestores de base de datos reconocidos en el mundo por las características que poseen que los hacen estar entre los más usados como son compatibilidad con otros gestores, integración con los diferentes lenguajes de programación, por el soporte que le ofrecen a las aplicaciones, los datos, entre otros. Entre los más exitosos podemos mencionar por parte de los gestores propietarios a Oracle, Microsoft SQL Server, Microsoft Access y por la parte libre tenemos a MySQL, PostgreSQL, Firebird, etc.

Un Sistema Gestor de Base de Datos (SGBD) no es más que un conjunto de programas de propósito general que permite controlar el acceso y la utilización de la base de datos, para incluir, modificar o recuperar información, incluyendo prestaciones para conseguir la independencia, la integridad y la seguridad de los datos y la concurrencia de diversas personas.

Después de un análisis se llegó a la conclusión de que debería usarse Microsoft SQL Server 2005 por el alto por ciento de integración que tiene con Visual Studio, propiciando una alta interoperabilidad y por ende una mayor eficiencia agregada a los sistemas que se desarrollen usando a los mismo, proporcionándole al desarrollador una gran comodidad, pueden crear procedimientos almacenados, funciones y tipos definidos por el usuario en CLR, y agregados definidos por el usuario directamente desde el entorno de desarrollo de Visual Studio.

Pueden utilizar estos nuevos objetos de base de datos directamente desde Visual Studio sin necesidad de cambiar de herramienta. Visual Studio 2005 es directamente compatible con todos los tipos de datos nuevos de SQL Server, tales como XML nativo. También puede agregar los objetos de base de datos en CLR al mismo sistema de control de origen que utiliza para todos sus proyectos en Visual Studio, y de esta manera el nivel de integración y seguridad de los procesos de desarrollo serán aún mayores.

La integración entre SQL Server y CLR brinda una variedad de importantes beneficios:

Mejor modelo de programación. Los lenguajes de programación compatibles con .NET Framework son más ricos que Transact-SQL en muchos aspectos, y ofrecen construcciones y capacidades novedosas para los encargados del desarrollo de SQL Server.

Más seguridad. El código administrado se ejecuta en un entorno CLR, alojado en el motor de la base de datos. Esto permite que los objetos de la base de datos .NET Framework sean más seguros que los procedimientos almacenados extendidos de versiones anteriores de SQL Server.

Se pueden desarrollar servicios Web XML a nivel de la base de datos, haciendo que SQL Server se convierta en una escucha http. De esta forma ofrece un nuevo tipo de capacidad de acceso a los datos para aplicaciones que están centralizadas en los servicios Web, proporciona protección contra el SQL Injection. El soporte para los estándares abiertos existentes y emergentes tales como el protocolo de transferencia de hipertexto (HTTP), XML, Simple Object Access Protocol (SOAP), XQuery y el lenguaje XML Schema definition (XSD) también facilitan la comunicación a través de gran cantidad de sistemas empresariales.

Con frecuencia, los sistemas de bases de datos son demasiado complicados para construir aplicaciones simples. Microsoft Visual Studio y SQL Server 2005 reducen esta complejidad al proporcionar un entorno de desarrollo simple pero eficaz para construir aplicaciones orientadas a los datos. Los encargados del desarrollo pueden diseñar esquemas, agregar datos y consultar bases de datos locales, todo ello dentro del entorno de Visual Studio 2005.

1.9- Argumentación de la metodología de desarrollo de software escogida.

El uso de las metodologías en la construcción de software es de suma importancia ya que le permite al equipo de desarrollo hacer un planeación y modelación del problema a automatizar con un alto nivel de detalle con el objetivo de comprender un problema determinado del cual se debe hacer una propuesta de solución, mientras mejor sea el grado de comprensión del problema y compromiso de los integrantes mayor será la calidad del producto final. Estas metodologías están orientadas a lograr la eficiencia del trabajo en equipo de un grupo de personas.

Estas metodologías utilizan un conjunto de herramientas para la modelación de los problemas, entre estas se encuentran las herramientas Rational Software, Visual Paradigm y Umbrello entre otras.

Hoy en día entre todas las metodologías existentes es el proceso unificado de modelado RUP la que goza de mayor popularidad y esto se debe a que se caracteriza por ser:

Dirigido por casos de uso, los casos de uso reflejan los que los usuarios futuros quieren y desean, esto se representa a través de los requerimientos, de aquí en adelante los casos de uso guían el proceso de desarrollo y todos los diagramas que se obtengan representan la realización de los casos de uso.

Centrado en la arquitectura, se llega a un consenso y se establece la arquitectura básica a se crean los cimientos del sistema que son necesarios como base para comprenderlo, desarrollarlo y producirlo económicamente.

Iterativo incremental, RUP propone que cada fase se desarrolle en iteraciones, una iteración involucra actividades de todos los flujos y se obtendrá como resultado final una versión del producto.

1.10- Otros programas utilizados en la propuesta realizada.

Después de un largo análisis y de haberse propuesto una solución se tomaron algunos consensos sobre las herramientas que sería necesario utilizar para llegar a concretar esta propuestas las mismas son Adobe Photoshop CS2 para editar las imágenes del sitio por su alta resolución para procesarlas, por la gran variedad de extensiones soportadas, fácil de usar; otro fue el Macromedia

Flash 8 para el trabajo con las películas flash, con el fin de darle dinamismo al sitio web y por las posibilidades de crear objetos, acciones todas en un componente ajeno al código de la aplicación, el Embarcadero Erwin Studio 6.6 por las facilidades a la hora de generación de la base de datos, procedimientos almacenados, entre otros todo a parte de un modelo relacional, Rational Rose Enterprise Edition 2003 para lo que es toda la modelación del negocio por todas las ventajas que aporta, así como el Visual Studio .Net y SQL Server 2005 para el desarrollo principal de la aplicación por las características anteriormente mencionadas.

1.11- Sistemas Actuales (Estado del Arte).

1.11.1- Internacionales.

En la actualidad el Internet tiene una amplia gama de usos desde los entretenimientos y servicios de correo, música y video online hasta los negocio, muchas empresas tienen sus negocio online para expandir sus horizontes por lo que se les hace necesario tener sistemas de gestión de información que integran y automatizan muchas de las prácticas de negocio asociadas con los aspectos operativos o productivos de una empresa.

El propósito fundamental de un ERP es otorgar apoyo a los clientes del negocio, tiempos rápidos de respuesta a sus problemas así como un eficiente manejo de información que permita la toma oportuna de decisiones y disminución de los costos totales de operación.

Los objetivos principales de los sistemas ERP son:

Optimización de los procesos empresariales.

Acceso a toda la información de forma confiable, precisa y oportuna (integridad de datos).

La posibilidad de compartir información entre todos los componentes de la organización.

Eliminación de datos y operaciones innecesaria de reingeniería).

Las características que distinguen a un ERP de cualquier otro software empresarial es que deben de ser sistemas:

- **Integrales**, porque permiten controlar los diferentes procesos de la compañía entendiendo que todos los departamentos de una empresa se relacionan entre sí.

- **Modulares**. Los ERP entienden que una empresa es un conjunto de departamentos que se encuentran interrelacionados por la información que comparten y que se genera a partir de sus procesos.

- **Adaptables**. Los ERP están creados para adaptarse a la idiosincrasia de cada empresa. Esto se logra por medio de la configuración o parametrización de los procesos de acuerdo con las salidas que se necesiten de cada uno.

1.11.2- Nacionales.

Se trata de un sistema para realizar la gestión de la información en un departamento docente, teniendo en cuenta la necesidad del intercambio de información, rápido acceso y nivel de actualización. Se desarrolla en ambiente Web, utilizando PHP e Internase y fue elegido como gestor de base de datos Firebird. Está constituido por cuatro módulos:

- Módulo de Capital Humano.
- Módulo de Formación del Profesional.
- Módulo de Educación de Postgrado.
- Módulo de Ciencia y Técnica.

El presente trabajo abarcó también el trabajo de atención de los profesores con alumnos ayudantes e instructores recién graduados. Por otra parte, asiste en la elaboración de reportes e informes, que servirán para tomar decisiones, puesto que se almacena un gran volumen de información relacionada con esta rama, que además estará disponible para todos los usuarios que deseen consultarla, teniendo en cuenta diferentes niveles de acceso.

Como se puede ver es un trabajo que contiene muchos puntos coincidentes con el nuestro y la diferencia de que fue hecho para un departamento completo y el nuestro solo abarca la parte de la docencia y un planificación, este sistema fue desarrollado en La Universidad de Matanzas "Camilo Cienfuegos" y presentado en **XII CONGRESO DE INFORMÁTICA EN LA EDUCACIÓN.**

1.12- Conclusiones.

Después de un crítico y exhaustivo análisis sobre las posibles herramientas a usar para desarrollar la propuesta hecha, se pudo arribar a la conclusión de que para la creación y generación del modelo de base de datos debería usarse Embarcadero Edwin Studio 6.6, para la modelación del negocio a RUP utilizando el software Rational Rose Enterprise Edition 2003, para la implementación del sistema se seleccionó a Visual Studio .NET 2005 y para el almacenamiento de los datos el gestor Microsoft SQL Server 2005. La selección de estas herramientas se basó en sus características y ventajas que brindan en las condiciones actuales en las que se encuentra la propuesta para el desarrollo de un sistema, por lo que según el resultado aquí expresado, el análisis arrojó a estas como las más eficientes para lograr un desarrollo exitoso de la propuesta realizada.

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

2.3- Problema y Situación Problemática.

2.3.1- Flujo actual de los procesos involucrados en el campo de acción.

Los procesos involucrados en el negocio en estos momentos los realizan los integrantes del departamento, quienes tienen un conjunto de tareas asignadas que deben cumplir para lograr el funcionamiento del departamento, dichos procesos son los siguientes:

Actualizar datos: Este comienza cuando el jefe de departamento, profesor, vice-decana docente u otro solicitan una nueva versión al día de cualquier información que se necesite en el departamento.

Confecionar Plan de Trabajo: Es solicitado por el jefe de asignatura, colectivo o departamento al profesor y este confecciona el orden de actividades a realizar en la asignatura que imparte para un tiempo determinado en un documento Word o Excel y se lo envía por correo a los solicitantes, el mismo puede estar sujeto a cambios.

Evaluar Alumno Ayudante: Cuando culmina una etapa docente el profesor que atiende a cada alumno ayudante da una opinión del trabajo que ha realizado en la etapa de un semestre basada en las actividades realizadas, calidad de las clases y esta es registrada en un documento plantilla que puede ser un Word o Excel con la calificación otorgada.

Evaluar Profesor: No es más que la valoración hecha de un profesor según su trabajo impartiendo clases, atendiendo a sus estudiantes, por las actividades realizadas, entre otras que su jefe hace de él, una vez termina un periodo docente cuyo resultado es una calificación guardada en un documento plantilla que puede ser un Word o Excel

Publicar Horario Docente por Años: Es la planificación realizada por la vice-decana docente en un documento de los locales docentes, asignaturas, profesores y distribuido a los estudiantes y profesores por correo y publicado impreso, dicho documento puede ser cambiado debido a posibles afectaciones en las aulas o laboratorios.

Publicar Horario de Profesor: Este es un documento que es elaborado por la vice-decana docente al inicio de cada semestre, donde se les da a conocer a través del correo a cada profesor las horas de comienzo, duración y días de la semana de los turnos de clases que deben impartir por quincena durante un semestre.

Publicar Horario de Laboratorio: Es la planificación de los laboratorio ocupados y libres de dos semanas para un semestre docente, que puede ser modificado.

Publicar Horario de Aula: Es la planificación de las aulas ocupadas y libres de dos semanas para un semestre docente, que puede ser modificado.

Publicar Noticias de la Semana: Al comenzar de cada semana, el jefe del departamento envía por correo las actividades de carácter semanal a cada integrante del departamento, redactadas en un documento Word o Excel, el cual puede ser modificado.

Publicar Noticias de Última Hora: Nos son más que las informaciones de corto plazo de duración y carácter rápido.

Realizar Plan de Actividades: Cuando se inicia un nuevo semestre, el jefe del departamento, colectivos y asignaturas elaboran un documento Word o Excel donde establece las fechas, horarios y detalles de las distintas tareas que realizaran en el departamento para un período de tiempo determinado, el cual puede cambiar debido a la dinámica diaria de la universidad.

Administrar Dpto: Es el control que ejerce una persona sobre las personas, tareas y actividades de un departamento.

Actualizar Documentos: Es el proceso de sobre escribir versiones actualizadas de documentos sobre viejas versiones del mismo.

2.3.2- Análisis crítico de cómo se ejecutan actualmente esos procesos.

En estos momentos en el departamento todos los procesos que se hacen son a través de la vía tradicional ubicándonos en un marco como lo es nuestra universidad, ya que no posee un sistema o herramienta que centralice, almacene y administre toda la información del mismo por lo que el jefe de departamento tiene que estar constantemente enviando correos con las informaciones necesarias para el correcto funcionamiento del mismo, como son los distintos documentos como son el plan de actividades del departamento, las noticias semanales, de ultima hora, horarios docentes, tiene que tener todos los planes de trabajo de los colectivos, que le son enviados por

correo por cada uno de jefes de colectivo de cada asignatura, así como llevar documentos con las incidencias y actividades de estos que le servirán para al finalizar un proceso docente darles una evaluación a los mismos que se contempla en una plantilla que se basa en el cumplimiento, incumplimientos y actividades realizadas con el fin de valorar el trabajo desempeñado al finalizar un periodo de tiempo, todo lo mencionado anteriormente es almacenado en su computadora personal.

Los jefes de colectivos tiene los mismos documentos que el jefe de departamento con la única diferencia de que los planes de actividades son confeccionados para el colectivo, lo único que varia es el plan de actividades que es para los profesores de su colectivo solamente. En el caso de los jefes de asignaturas también tiene los mismos documentos que el jefe de departamento y la diferencia la constituye el plan de actividades que es para una asignatura específica. Los profesores tienen que tener los documentos en los que mantengan almacenadas las actividades de cada uno de sus alumnos ayudantes, en el caso de que tenga alguno, así como los planes de trabajo el suyo y de su (s) alumno (s) ayudante (s) para al culminar un semestre emitir una evaluación del desempeño de el (los) mismo (s), todas estas informaciones son almacenadas y enviadas por correo en un documento por el profesor a su jefe.

La vise-decana docente es la encargada de hacer los horarios de las asignaturas de un semestre para cada uno de los años correspondientes, la cual elabora en un documento Excel y lo envía por correo a todos los estudiantes de la facultad, decana y jefes de departamentos.

Todos estos métodos mencionados anteriormente traen como consecuencia una demora en la entrega ya que una vez confeccionados cada uno de los documentos los cuales son posteriormente enviados por la misma vía el correo siguiendo la estructura de niveles establecida en el departamento, que explicamos seguidamente, los alumnos ayudantes los envían a sus tutores que son los profesores, estos los revisan en caso de que tengan algún error los regresan a sus emisores hasta que estén correctos de lo contrario los guardan en su computadora para velar por su cumplimiento, los profesores envían sus planes de trabajo a sus jefes de asignaturas, así los jefes de asignaturas reportan a los jefes de colectivos y estos últimos al jefe de departamento, todos siguiendo los mismos pasos que siguió el profesor con su alumno ayudante.

Como se puede notar todos estos procesos generan una notable demora en la entrega en tiempo ya que los procesos son los mismos lo que los diferencia es que son ejecutados a diferentes niveles que es en fin el origen del retraso, disponibilidad de dicha información, se hace difícil

consultar la misma incluso estando en el mismo departamento, un ejemplo para que sea de mejor entendimiento es que el colectivo de Programación no puede consultar los documentos del colectivo de Ingeniería de Software debido a que esta información solo la tienen los integrantes de ese colectivo y el jefe de departamento, para poder ser consultada tiene primero ser solicitada y después enviada, las dos acciones por lo general se hacen a través del correo, otra de las incidencias es ocasiones si el volumen de la información resulta ser grande le puede traer como consecuencias perdida, duplicación o atraso para encontrar la información que se necesita en un momento determinado, por lo que le reporta una demora en la realización de su trabajo y por último también cuando existe algún error en las informaciones proporcionadas tiene que enviar un correo señalándolos y entonces el documento le será enviado nuevamente ya corregido por la persona que cometió el error y nuevamente revisado, adicionando más pérdida de tiempo para la realización de su trabajo.

Para dar a conocer a los integrantes del departamento las distintas informaciones mientras más urgentes sean más problemas ocasionan, a veces no llegan a todos o simplemente la reciben cuando ya paso la actividad, tienen que estar constantemente revisando el correo o preguntándose entre ellos para saber si hay algo planeado por el departamento siendo los mas afectados los que están en proyecto y los externos, esto se debe a que no hay un lugar donde se pueda publicar las distintas informaciones y noticias que puedan ser accedidas por todos, en cualquier momento y desde cualquier lugar de la red. En ocasiones se necesita saber de un profesor determinado si se encuentra impartiendo clases o no, si lo está en que aula o laboratorio, con que grupo se encuentra, a que hora y cuantos turnos tiene que impartir en un día determinado, pero como no se dispone de un horarios y un lugar en el que se pueda publicar la carga docente de cada profesor para que todos puedan consultarla es muy poco probable saberlo.

Todas estas dificultades pueden ser resueltas con una base de datos que contenga toda esta información, la cual sería mostrada con consultas a la misma. Por lo que se evidencian la gestación de este sistema con el que se piensa hacer más eficientes, fáciles y cómodas todas las tareas y procesos que en el departamento se desarrollan en aras de elevar la calidad de su funcionamiento.

2.3.3- Descripción del sistema propuesto.

Por la necesidad de gestionar toda la información del departamento de la especialidad de la facultad 4, de eliminar la duplicación de documentos, posibles errores de confección por la dependencia entre cada documento, de acelerar los procesos involucrados, así como mantener la organización y control del personal que conforma el departamento con toda su información personal actualizada, de publicar, acceder a toda la información y datos importante del departamento es que se propone la realización de esta aplicación.

El sistema ofrece la automatización de todas las solicitudes de reportes por parte de los usuarios pertenecientes al departamento, todos estos datos e informaciones estarán almacenados, se accederán a través de peticiones que serán atendidas dependiendo de los mecanismos implementados. Todos los profesores podrán confeccionar y almacenar sus planes de trabajo, actividades realizadas, entre otros documentos en el sistema los cuales podrán ser accedidos por cualquier integrante del departamento.

Todas las informaciones y noticias se encontrarán accesibles para todos de forma que puedan mantener al día con el estado del departamento, estas podrán ser insertadas y eliminadas por las personas autorizadas. Los horarios serán insertados por la Vise-Decana Docente, así como editados y eliminados, los que servirán para organizar el trabajo de los profesores, para la planificación de los distintos locales con que cuenta la facultad 4 y comunicar a los estudiantes como estarán organizadas las distintas materias que recibirán para cada grupo, por cada año el lugar donde se impartirán las clases y serán accedidas a través de solicitudes echar por los usuarios que les interese consultarla.

Se concibió con el objetivo de hacer más simple el desarrollo dividir el sistema en cuatro módulos:

Módulo de Administración.

Es el encargado de gestionar la configuración de la aplicación, los roles de los diferentes usuarios del sistema.

Módulo de Publicaciones.

Es el que realizará la inserción, almacenamiento y publicación de todos los documentos, informaciones y noticias del departamento.

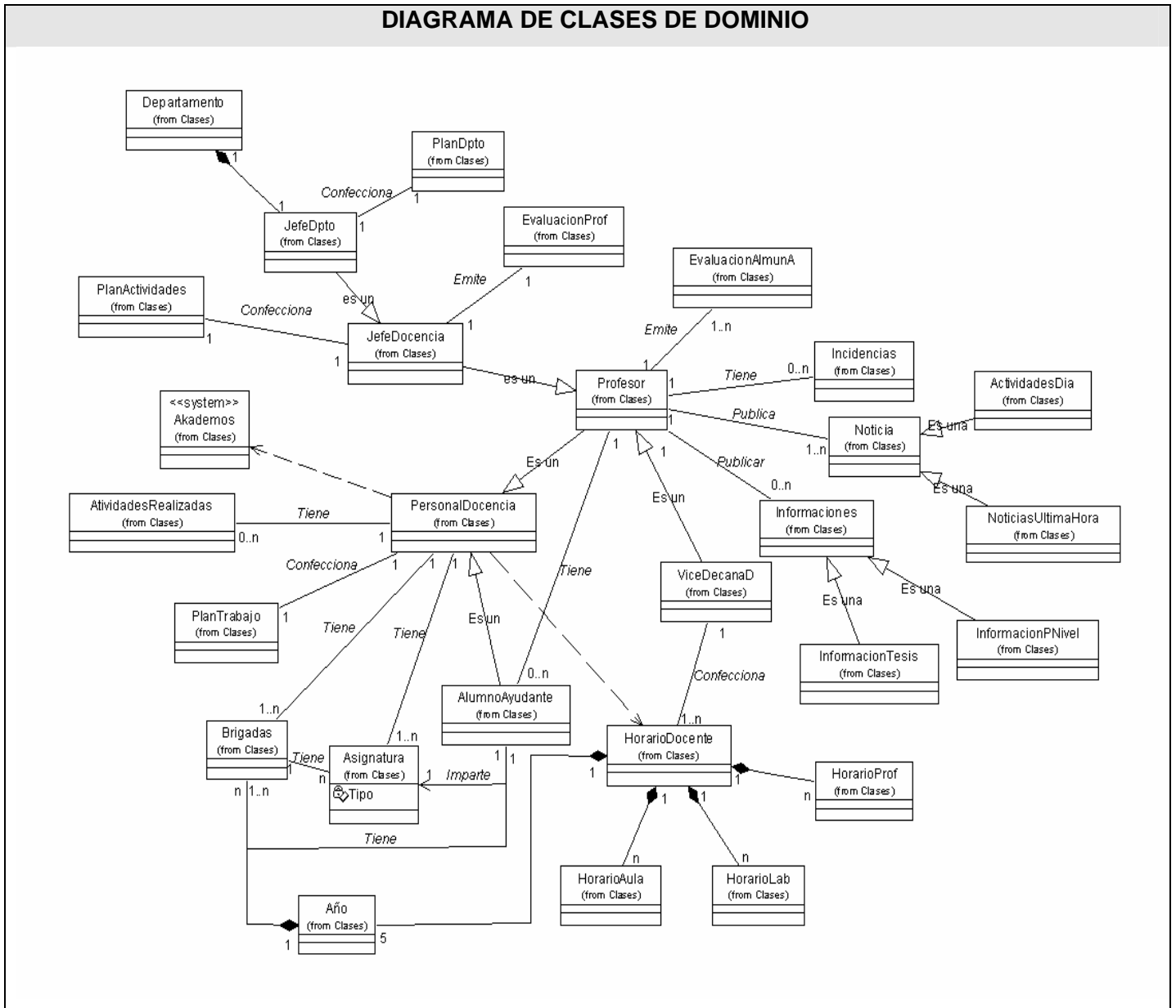
Módulo de Horarios.

A través de este se podrá tener acceso a la planificación realizada de los horarios docentes de los profesores, estudiantes y locales de la facultad, los que podrán ser insertados, editados y eliminados.

Módulo de Reportes.

En este módulo se obtiene reportes y estadísticas de los usuarios almacenados, de las asignaturas, los profesores y las diferentes categorías docentes.

2.3.4- Modelo Conceptual (M. de Dominio).



2.3.5- Glosario de Términos.

Departamento: Es el lugar físico que agrupa a un conjunto de personas de especialidades docentes distintas y todos son dirigido por un jefe.

JefeDpto: Es la persona que controla y dirige las tareas y personas del departamento.

PlanDpto: Son las actividades y que se establecen en el departamento por el cual sus integrantes deberán regirse y realizar las acciones que aquí se acuerden.

PersonalDocencia: Es un rol que generaliza ActividadesRealizadas, PlanTrabajo, Asignatura, Grupo y HorarioDocente.

ActividadesRealizadas: Se refiere al cumplimiento de las tareas asignadas a una persona determinada por su jefe para un periodo o semestre docente.

PlanTrabajo: Son un conjunto de tareas y responsabilidades que son planificadas por y para una persona para su posterior cumplimiento que puede ser un alumno ayudante o profesor.

Asignatura: Materia docente que se le imparte a los estudiantes según su año y contribuye a su formación curricular en una carrera.

Grupo: Es un conjunto de personas que forman la matrícula de un aula.

HorarioDocente: Es la planificación y organización de las aulas, laboratorios, grupos, asignaturas de forma que se pueda hacer posible el proceso docente educativo de los estudiantes.

AlumnoAyudante: Es aquellos estudiantes preparados que asume la tarea de impartir las clases de una asignatura determinada y desempeña el rol de un profesor de forma total o parcial.

Profesor: Es aquella persona con un título en una materia que imparte una asignatura determinada a un grupo de estudiantes y que por lo general es un licenciado o ingeniero.

InformacionesUltimaHora: Son datos que se dan a conocer del departamento, de duración a corto plazo que pueden ser reuniones, informaciones, etc.

Actividades del Día: Son informaciones que del departamento da a conocer con frecuencia semanal.

Información de Pruebas de Nivel (PN): Son todos los datos que se emitan de las mismas como son; Fechas de las distintas convocatorias, notas, aclaraciones, etc.

Informaciones de Tesis: No son más que la publicación de las fechas cuando se van a realizar, los distintos, cortes, predefensa, defensa, aclaraciones u otros datos.

Incidencias: Son los incumplimientos y otras hechos negativos hechos por un profesor que le llevan a ser calificado de incumplidor.

HorarioCargaProfesor: Es donde se muestra los grupos, turnos en los que el profesor estará impartiendo clases o sea las horas en las que tendrá ocupadas, desocupadas y así como las aulas en las que estará.

EvaluacionAlumnoA: Documento que basado en el cumplimiento o incumplimiento de las actividades, tareas y deberes como alumno ayudante le sirve al tutor como respaldo con el fin de acreditar la capacidad para impartir clases, así como su desempeño.

Incidencias: Son los incumplimientos de los profesores reflejados en un documento.

Noticias: Es un rol que generaliza NoticiasUltimaHora, ActividadesDia, InformacionPNivel, InformacionTesis.

NoticiasUltimaHora: Son las informaciones de carácter urgente o de corto plazo para los integrantes del departamento.

ActividadesDia: Son las informaciones de carácter semanal las cuales se quieren dar a conocer a los integrantes del departamento.

InformacionPNivel: Son las noticias, todo lo concerniente al tema de las pruebas de nivel como son las fechas, años y asignaturas que serán examinados, posibles cambios, entre otras que sean por la comisión que atiende este tema.

InformacionTesis: Son todos los detalles como fechas, informaciones de cortes, predefensas, defensas, comisiones de revisión de tesis, entre otras planificadas por el departamento o la universidad.

JefeDocencia: Es un rol que generaliza EvaluacionProf y PlanActividades.

EvaluacionProf: Es el documento que basado en el cumplimiento, incumplimiento, actividades realizadas y deberes de un profesor le permite a su jefe evaluar y valorar el trabajo desempeñado por él durante un periodo determinado.

PlanActividades: Son el conjunto de tareas y eventos que son planificados para organizar y planificar el funcionamiento de una asignatura o un colectivo, por el cual los profesores registrarán su comportamiento.

2.4- Especificación de los requisitos del software.

2.4.1- Requerimientos Funcionales.

R1- Permitir a los usuarios autenticarse en el sistema con su usuario de dominio.

R2- Gestionar usuarios del sistema de acuerdo a los niveles de acceso.

R2.1- Insertar Role.

R2.2- Modificar Role.

R2.3- Eliminar Role.

R3- Administrar los reportes del sistema.

R3.1- Mostrar reportes con la cantidad de profesores por asignaturas.

R3.2- Mostrar reporte con la cantidad de alumnos ayudantes por asignaturas.

R3.3- Mostrar reportes de la cantidad de profesores por categorías docentes.

R3.4- Obtener reporte con la información del profesor.

R3.5- Obtener reporte con la información del alumno ayudante.

R3.6- Mostrar reporte con la cantidad de alumnos ayudantes por profesor.

R4- Publicar Plan del Departamento.

R5- Publicar los planes de trabajo, planes de actividades, plan del departamento de los profesores y alumnos ayudantes.

R6- Administrar noticias del sistema.

R6.1- Publicar las informaciones de última hora.

R6.2- Publicar las actividades de la semana.

R7- Administrar informaciones del sistema.

R7.1- Publicar la información de las pruebas de nivel para los estudiantes.

R7.2- Publicar la información para los estudiantes sobre la tesis.

R8- Administrar horarios del sistema.

R8.1- Consultar los horarios docentes de los grupos.

R8.2-Consultar los horarios de los laboratorios.

R8.3- Consultar los horarios de las aulas.

R8.4- Consultar los horarios docente de los profesores.

R8.5- Insertar Horario.

R8.6- Modificar Horario.

R8.7- Eliminar Horario.

R9- Permitir a los usuarios contactar al administrador del sistema por el correo.

R10- Mostrar vínculos a otros sitios web en nuestra Intranet.

R11- Gestionar las Asignatura.

R11.1- Insertar Asignatura.

R11.2- Modificar Asignatura.

R11.3- Eliminar Asignatura.

R12- Gestionar Profesor.

R12.1- Insertar Profesor.

R12.2- Modificar Profesor.

R12.3- Eliminar Profesor.

R13- Gestionar Alumno Ayudante.

R13.1- Insertar Alumno Ayudante.

R13.2- Modificar Alumno Ayudante.

R13.3- Eliminar Alumno Ayudante.

2.4.2- Requerimientos no Funcionales.

Apariencia o interfaz externa.

- Cada página no debe exceder los 75 kb en imágenes.
- No debe usar tecnología de frames.
- Se hace uso de plantillas web para mejorar la apariencia de las páginas.
- Interfaz fácil de utilizar y de manejar para los usuarios.

Usabilidad.

- Es necesaria una explicación a la ViceDecana docente a la hora de llenar los horarios docentes en aras de que la información sea entrada correctamente y no provoque ningún error o inconsistencia en el momento en que se vaya a mostrar un horario que se solicite.

- Consistencia en la interfaz de usuario.
- Reducción de errores humanos.

Software.

Servidor

- El sistema se desarrollará en Visual Studio .NET 2005.
- El sistema utilizará una Base de Datos implementada en Microsoft SQL Server 2005.
- El servidor puede estar montado en una máquina con Windows XP o Advanced Server versión 2000 o superior, debe tener instalado el framework 2.0 o superior además del gestor de SQL Server 2005 y el IIS (Internet Information Service) o Apache 1.3.27 o superior actualizadas.
- También puede ser montado en un sistema libre como GNU/Linux, por lo que hay que instalar la herramienta libre MONO 1.2.4 o superior con un servidor Apache XSP (eXtensible Server Pages).

Cliente.

- El sistema se podría visualizar en terminales clientes que puede tener sistemas operativos Windows desde NT/2000/XP/2003/Vista y GNU/Linux.
- La aplicación podrá ser vista en navegadores como Internet Explorer 5 o superior, Mozilla Firefox todas sus versiones y Opera 7.54 o superior.

Hardware

- Se requiere como mínimo un PC con un Celeron 1.6 GHz o Pentium IV a 1.6 GHz con 512 MB de RAM, 15 GB de disco duro y una tarjeta de red que soporte velocidades de 100 Mbps o superior.

- Máquina con una Pentium III o superior para los clientes.
- La comunicación de las terminales clientes con el servidor será a través de la red local.

Seguridad y Privacidad.

- Los datos no pueden viajar de forma transparente por la red.
- El sistema contará con un conjunto de permisos (roles) para usuarios que permitirá controlar el acceso y proteger la información de los mismos.
- Las operaciones realizadas por los usuarios serán registradas en el sistema, permitiendo el uso de cookies por el cliente.
- Se le da la opción al usuario de seleccionar la mayor parte de las entradas que quiera hacer, evitando así que tenga que escribirlas y las que escribe son validadas para evitar inconsistencias y errores.
- Los permisos de los usuarios son los menos posibles para proteger la consistencia y disponibilidad de los datos.

Funcionalidades

- Se construyeron el número de páginas mínimas, necesarias con el objetivo de hacerle la navegación al usuario lo más fácil posible y en la misma se ejecutan todas las funcionalidades posibles.

Confiabilidad.

- Se garantiza que los accesos a los distintos recursos estén bien especificadas y así como todas las entradas al sistema se encuentren bien validadas.

Portabilidad.

- Se prevé que el sistema funcione y se muestre sobre cualquier plataforma siendo Windows la más compatible, puesto que fue desarrollado sobre la misma.

2.5- Definición de los casos de uso.

2.5.1- Definición de los actores.

Actores	Justificación
Jefe de Departamento	Es el que administrará el sistema, además de confeccionar el plan del departamento.
Jefe Docencia	Es un rol que generaliza al jefe de colectivo, asignatura y confeccionar un plan de actividades.
Profesor	Es un rol que generaliza a ViceDecana Docente y Jefe Docencia y el encargado de publicar noticias e informaciones.
Usuario del Sistema	Es el rol que generaliza a alumno ayudante y profesor, además se autenticará en el sistema, realizar reportes y publicar plan de trabajo.
ViceDecana Docente	Es la encargada de publicar los horarios de prof, docente, aula, lab
Directorio UCI	Es el encargado de proporcionar el acceso al sistema con el usuario de dominio.

2.5.2- Listado de casos de uso del sistema.

CUS- 1.	Autenticar en el Sistema
Actor	Usuario del Sistema
Descripción	Es la entrada al sistema utilizando los datos del directorio de la uci a través del protocolo LDAP para comprobar los datos.
Referencia	R1- Permitir a los usuarios autenticarse en el sistema con su usuario de dominio, CUS_Comprobar Datos.

CUS- 2.	Realizar Reportes
Actor	Usuario del Sistema
Descripción	El usuario solicita una información determinada del sistema a través de un reporte, la aplicación la busca y se la muestra.
Referencia	R3- Administrar los reportes del sistema. R3.1- Mostrar reportes con la cantidad de profesores por asignaturas. R3.2- Mostrar reporte con la cantidad de alumnos ayudantes por asignaturas. R3.3- Mostrar reportes de la cantidad de profesores por categorías docentes. R3.4- Obtener reporte con la información del profesor. R3.5- Obtener reporte con la información del alumno ayudante. R3.6- Mostrar reporte con la cantidad de alumnos ayudantes por profesor.

CUS- 3.	Confeccionar Horario
Actor	ViceDecana Docente
Descripción	Se planifica y organiza por los grupos de cada años, profesores y locales las tareas a realizar para un semestre docente.
Referencia	R8- Administrar horarios del sistema. R8.1- Consultar los horarios docentes de los grupos. R8.2-Consultar los horarios de los laboratorios. R8.3- Consultar los horarios de las aulas. R8.4- Consultar los horarios docente de los profesores.

CUS- 4.	Publicar Informaciones.
Actor	Profesor
Descripción	Se pone a disposición de los usuarios un conjunto de informaciones de interés para profesores y alumnos.
Referencia	R7- Administrar informaciones del sistema. R7.1- Publicar la información de las pruebas de nivel para los estudiantes. R7.2- Publicar la información para los estudiantes sobre la tesis.

CUS- 5.	Publicar Noticias.
Actor	Profesor
Descripción	Se publican un grupo de noticias del departamento de interés para sus interantes.
Referencia	R6- Administrar noticias del sistema.

	R6.1- Publicar las informaciones de última hora. R6.2- Publicar las actividades de la semana.
--	--

CUS- 6.	Publicar Plan de Actividades.
Actor	Jefe Docencia
Descripción	Se publican un conjunto actividades que deben ser cumplidas por un colectivo o por una asignatura.
Referencia	R5- Publicar los planes de trabajo, planes de actividades, plan del departamento de los profesores y alumnos ayudantes.

CUS- 7.	Publicar Plan del Dpto.
Actor	Jefe del Departamento
Descripción	Grupo de actividades planificadas para el departamento que deben ser ejecutadas.
Referencia	R4- Publicar Plan del Departamento.

CUS- 8.	Administrar.
Actor	Jefe del Departamento
Descripción	Es el control total de la aplicación por parte de un usuario.
Referencia	R7- Administrar informaciones del sistema.

CUS- 9.	Publicar Plan de Trabajo.
Actor	Usuario del Sistema
Descripción	Grupo de tareas que deben cumplirse.

Referencia	R5- Publicar los planes de trabajo, planes de actividades, plan del departamento de los profesores y alumnos ayudantes.
-------------------	---

CUS- 10.	Validar Datos de Usuario.
Actor	Autenticar en el Sistema.
Descripción	Verifica que el usuario y la contraseña entrados por la aplicación coincidan con los de dominio.
Referencia	R1- Permitir a los usuarios autenticarse en el sistema con su usuario de dominio.

CUS- 11	Gestionar Horario.
Actor	Confeccionar Horario.
Descripción	Incluye todos los procesos de insertar, modificar y eliminar un horario docente.
Referencia	R8- Administrar Horarios del Sistema. R8.5- Insertar Horario. R8.6- Modificar Horario. R8.7- Eliminar Horario.

CUS- 12.	Gestionar Asignatura.
Actor	Confeccionar Horario.
Descripción	Son todos los procesos de insertar, modificar y eliminar las asignaturas.
Referencia	R11- Gestionar las Asignatura. R11.1- Insertar Asignatura.

	R11.2- Modificar Asignatura. R11.3- Eliminar Asignatura.
--	---

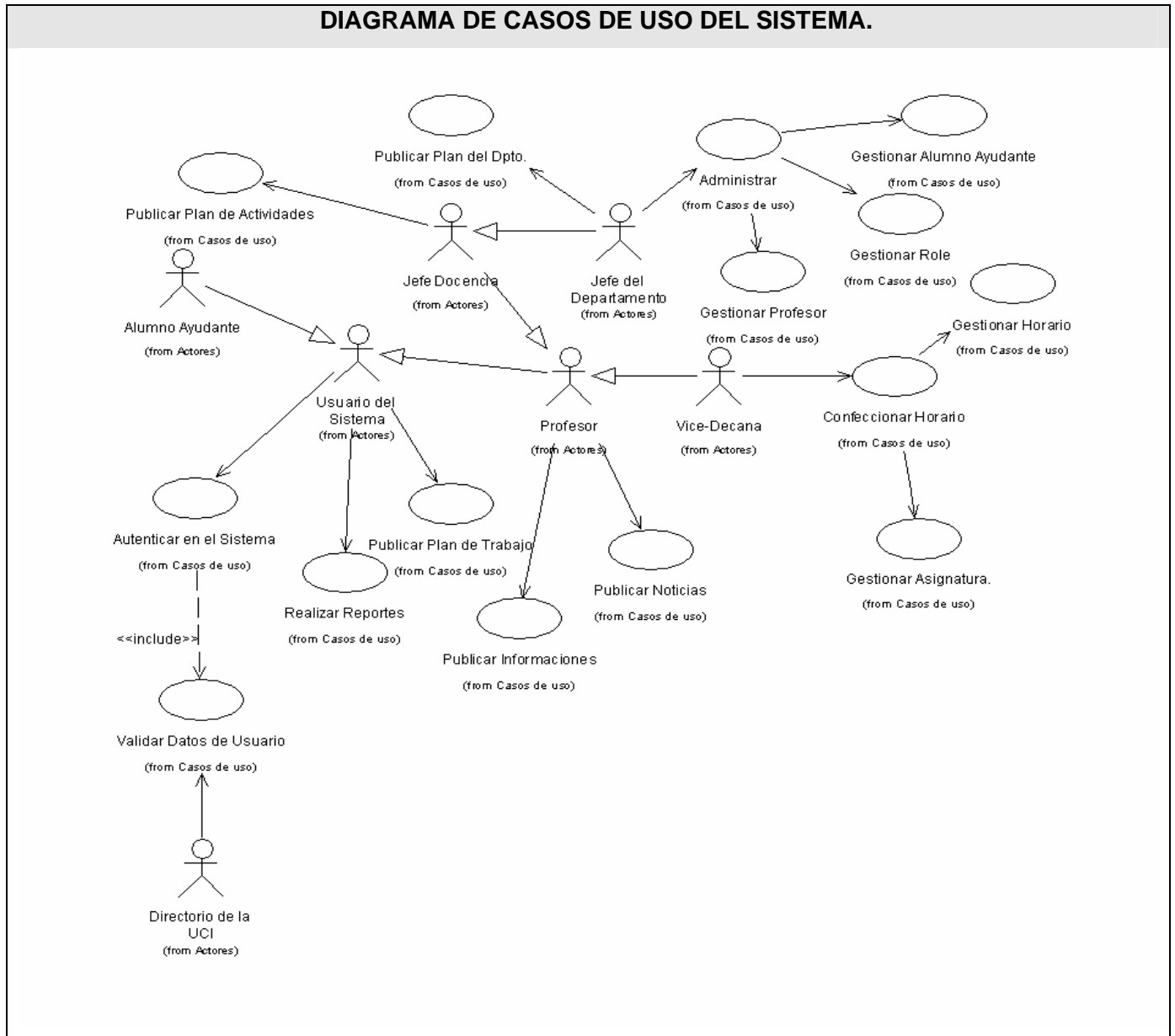
CUS- 13	Gestionar Profesor.
Actor	Administrar.
Descripción	Son todos los procesos de insertar, modificar y eliminar un profesor.
Referencia	R12- Gestionar Profesor. R12.1- Insertar Profesor. R12.2- Modificar Profesor. R12.3- Eliminar Profesor.

CUS- 14	Gestionar Role.
Actor	Administrar.
Descripción	Son todos los procesos de insertar, modificar y eliminar un role de un usuario.
Referencia	R2- Gestionar usuarios del sistema de acuerdo a los niveles de acceso. R2.1- Insertar Role. R2.2- Modificar Role. R2.3- Eliminar Role.

CUS- 15	Gestionar Alumno Ayudante.
Actor	Administrar.

Descripción	Son todos los procesos de insertar, modificar y eliminar un alumno ayudante.
Referencia	R13- Gestionar Alumno Ayudante. R13.1- Insertar Alumno Ayudante. R13.2- Modificar Alumno Ayudante. R13.3- Eliminar Alumno Ayudante.

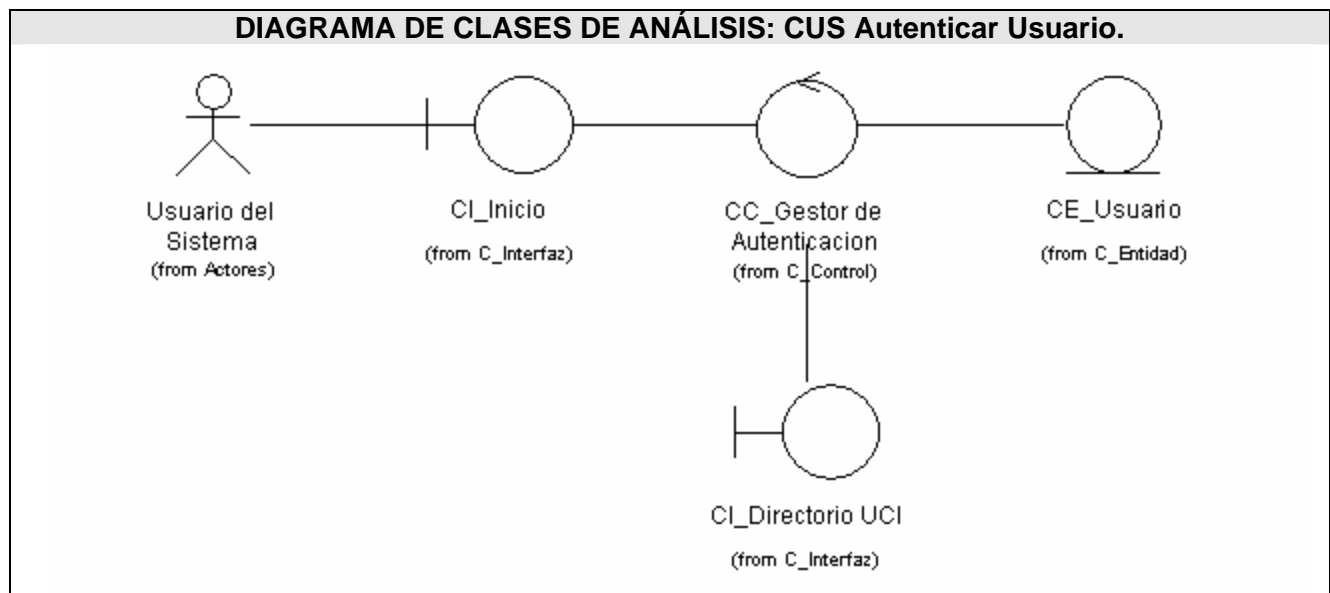
2.5.3- Diagrama de casos de uso del sistema.



CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

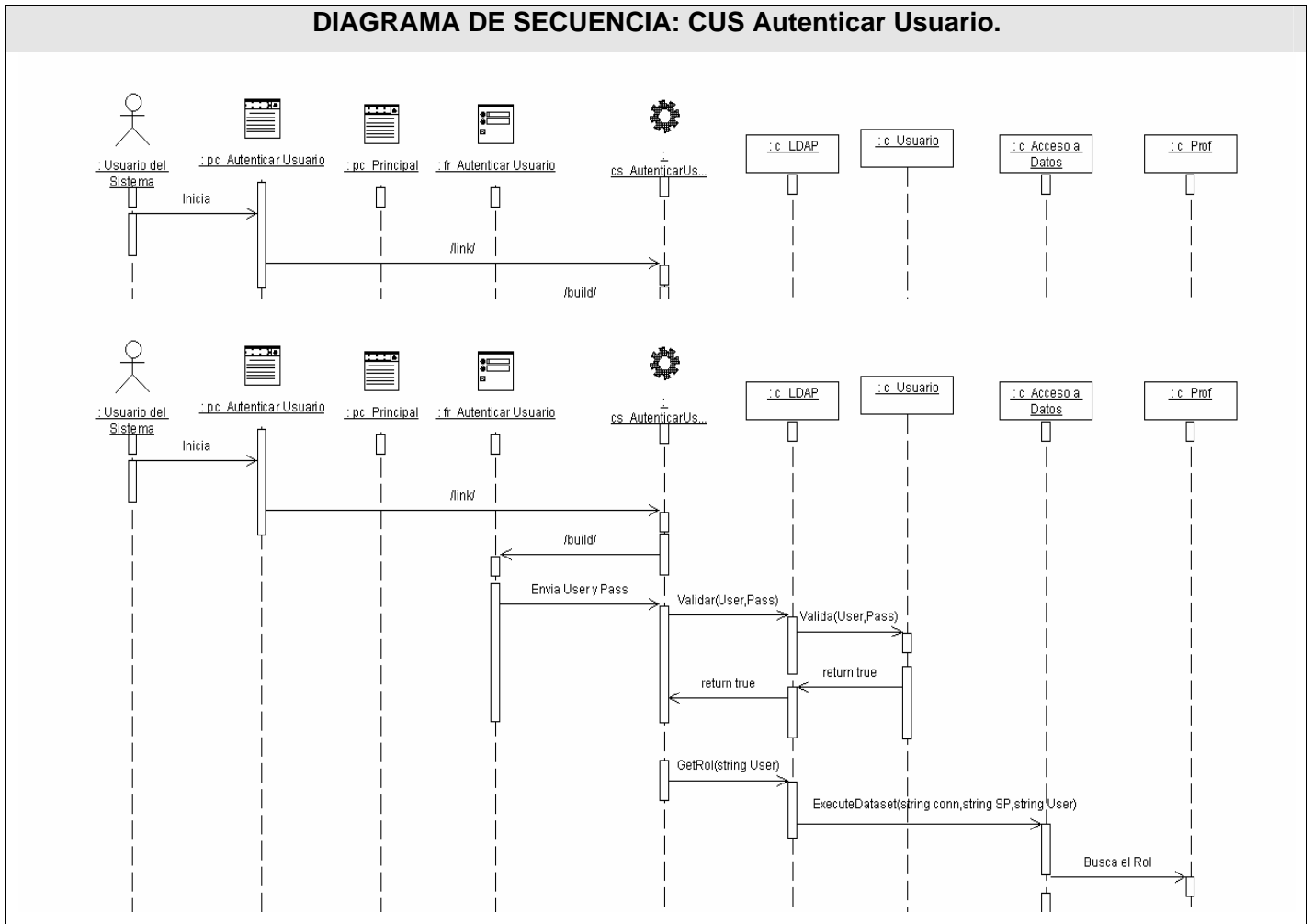
3.1- Análisis.

3.1.1- Diagrama de clases del Análisis.

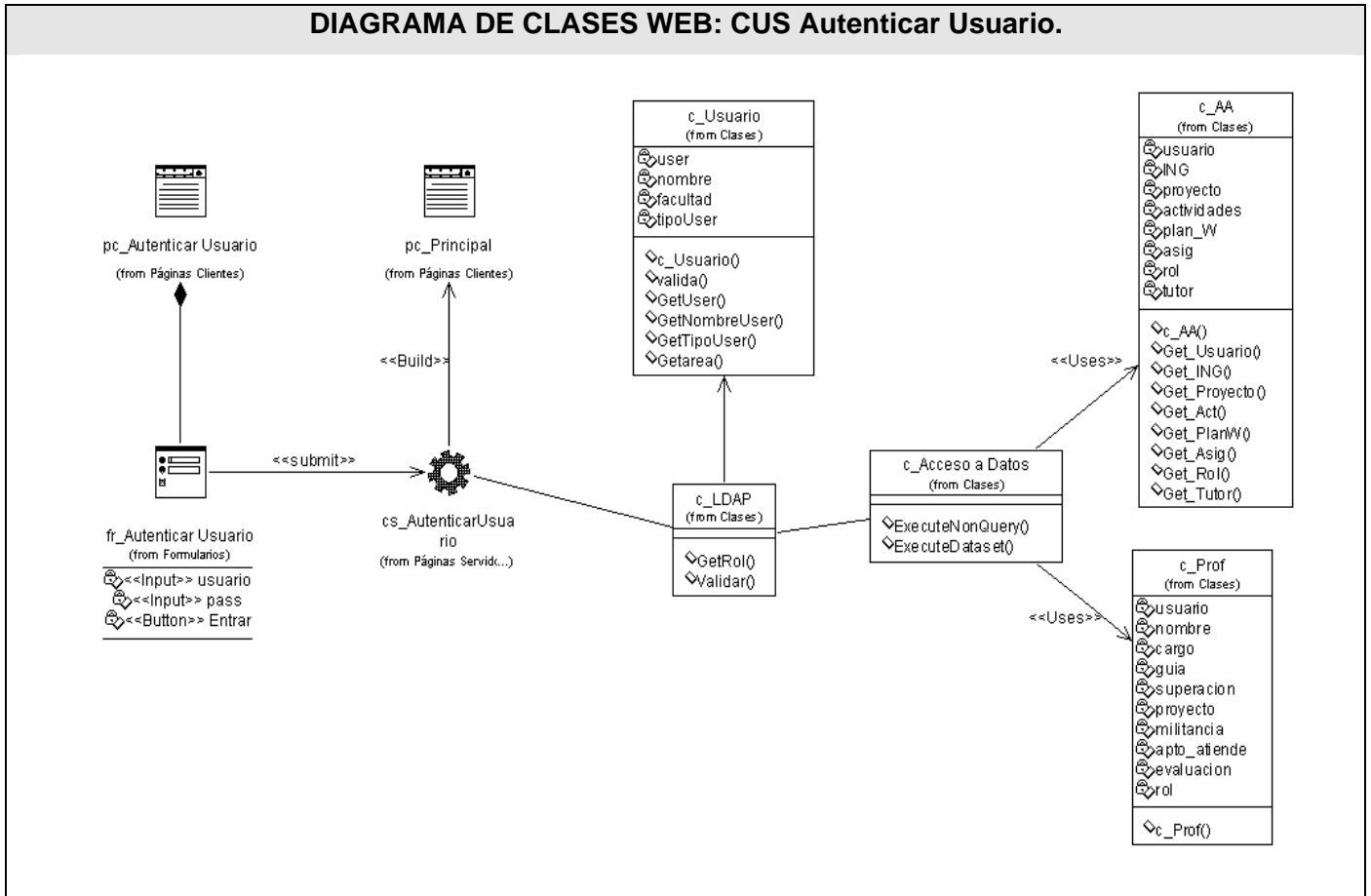


3.2- Diseño.

3.2.1- Diagramas de interacción.



3.2.2- Diagrama de clases web.



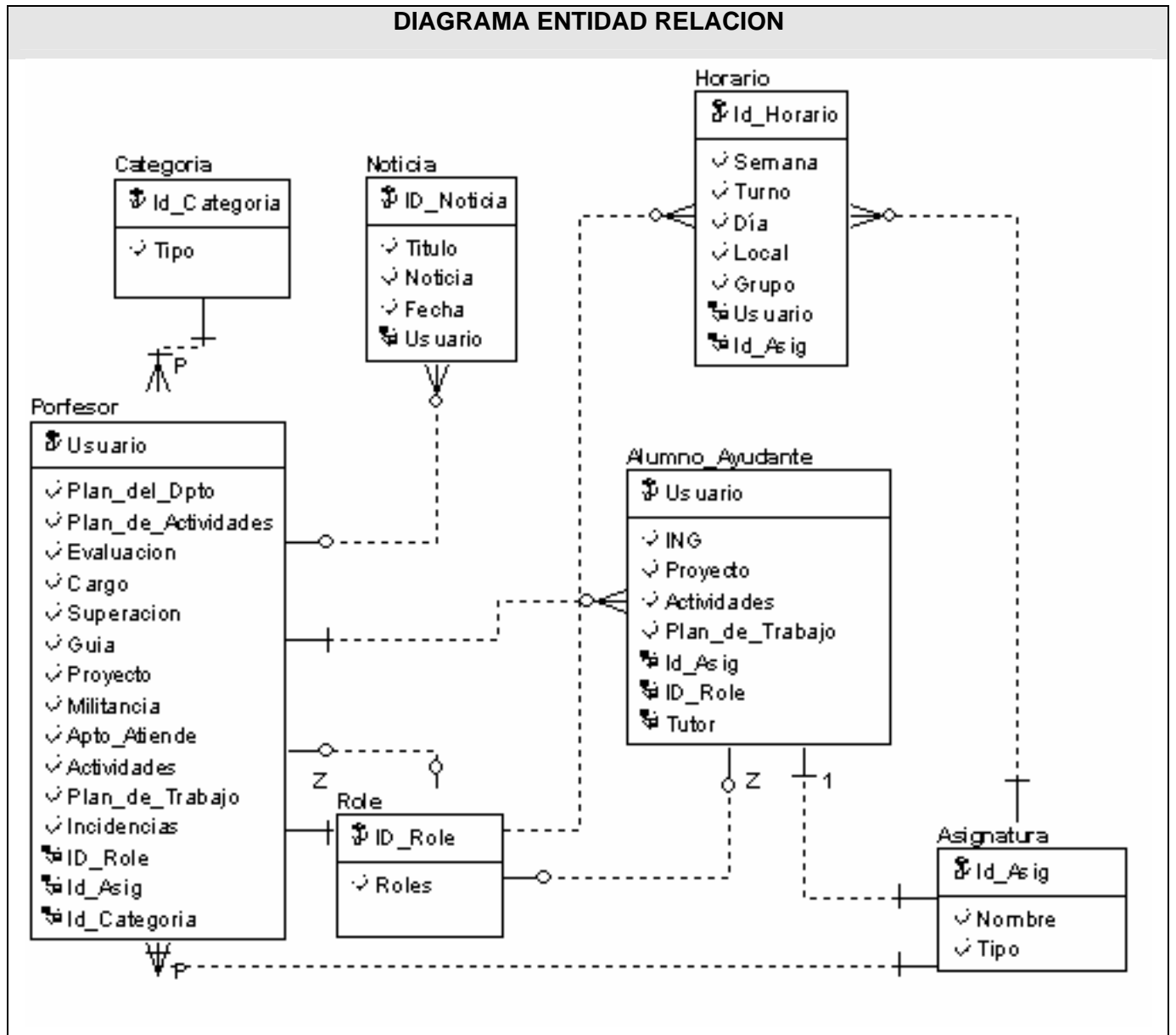
3.2.3- Descripción de las clases.

Nombre: c_LDAP	
Tipo de clase: Controladora	
Para cada responsabilidad:	
Nombre:	String Get_rol (string User)

Descripción:	Método que se le pasa un usuario y te devuelve su rol.
Nombre:	Bool Validar(string User, string Pass)
Descripción:	Método que valida el usuario y la contraseña.

3.3- Diseño de la Base de Datos.

3.3.1- Diagrama Entidad Relacional de la Base de Datos.



3.3.2- Descripción de las tablas.

Nombre: Profesor		
Descripción: Almacena los datos personales de los profesores, así como las informaciones que se encuentran ligadas al departamento así como sus obligaciones, entre otros.		
Atributo	Tipo	Descripción
Usuario	VARCHAR	Usuario de dominio
Plan_de_Dpto	TEXT	Permite conocer las tareas y actividades del Dpto.
Plan_de_Actividades	TEXT	Planificación de las actividades a realizar.
Evaluacion	VARCHAR	Criterio emitido por la labor desempeñada.
Cargo	VARCHAR	Rol que desempeña una persona dentro del Dpto.
Superacion	NVARCHAR	Vencimiento de ciertos niveles a través de pruebas.
Guia	BIT	Profesor que atiende un grupo.
Proyecto	NVARCHAR	Nombre del proyecto productivo al que pertenece.
Militancia	VARCHAR	Organización política a las que pertenezca.
Apto_Atiende	VARCHAR	Apto que atiende un profesor.
Actividades	TEXT	Tareas cumplidas que le fueron asignadas.
Plan_de_Trabajo	TEXT	Planificación de las tareas y

		actividades de un profesor.
Incidencias	TEXT	Incumplimientos de un profesor.

Nombre: Alumno Ayudante		
Descripción: Almacena los datos personales de los alumnos ayudantes, así como la información que lo une a un departamento, conjuntamente con sus obligaciones.		
Atributo	Tipo	Descripción
Usuario	VARCHAR	Usuario de dominio
Proyecto	NVARCHAR	Nombre del proyecto productivo al que pertenece.
Actividades	TEXT	Tareas cumplidas que le fueron asignadas.
Plan_de_Trabajo	TEXT	Planificación de las tareas y actividades de un profesor.

Nombre: Horario		
Descripción: Almacena los datos necesarios para hacer la planificación del proceso docente.		
Atributo	Tipo	Descripción
Semana	BIT	Indica si es semana par o impar.
Dia	VARCHAR	Días de la semana.
Local	BIT	Aula o Laboratorio donde se impartirán las clases
Grupo	VARCHAR	Número que indica el facultad, año y

		número de un grupo
--	--	--------------------

Nombre: Asignatura.		
Atributo	Tipo	Descripción
Nombre	VARCHAR	Nombre que identifica a la asignatura.
Tipo	VARCHAR	Rama en la que se agrupan las asignaturas.

Nombre: Role		
Descripción: Almacena los datos que le darán acceso a los recursos destinados según los permisos de los usuarios.		
Atributo	Tipo	Descripción
Roles	VARCHAR	

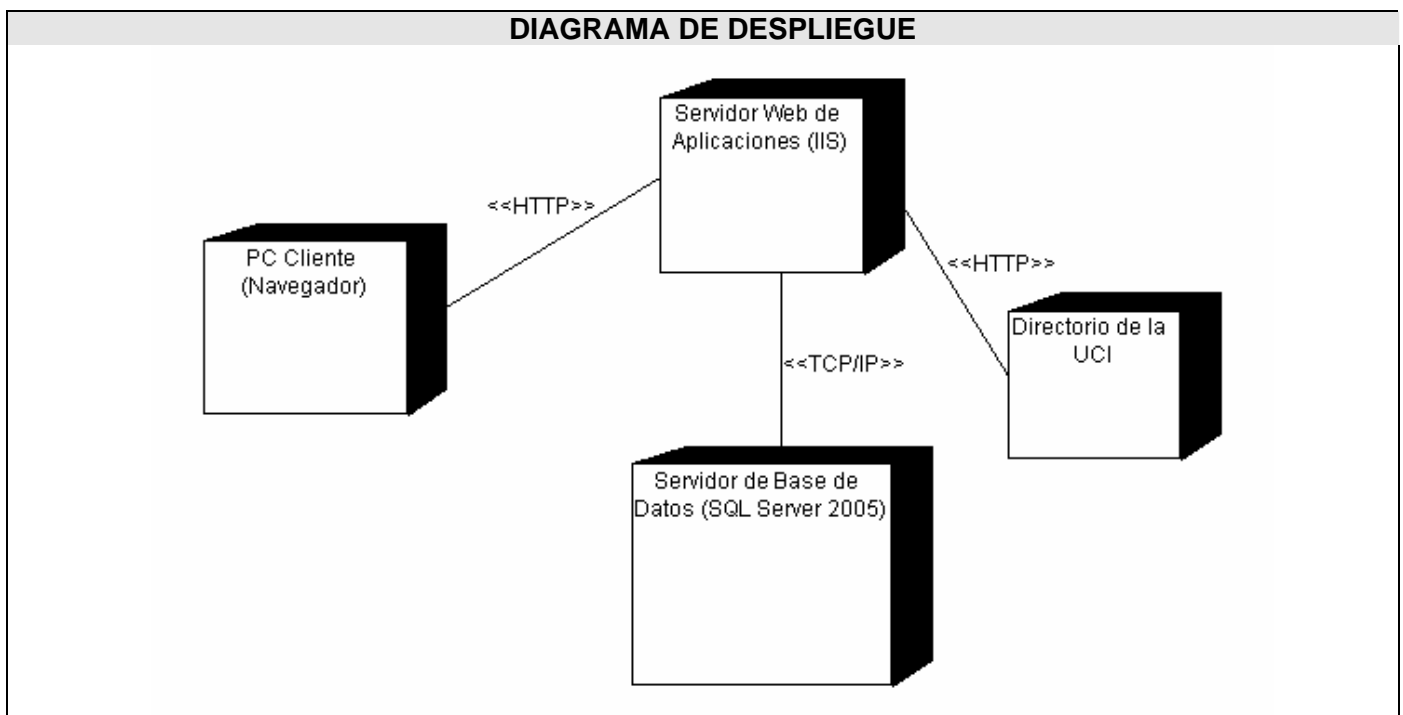
Nombre: Noticia		
Descripción: Almacena los datos de todas las noticias que se publicaran en el sistema.		
Atributo	Tipo	Descripción
Titulo	LONG VARCHAR	Es el título de la noticia
Noticia	TEXT	Cuerpo de la noticia.
Fecha	DATETIME	Fecha de publicación de la noticia.

Nombre: Categoría.		
Descripción: Almacena los datos e informaciones de todas las categorías docentes y científicas del departamento.		
Atributo	Tipo	Descripción
Tipo	VARCHAR	Almacena los tipos de categorías docentes.

CAPÍTULO 4: IMPLEMENTACIÓN Y PRUEBA

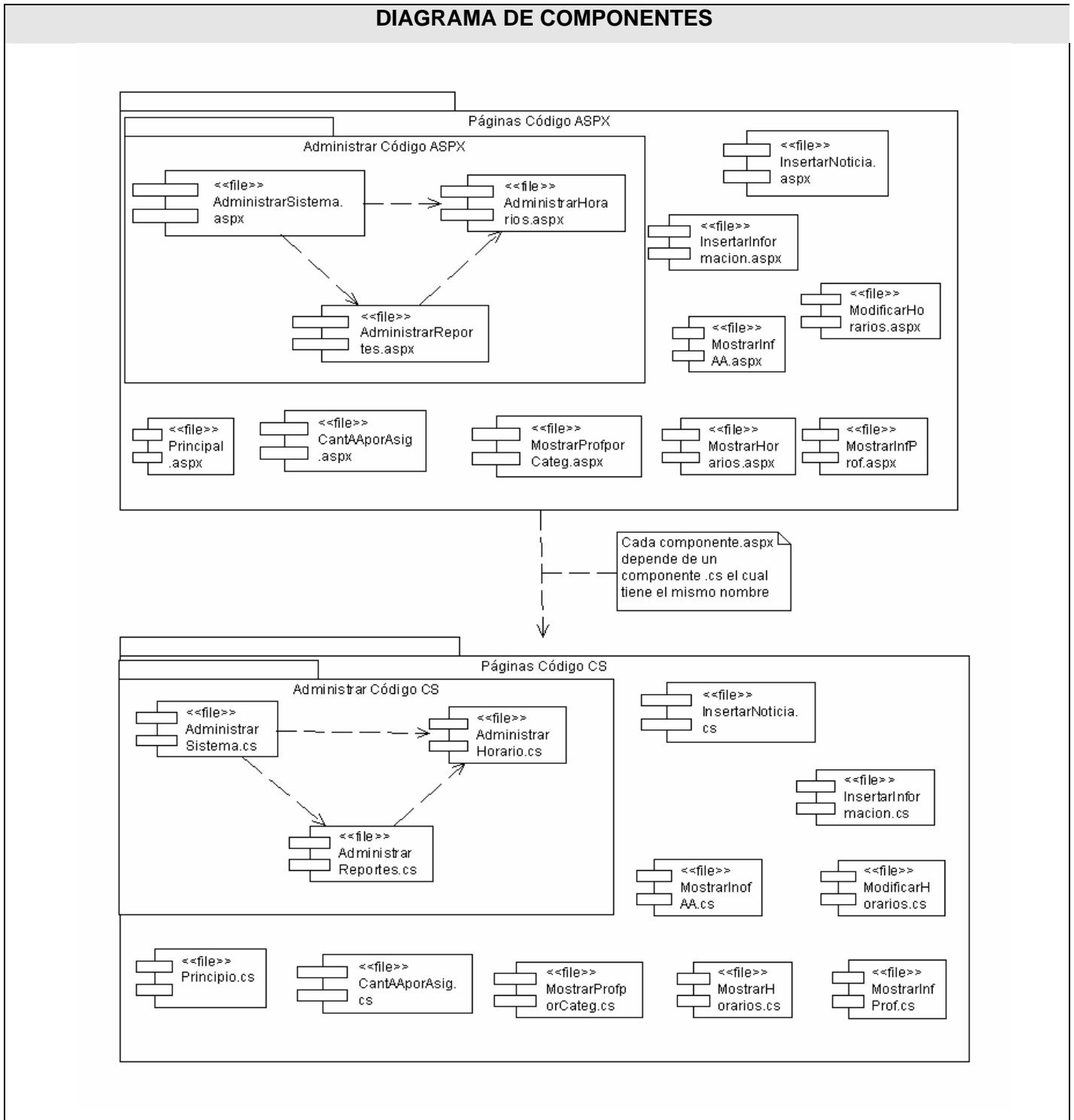
IMPLEMENTACION.

4.1- Diagrama de Despliegue.

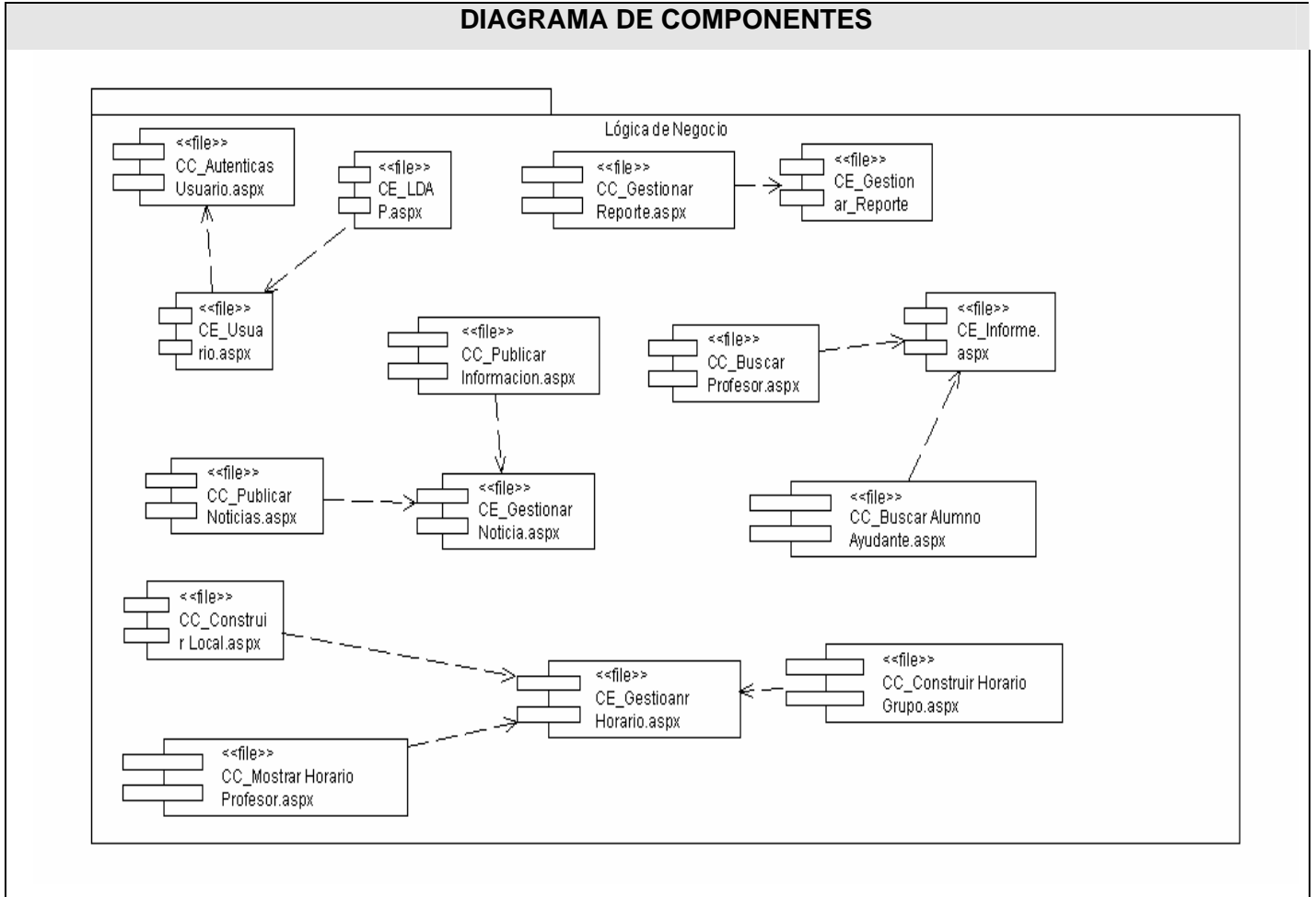


4.2- Diagrama de Componentes.

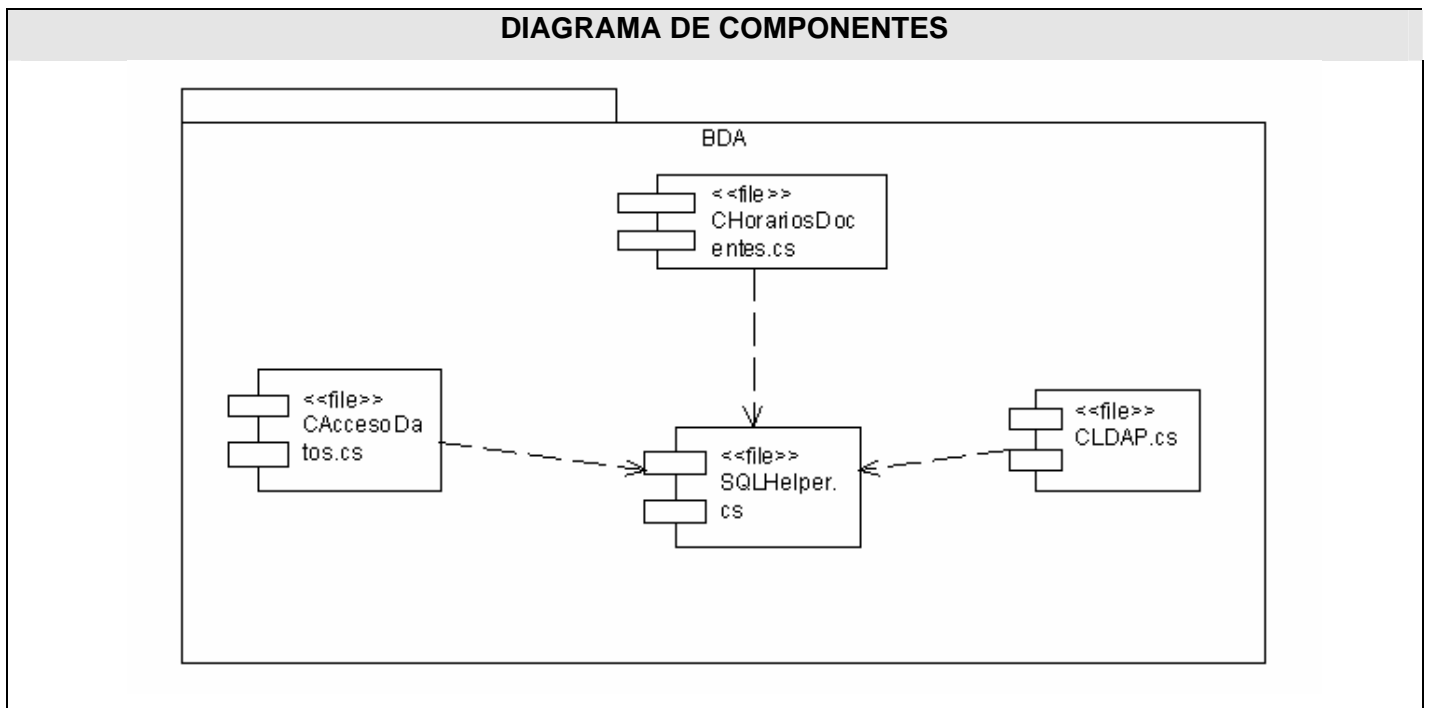
4.2.1- Paquete de Presentación.



4.2.2- Paquete de Entidades y Lógica del Negocio.



4.2.3- Paquete de Acceso a Datos.



PRUEBA.

Las pruebas de software son un elemento crítico para la garantía de calidad del software y representa una revisión final de las especificaciones, del diseño y de la codificación.(PRESSMAN 2002)

El modelo de pruebas describe principalmente como se prueban los componentes ejecutables en el modelo de implementación con pruebas de integración y de sistema. (JACOBSON *et al.* 2000)

Nombre del Caso de Uso: Autenticar en el Sistema.

Entrada	Resultados	Condiciones
Entrada Válida: Nombre: admin. Contraseña: admin	Accede al menú del administrador.	Debe dar acceso al menú de Administración
Entrada Válida: Nombre: vise-decana Contraseña: vise-decana	Accede al menú de los Horarios.	Debe dar acceso al menú de los Horarios.
Entrada Inválida: Nombre: proyecto Contraseña: proyecto	Muestra un mensaje diciendo que los datos son incorrectos.	Debe lanzar un mensaje dado que no es administrador ni usuario del sistema.
Entrada Inválida: Nombre: Contraseña : vise-decana	Muestra un mensaje diciendo que hay campos vacíos.	Debe mostrar un mensaje diciendo que faltan datos por entrar.
Entrada Inválida: Nombre: vise-decana Contraseña :	Muestra un mensaje diciendo que hay campos vacíos.	Debe mostrar un mensaje diciendo que faltan datos por entrar.

Nota: Se permiten hasta tres intentos con un mismo usuario, cuando sobrepase el número de intentos tendrá que tratar con otro usuario diferente al antes mencionado.

Entrada	Resultados	Condiciones
Entrada Válida: Nombre: Turno	Se muestra en el horario el turno insertado.	Debe insertarse, seleccionarse y visualizarse el turno seleccionado.
Entrada Válida: Nombre: Semana	Se muestra en el horario la semana insertada.	Debe insertarse, seleccionarse y visualizarse la semana seleccionada.
Entrada Válida: Nombre: día.	Se muestra en el horario el día de la semana insertado.	Debe insertarse, seleccionarse y visualizarse el día de la semana seleccionado.
Entrada Válida: Nombre: local.	Se muestra en el horario el local insertado.	Debe insertarse, seleccionarse y visualizarse el local seleccionado.
Entrada Válida: Nombre: grupo	Se muestra en el horario el grupo insertado.	Debe insertarse, seleccionarse y visualizarse el grupo seleccionado.
Entrada Válida: Nombre: asignatura.	Se muestra en el horario la asignatura insertada.	Debe insertarse, seleccionarse y visualizarse la asignatura seleccionada.
Entrada Inválida: Nombre: semana.	Se muestra un mensaje en el que se informe de que la semana ya fue insertada.	Debe mostrar un mensaje en el que se informe que la semana ya fue insertada anteriormente.
Entrada Inválida: Nombre: día.	Se muestra un mensaje en el que se informe de que ese día ya fue insertado en esa semana.	Debe mostrar un mensaje de que el día ya fue insertado en esa semana.
Entrada Inválida: Nombre: local.	Se muestra un mensaje en el que se informe que ese local se encuentra ocupado por otro	Debe mostrar un mensaje de que el local insertado ya esta ocupado por otro grupo.

	grupo.	
--	--------	--

Nombre del Caso de Uso: Realizar Reporte, sección Buscar Información de Profesor.

Entrada	Resultados	Condiciones
Entrada Válida: Nombre: Nombre	Se muestran los datos del profesor seleccionado.	Debe seleccionar el profesor.

Nombre del Caso de Uso. Confeccionar Horario, sección Mostrar Horario de un grupo.

Entrada	Resultados	Condiciones
Entrada Válida: Nombre: Nombre	Se muestran el horario de las dos semanas del grupo seleccionado.	Debe seleccionar el grupo.

Nombre del Caso de Uso. Confeccionar Horario, sección Mostrar Horario de Aula.

Entrada	Resultados	Condiciones
Entrada Válida: Nombre: aula.	Se muestran el horario de las dos semanas del aula seleccionado.	Debe seleccionar el aula.

Nombre del Caso de Uso. Confeccionar Horario, sección Insertar Horario Docente.

Entrada	Resultados	Condiciones
Entrada Válida: Nombre: Turno	Se muestra en el horario el turno insertado.	Debe insertarse, seleccionarse y visualizarse el turno seleccionado.
Entrada Válida: Nombre: Semana	Se muestra en el horario la semana insertada.	Debe insertarse, seleccionarse y visualizarse la semana seleccionada.
Entrada Válida: Nombre: día.	Se muestra en el horario el día de la semana insertado.	Debe insertarse, seleccionarse y visualizarse el día de la semana seleccionado.
Entrada Válida: Nombre: local.	Se muestra en el horario el local insertado.	Debe insertarse, seleccionarse y visualizarse el local seleccionado.
Entrada Válida: Nombre: grupo	Se muestra en el horario el grupo insertado.	Debe insertarse, seleccionarse y visualizarse el grupo seleccionado.
Entrada Válida: Nombre: asignatura.	Se muestra en el horario la asignatura insertada.	Debe insertarse, seleccionarse y visualizarse la asignatura seleccionada.
Entada Inválida: Nombre: turno.	Se muestra un mensaje en el que se informe que el turno insertado coincide con otro ya insertado el mismo día.	Debe mostrar un mensaje donde se le informe que ese turno ya fue insertado anteriormente ese mismo día.

Entrada Inválida: Nombre: semana.	Se muestra un mensaje en el que se informe de que la semana ya fue insertada.	Debe mostrar un mensaje en el que se informe que la semana ya fue insertada anteriormente.
Entrada Inválida: Nombre: día.	Se muestra un mensaje en el que se informe de que ese día ya fue insertado en esa semana.	Debe mostrar un mensaje de que el día ya fue insertado en esa semana.
Entrada Inválida: Nombre: local.	Se muestra un mensaje en el que se informe que ese local se encuentra ocupado por otro grupo.	Debe mostrar un mensaje de que el local insertado ya esta ocupado por otro grupo.

Nombre del Caso de Uso. Insertar Plan de Trabajo.

Entrada	Resultados	Condiciones
Entrada Válida: Nombre: plan de trabajo.	Se muestran el plan de trabajo seleccionado.	Debe seleccionar el profesor.

CONCLUSIONES GENERALES

Después de estudiar y analizar el problema se ha llegado al resultado que se consolidó con la implementación de una aplicación web sólida, confiable y eficiente que fue construida siguiendo los estándares web de punta, que automatiza el funcionamiento del departamento especialidad de la facultad 4 y que de seguro contribuirá a elevar la calidad del desarrollo del trabajo en el departamento.

Se analizaron diferentes propuestas de tecnologías y herramientas para tratar de hallar la más idónea para solucionar el problema que dió origen a este trabajo y se concluyó que el sistema debería ser desarrollado Visual Studio .NET 2005, en el lenguaje de C#, por ser la que mas adecuaba sus soluciones a las condiciones de la propuesta realizada, como son las limitantes de tiempo el cual con las diferentes componentes era considerablemente reducido, de la integración con el gestor de base de datos seleccionado que resultó ser SQL Server 2005 con el que no era necesario realizar configuraciones previas porque se integra muy bien a la plataforma de programación escogida, además de contar con un número de mejoras en lo que a seguridad y rendimiento se refieren que resultaron muy a tener en cuenta, por los aportes al desarrollo del trabajo.

Para el modelado y entendimiento de los procesos del negocio se utilizó la metodología RUP, desarrollada a través de la plataforma Rational Suite 2003 es por sus características la que más beneficios aportaría al desarrollo rápido y eficiente de la solución, con sus iteraciones se irían descubriendo nuevas funcionalidades que poco a poco se irían incrementando hasta llegar a la solución final, ayudó a comprender y emplear mejor la arquitectura escogida, entre otras cosas que permitieron una más sencilla y eficaz implementación.

Por lo que se considera que se han cumplido los objetivos planteados con el fin de dar solución al problema que originó el surgimiento de este trabajo.

RECOMENDACIONES

Para futuras versiones del software debe hacerse un aprovechamiento más intensivo de los **Web Service** de la universidad para hacer más sencillo el trabajo y no tener información que se encuentra en otro sistema, duplicada en la base de datos, tomando solo la necesaria, así como contemplar esta experiencia como un punto de partida para futuras soluciones más profundas y estándares del tema de la gestión de la información de un departamento docente aprovechando los más novedosos y nuevos estándares de desarrollo de la web.

BIBLIOGRAFÍA

CONSULTADAS:

C# Language Especifications. 2001. USA. Microsoft Corporation.

C# .NET Web Developer's Guide. USA.

Autores:

Adrian Turtschi

DotThatCom.com

Jason Werry

Greg Hack

Joseph Albahari

Saurabh Nandu Technical Editor

Wei Meng Lee Series Editor

Object Oriented Programming in C#. 2002 USA. y Object-Oriented C# Level 2.

Autores:

Robert J. Oberg

Howard Lee Harkness

ASP.NET 2.0, Nuevas Características.

Disponible en:

<http://www.microsoft.com/spanish/msdn/articulos/archivo/100904/voices/SecFeatNT2.asp>

Proyecto MONO para Sistemas Libres.

Disponible en:

http://www.mono-project.com/Main_Page

CITADAS:

JACOBSON, I.; G. BOOCH, *et al.* *El Proceso Unificado de Desarrollo de Software*. Addison Wesley., 2000.

438 p.

LARMAN, G. *UML y Patrones. Introducción al análisis y diseño orientado a objetos*. 2000. 507 p.

PRESSMAN, R. *Ingeniería de Software. Un enfoque práctico*, 2002.

Lenguajes de programación del lado del servidor:

Disponible en:

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

ANEXOS

ANEXOS I. Diagrama de clases del Análisis.

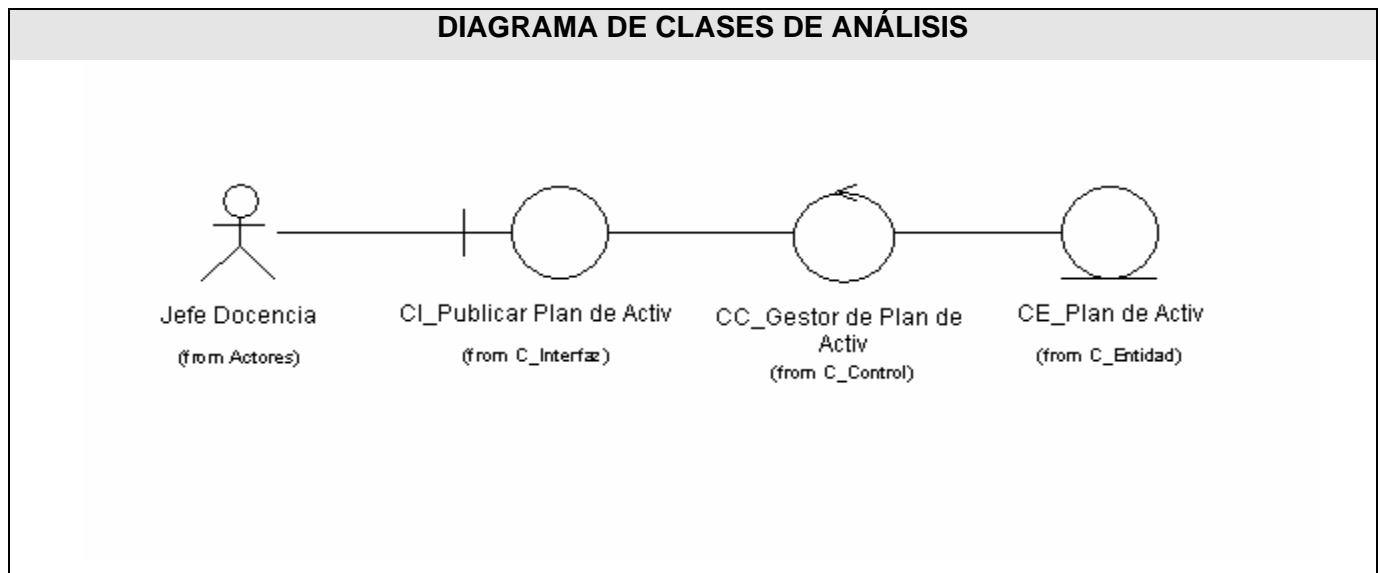


Figura 1. DCA. CUS Publicar Plan de Actividades.

DIAGRAMA DE CLASES DEL ANÁLISIS

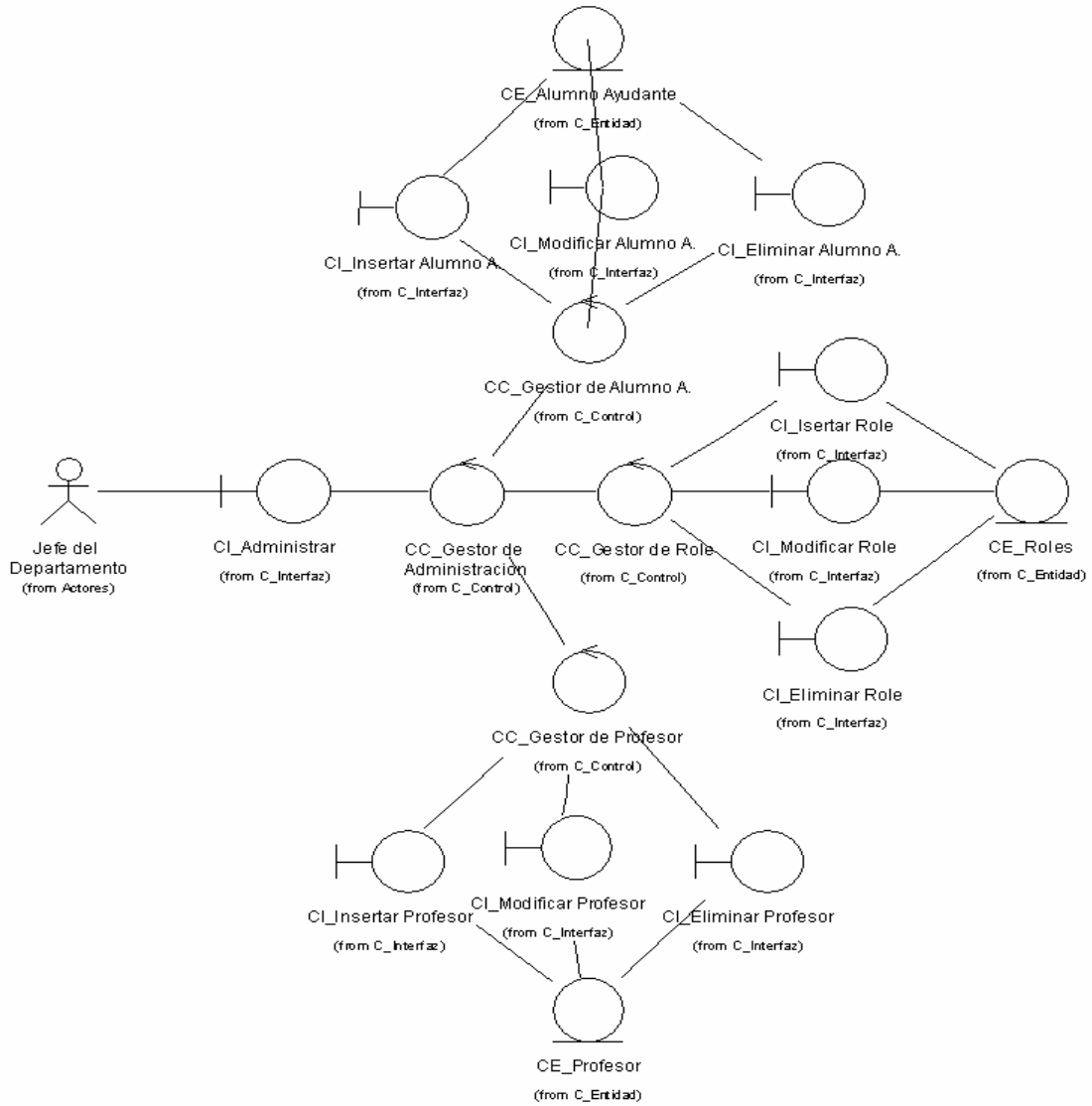


Figura 2. DCA. CUS Administrar.

DIAGRAMA DE CLASES DEL ANÁLISIS

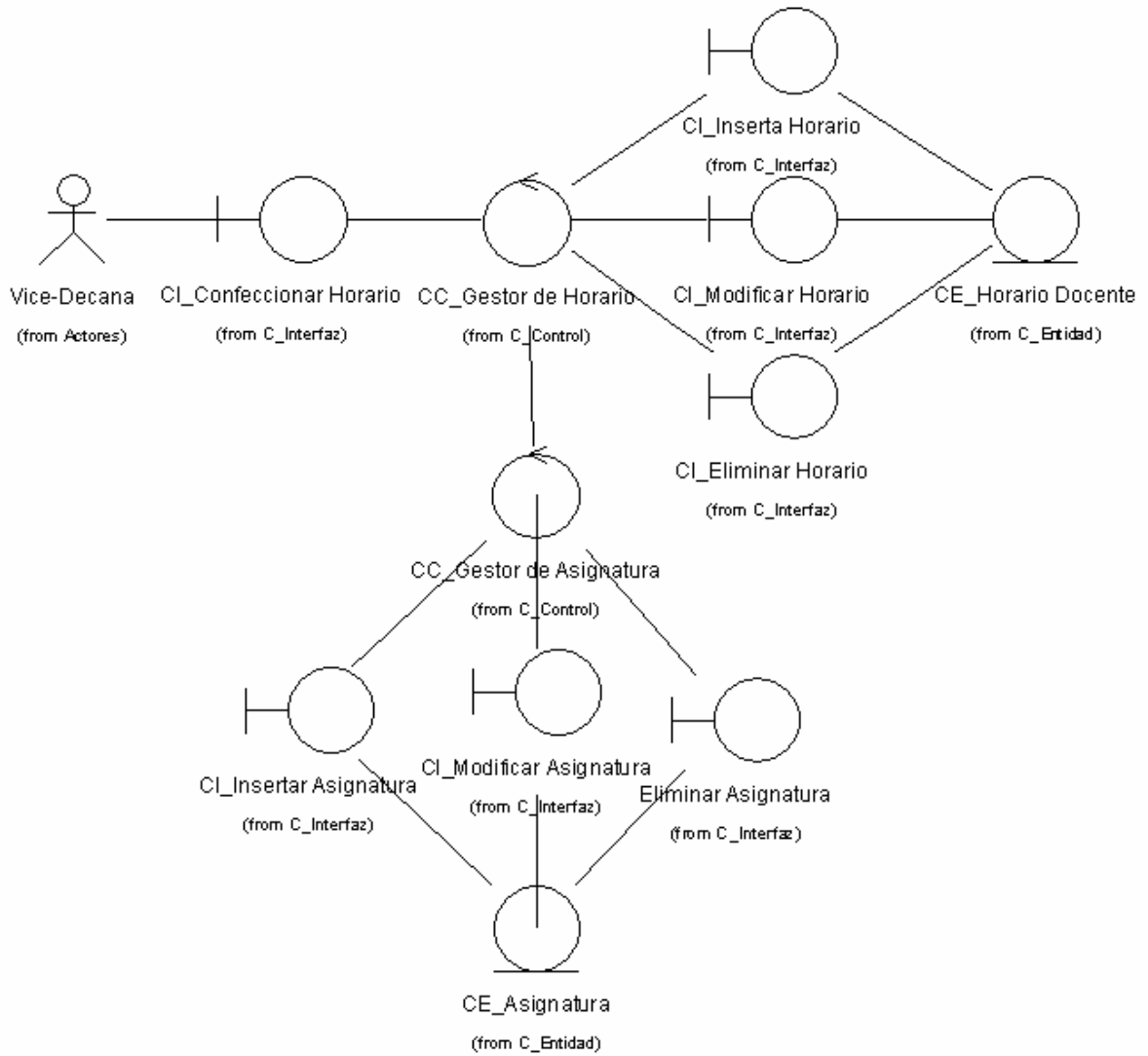


Figura 3. DCA. CUS Confeccionar Horario.

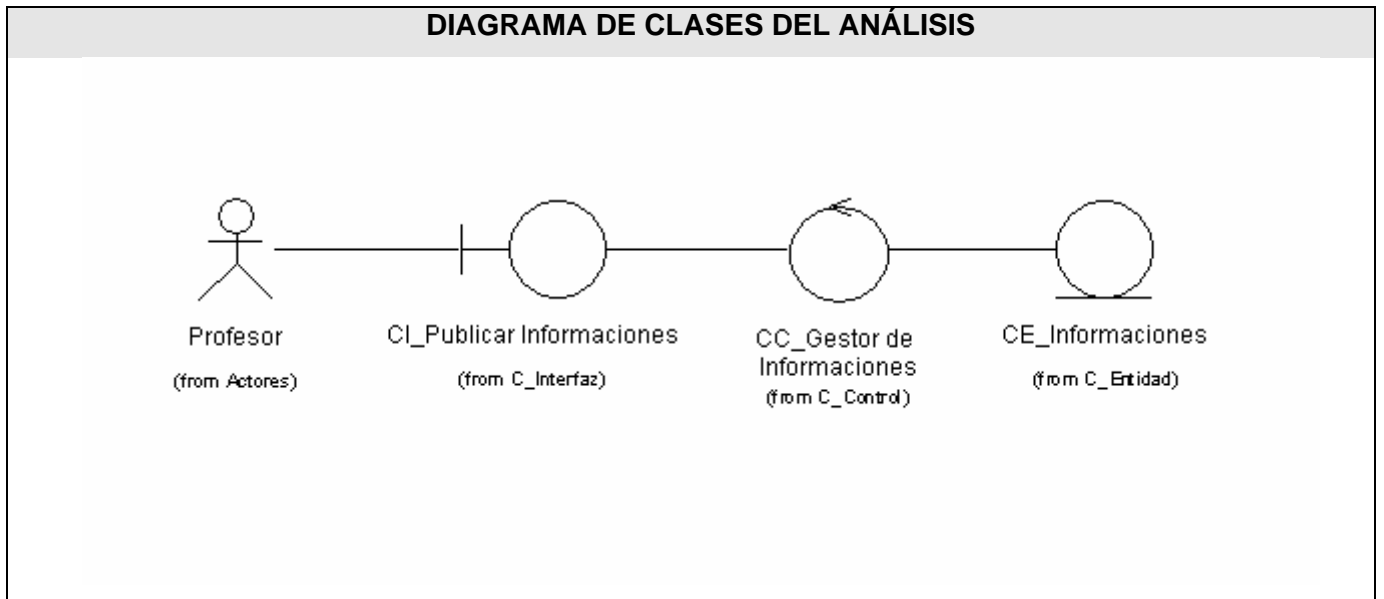


Figura 4. DCA. CUS Publicar Información.

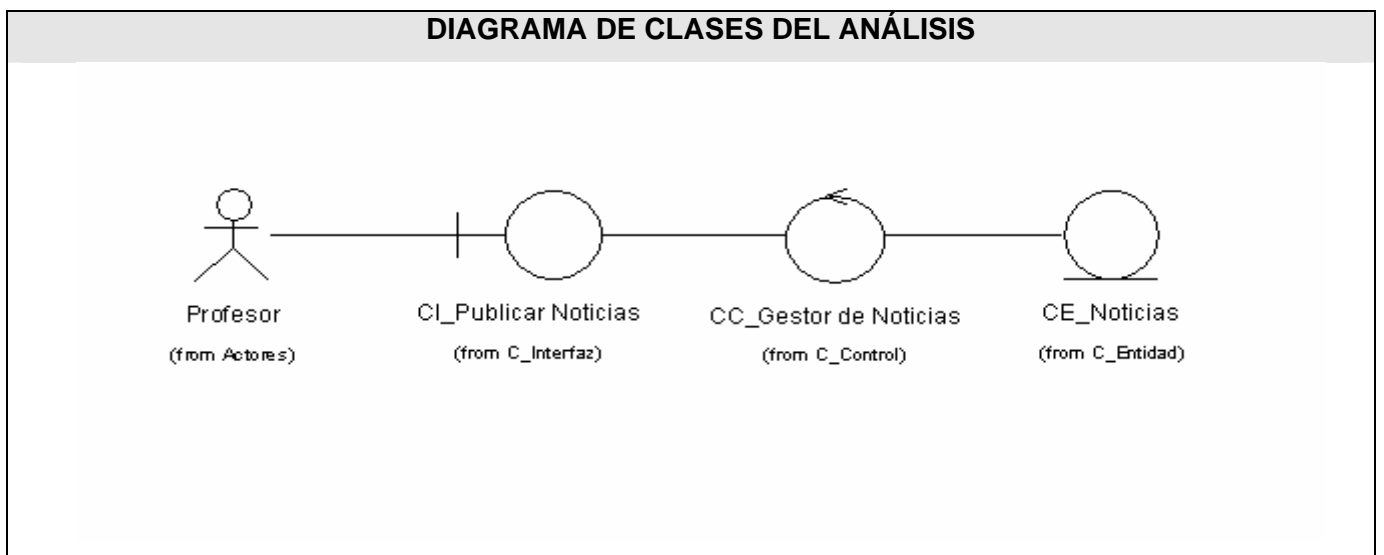


Figura 5. DCA. CUS Publicar Noticia.

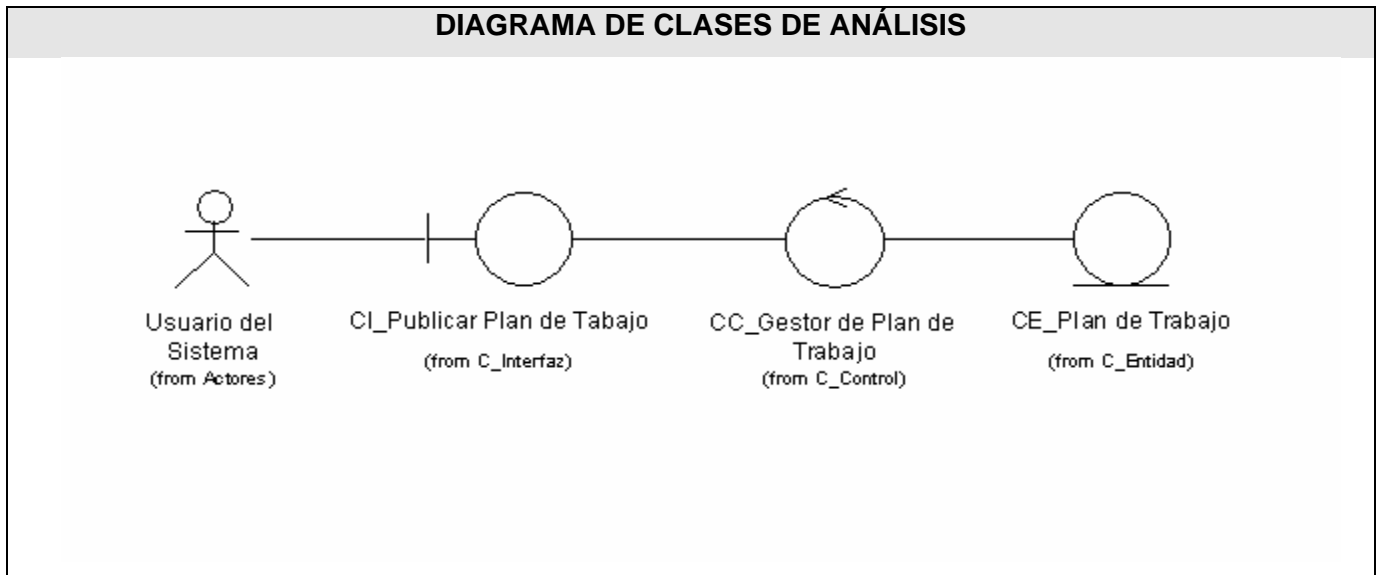


Figura 6. DCA. CUS Publicar Plan de Trabajo.

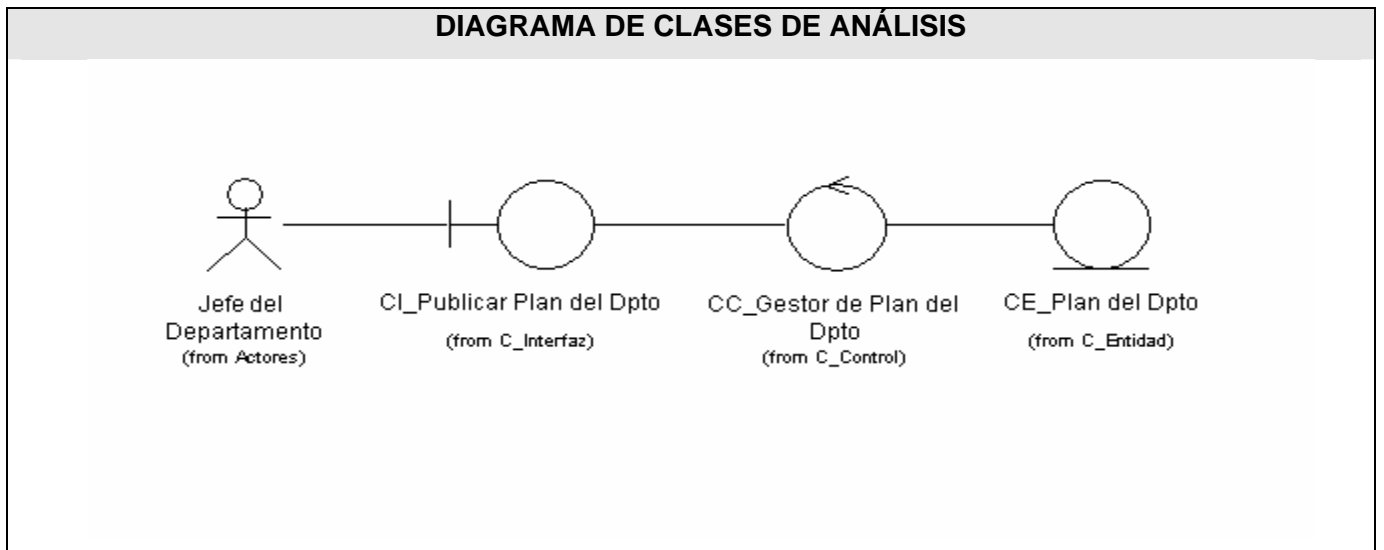


Figura 7. DCA. CUS Publicar Plan del Departamento.

DIAGRAMA DE CLASES DE ANÁLISIS

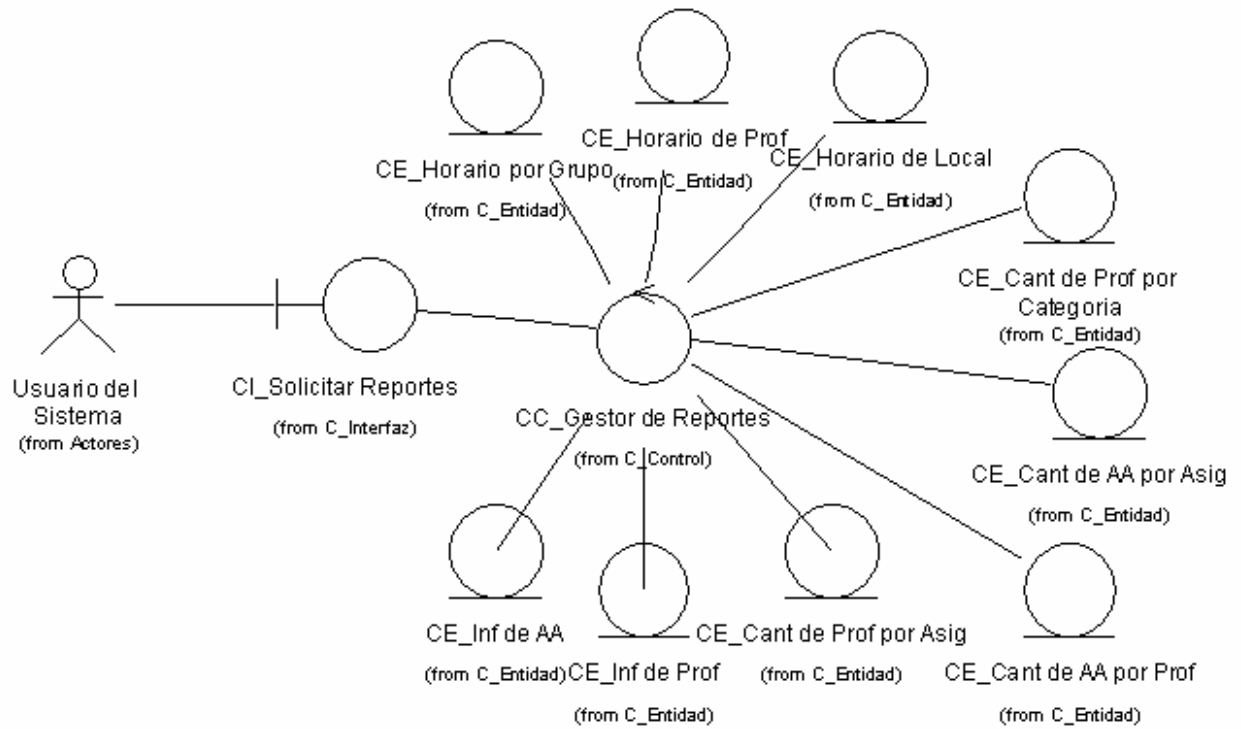


Figura 8. DCA. CUS. Realizar Reporte.

ANEXOS II Diagramas de Secuencia del Diseño.

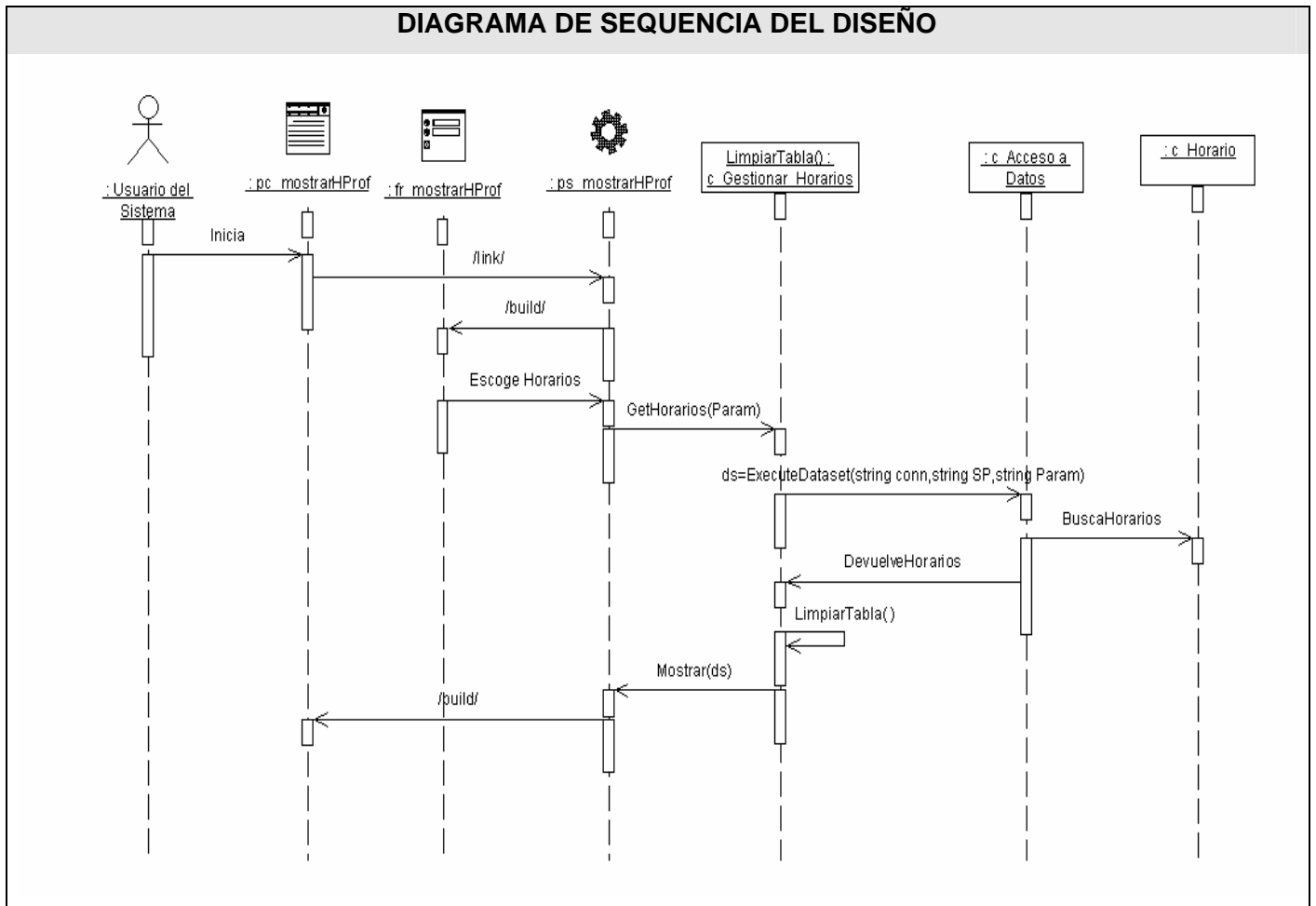


Figura 9. DSD. CUS Mostrar Horario.

DIAGRAMA DE SEQUENCIA DEL DISEÑO

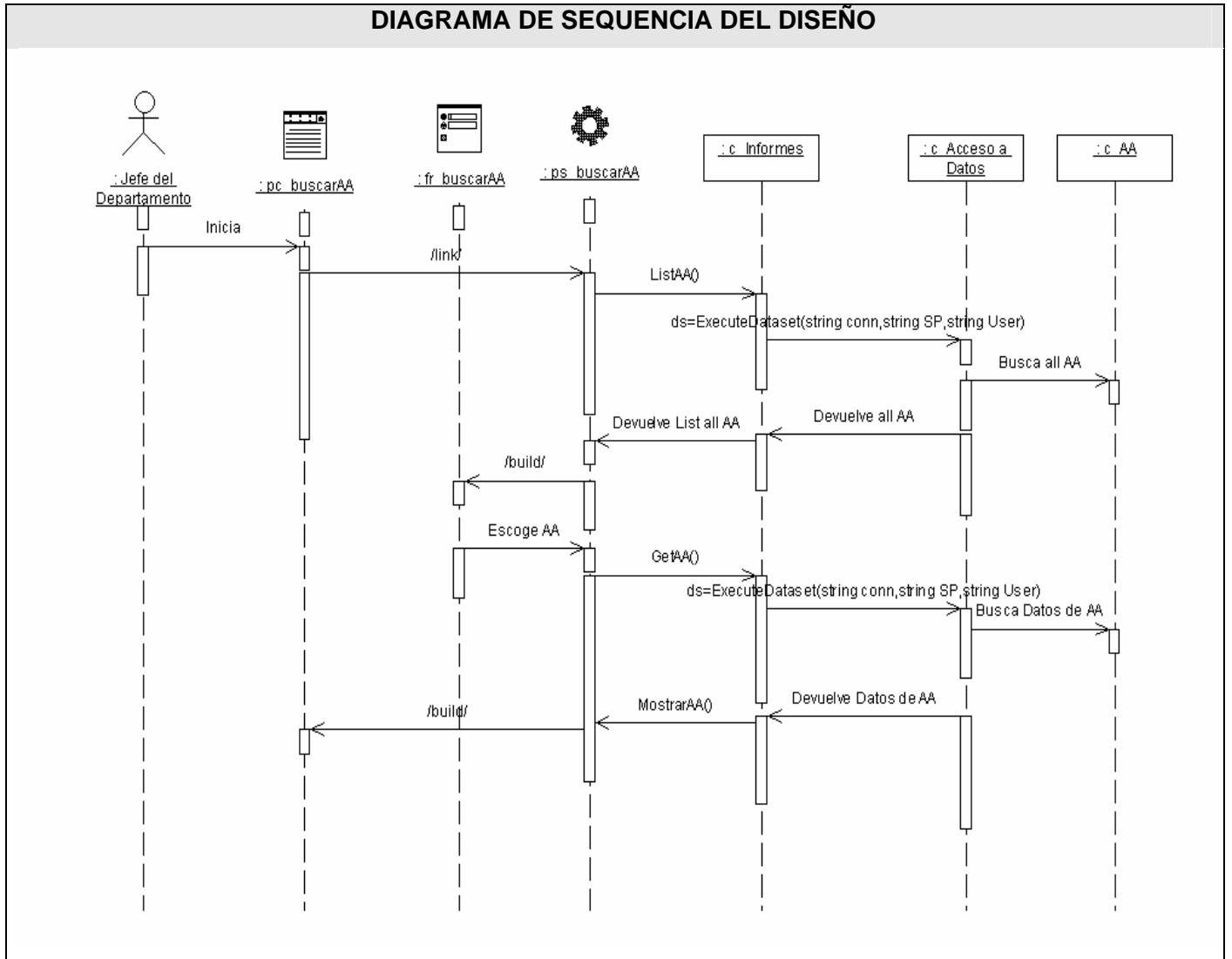


Figura 10. DSD. CUS Mostrar Información de Alumno Ayudante.

DIAGRAMA DE SEQUENCIA DEL DISEÑO

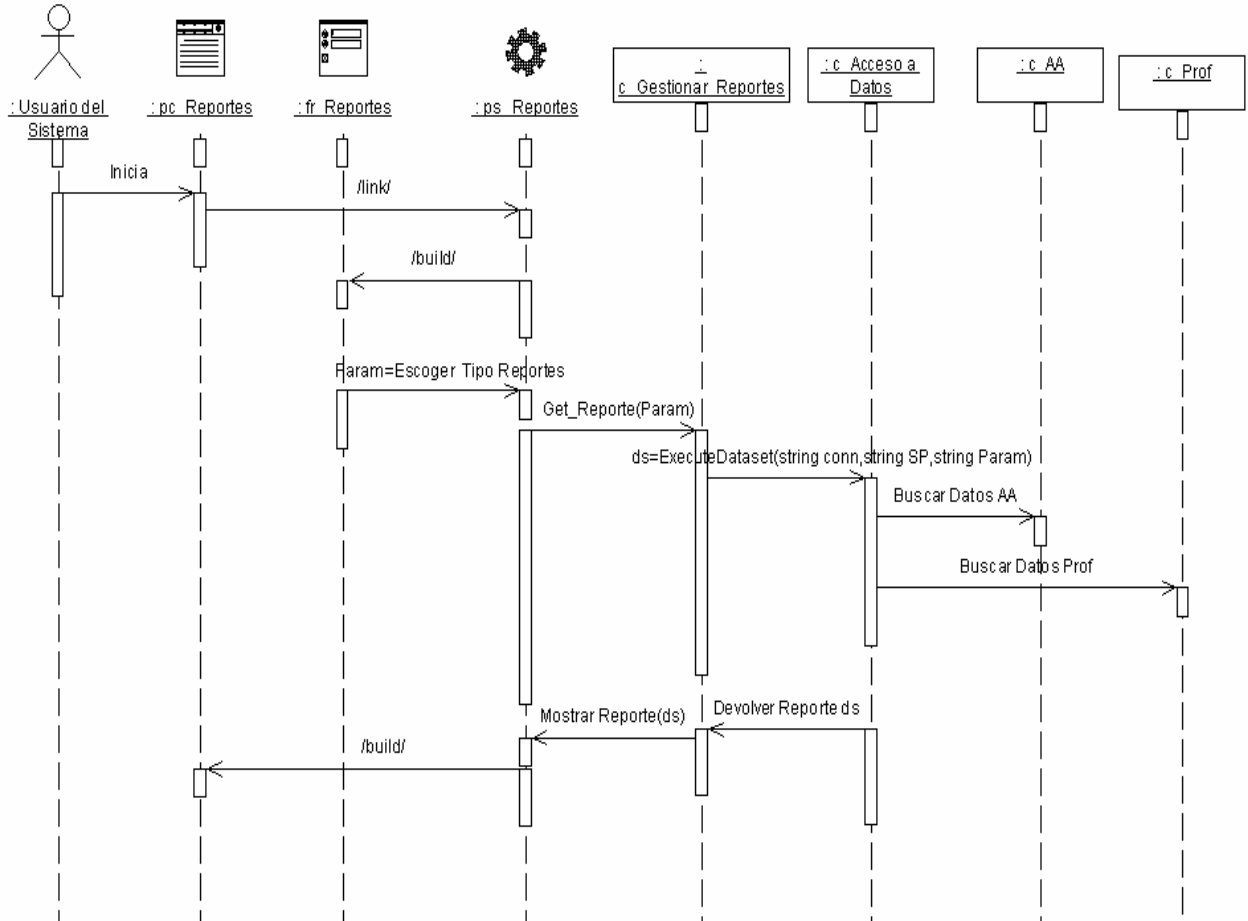


Figura 11. DSD. CUS Mostar Reporte.

ANEXOS II Diagramas de Clases del Diseño Web.

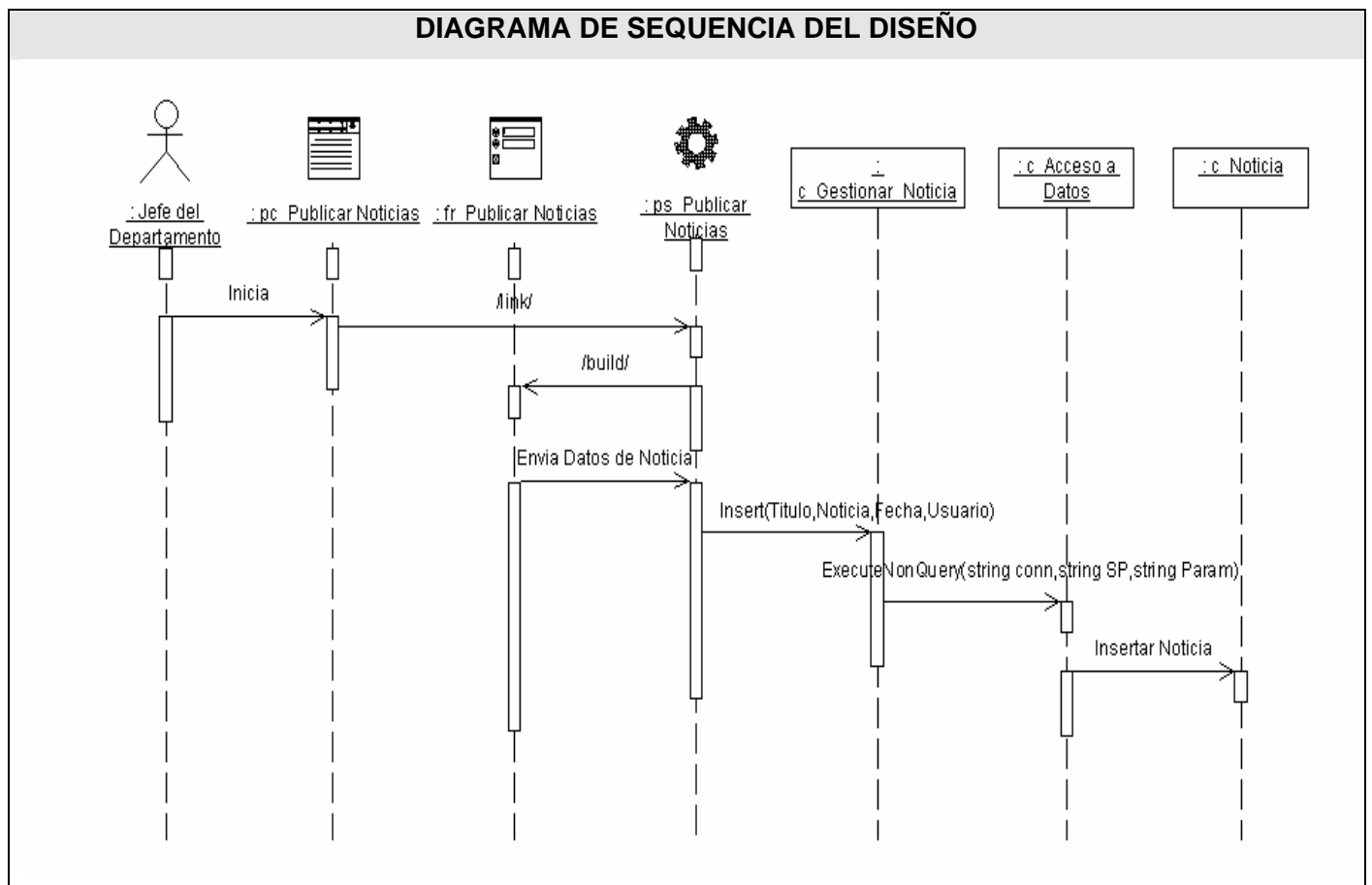


Figura 12. DSD. CUS Publicar Noticia.

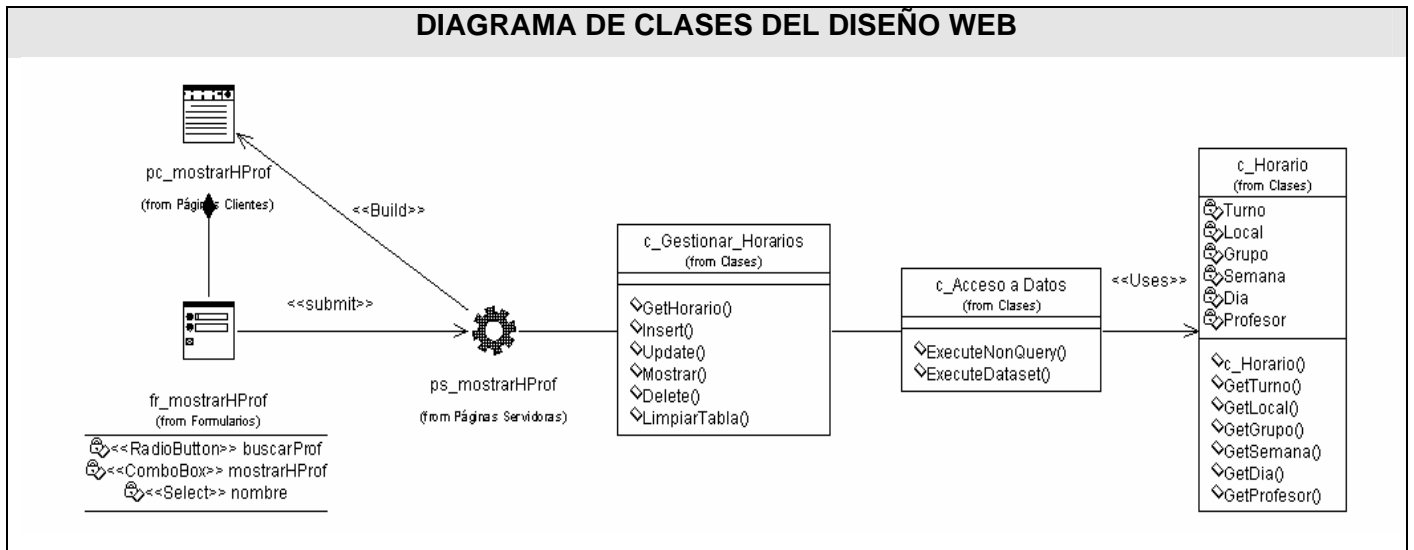


Figura 13. DCD. CUS Mostrar Horario del Profesor.

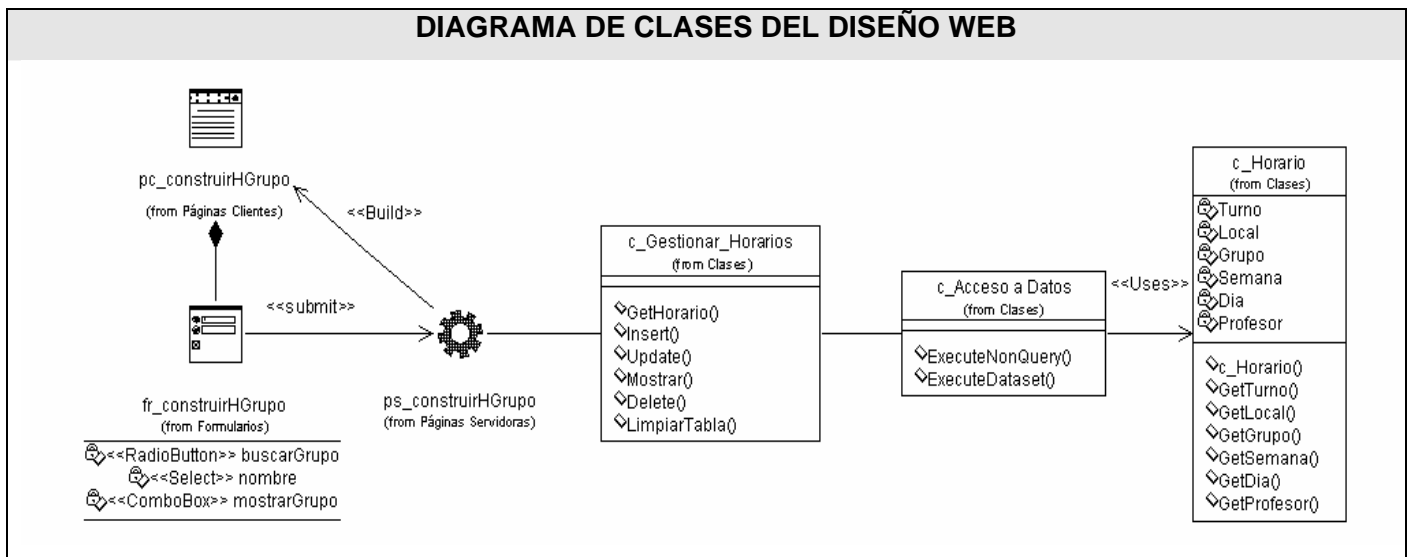


Figura 14. DCD. CUS Mostrar Horario de Grupo.

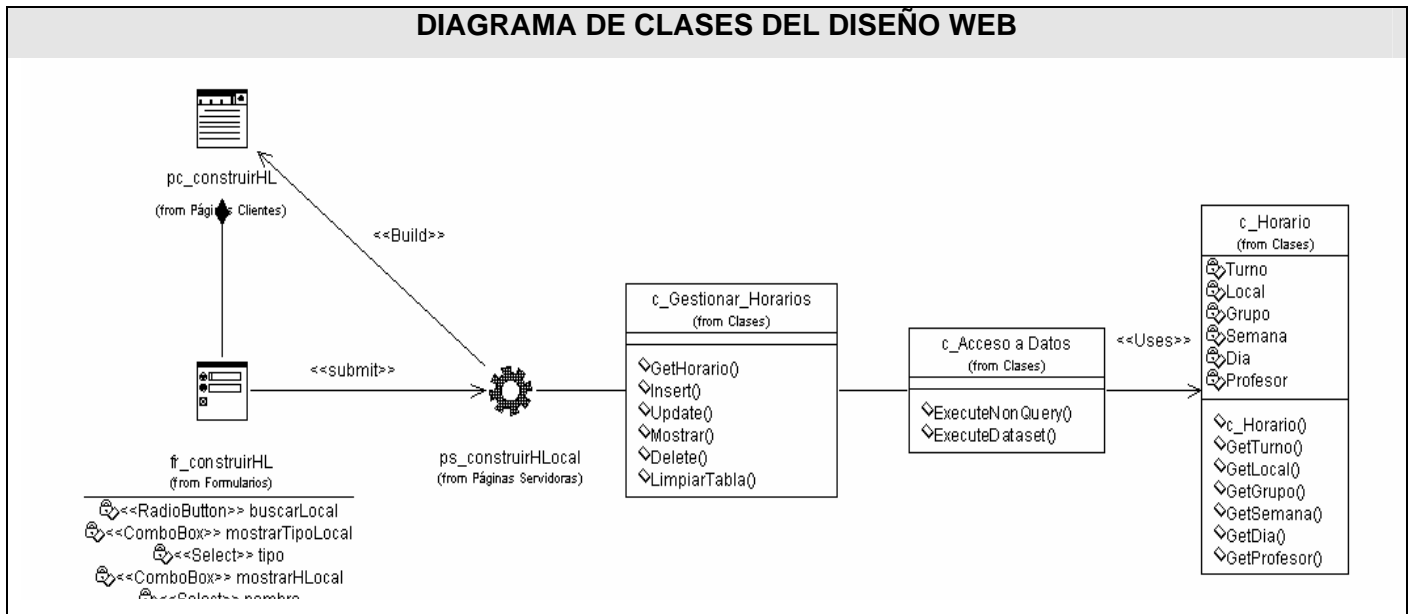


Figura 15. DCD. CUS. Mostar Horario de Local.

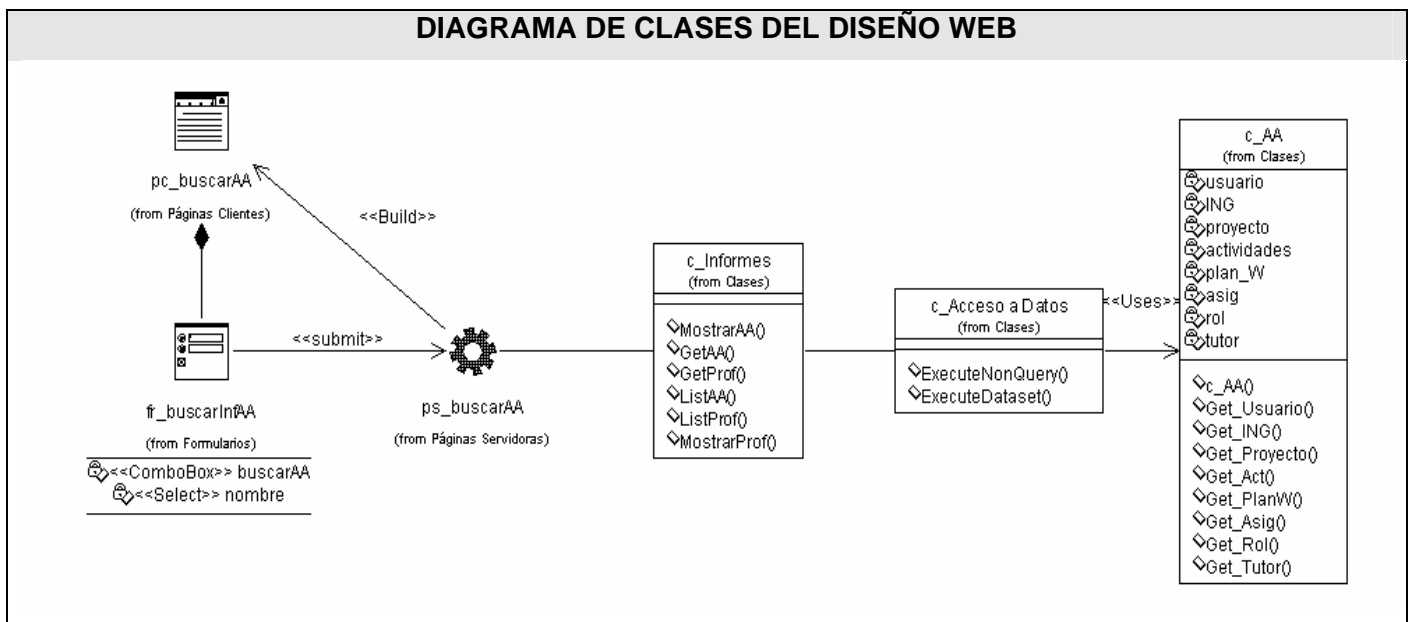


Figura 16. DCD. CUS. Mostrar Información de Alumno Ayudante.

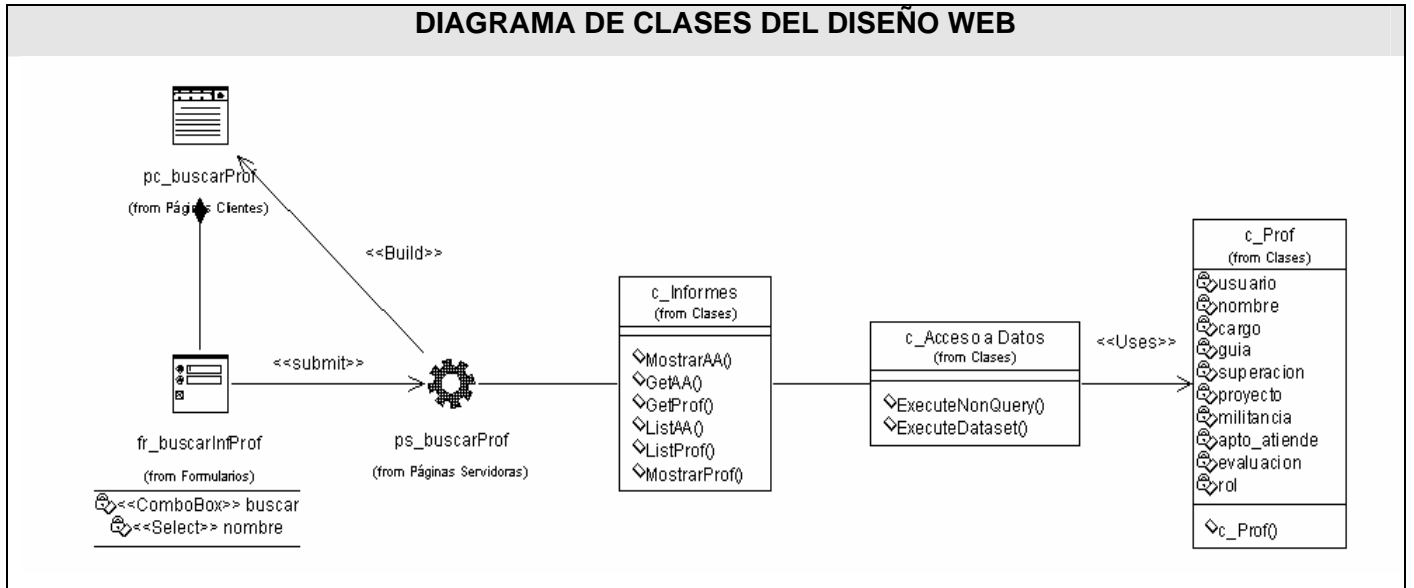


Figura 17. DCD. CUS. Mostrar Información del Profesor.

DIAGRAMA DE CLASES DEL DISEÑO WEB

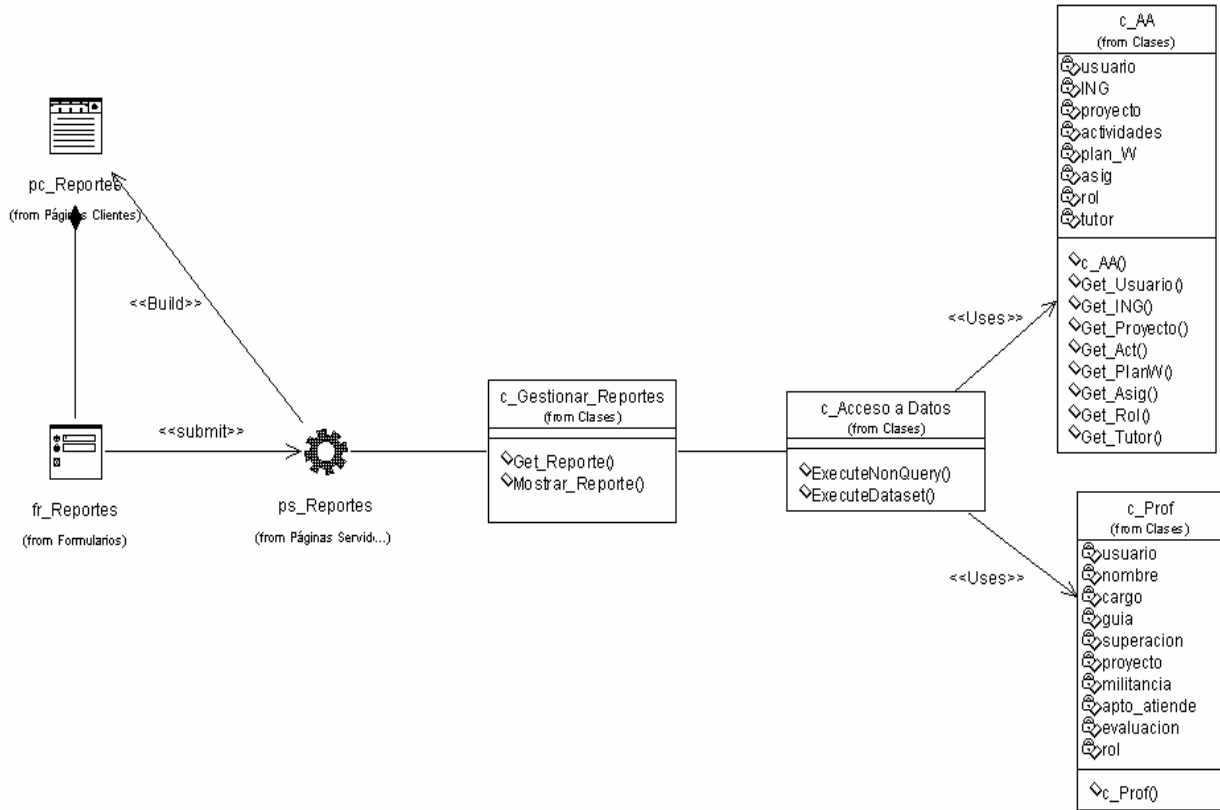


Figura 18. DCD. CUS. Mostrar Reporte.

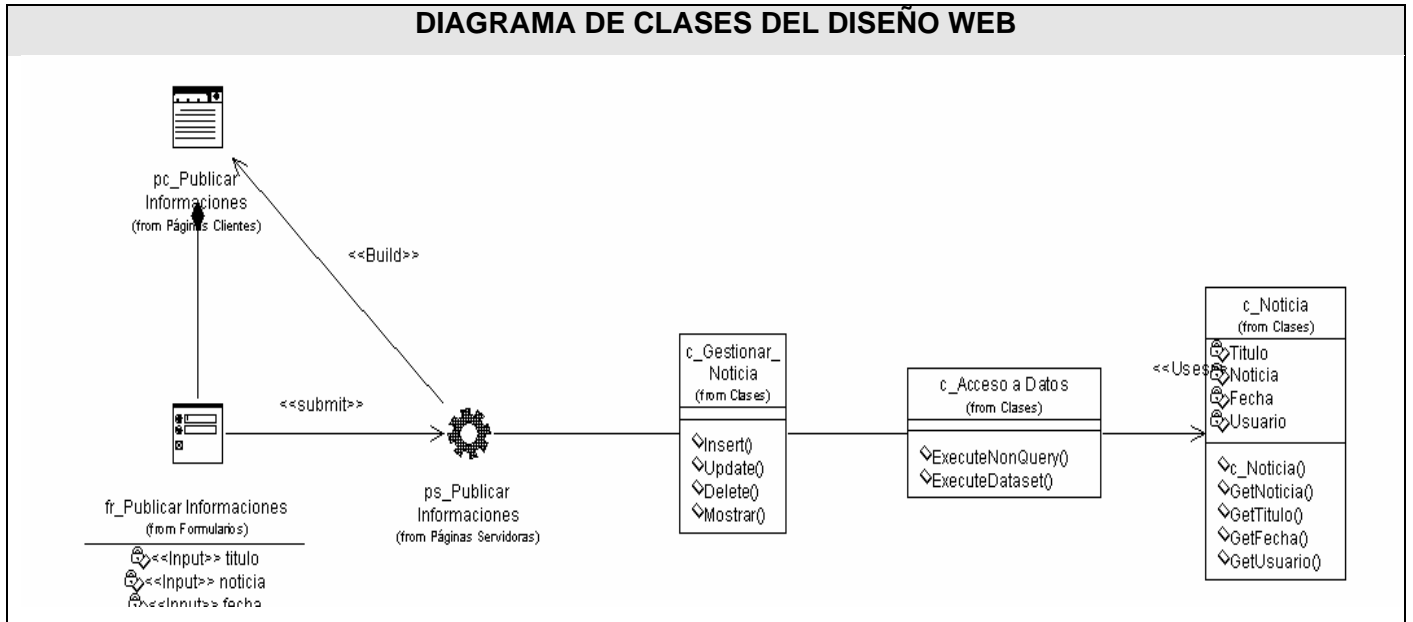


Figura 19. DCD. CUS. Publicar Información.

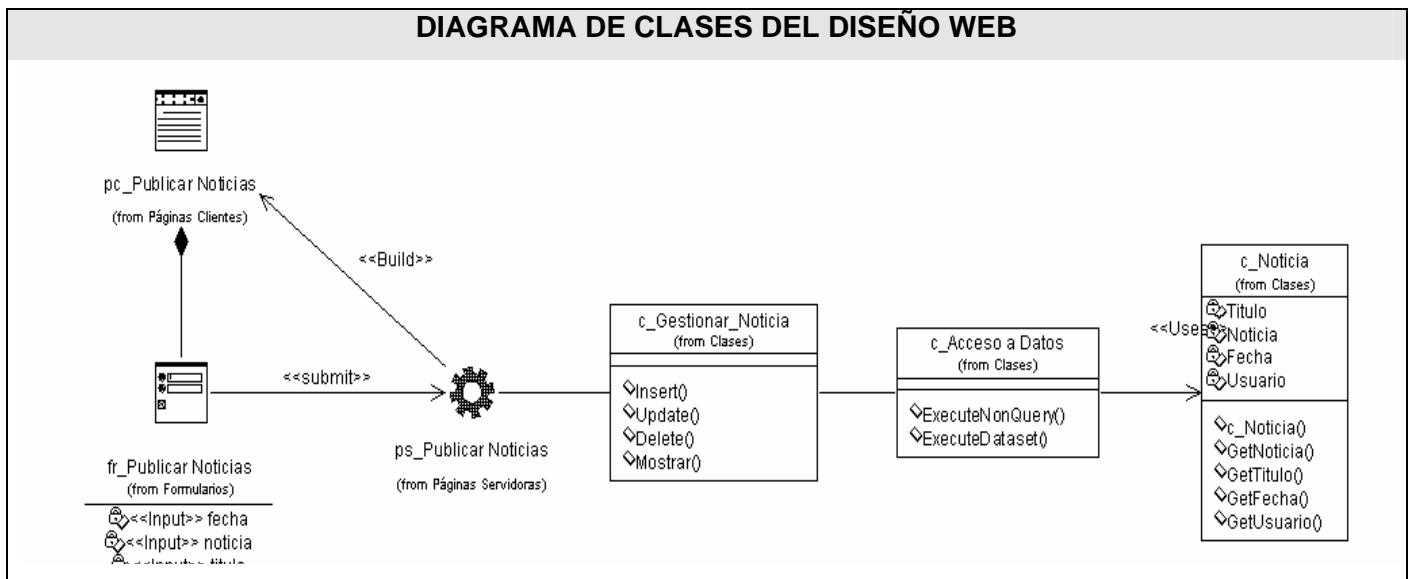


Figura 20. DCD. CUS. Publicar Noticia.

ANEXOS III Descripción de las Clases del Diseño Web.

Nombre: c_AA	
Tipo de clase: Entidad	
Atributo	Tipo
Usuario	string
ING	string
Proyecto	string
Actividades	string
Plan_W	string
Asig	string
Rol	string
Tutor	string
Para cada responsabilidad:	
Nombre:	c_AA(Usuario, ING, Proyecto, Actividades, Plan_w, Asig, Rol, Tutor)
Descripción:	Método constructor de la clase.
Nombre:	string Get_Usuario ()
Descripción:	Método que devuelve el Usuario de un Alumno Ayudante.
Nombre:	bool Get_ING()
Descripción:	Método que devuelve si el Alumno Ayudante es Instructor no Graduado.
Nombre:	string Get_Proyecto()

Descripción:	Método que devuelve el Proyecto de un Alumno Ayudante.
Nombre:	string Get_Actividades()
Descripción:	Método que devuelve las actividades de un Alumno Ayudante
Nombre:	string Get_Asig()
Descripción:	Método que devuelve la asignatura de un Alumno Ayudante
Nombre:	string Get_Rol()
Descripción:	Método que devuelve el rol de un Alumno Ayudante
Nombre:	string Get_Tutor()
Descripción:	Método que devuelve el nombre del tutor de un Alumno Ayudante

Nombre: c_Prof	
Tipo de clase: Entidad	
Atributo	Tipo
Usuario	string
Nombre	string
Cargo	string
Guia	string
Superacion	string
Proyecto	string
Militancia	string

Apto_Atiende	string
Evaluacion	string
Rol	string
Para cada responsabilidad:	
Nombre:	c_Prof (Usuario, Nombre, Cargo, Guia, Superacion, Proyecto , Militancia, Apto_Atiende,Evaluacion,Rol)
Descripción:	Método constructor de la clase.
Nombre:	string Get_Usuario ()
Descripción:	Método que devuelve el Usuario de un Profesor.
Nombre:	bool Get_Nombre()
Descripción:	Método que devuelve el Nombre de un Profesor
Nombre:	string Get_Cargo()
Descripción:	Método que devuelve el cargo de un Profesor.
Nombre:	bool Get_Guia()
Descripción:	Método que devuelve si es guía o no.
Nombre:	string Get_Superacion()
Descripción:	Método que devuelve la superación de un Profesor
Nombre:	string Get_Rol()
Descripción:	Método que devuelve el rol de un Profesor

Nombre:	string Get_Proyecto()
Descripción:	Método que devuelve el proyecto de un Profesor
Nombre:	string Get_Militancia()
Descripción:	Método que devuelve la militancia de un Profesor
Nombre:	string Get_Apto_Atiende()
Descripción:	Método que devuelve el Departamento que atiende el profesor.
Nombre:	string Get_Evaluacion()
Descripción:	Método que devuelve la evaluación de un profesor

Nombre: c_Noticia	
Tipo de clase: Entidad	
Atributo	Tipo
Titulo	string
Noticia	string
Fecha	Datetime
Usuario	string
Para cada responsabilidad:	
Nombre:	c_Noticia (Titulo, Noticia, Fecha, Usuario)
Descripción:	Método constructor de la clase.
Nombre:	string Get_Titulo ()

Descripción:	Método que devuelve el Usuario de un Alumno Ayudante.
Nombre:	bool Get_Noticia()
Descripción:	Método que devuelve si el Alumno Ayudante es Instructor no Graduado.
Nombre:	Datetime Get_Fecha()
Descripción:	Método que devuelve el Proyecto de un Alumno Ayudante.
Nombre:	string Get_Usuario()

Nombre: c_Horario	
Tipo de clase: Entidad	
Atributo	Tipo
Turno	string
Local	string
Grupo	string
Semana	String
Dia	String
Profesor	string
Para cada responsabilidad:	
Nombre:	c_Horario (Turno, Local, Semana, Dia, Profesor)
Descripción:	Método constructor de la clase.
Nombre:	string Get_Local ()

Descripción:	Método que devuelve el Local.
Nombre:	bool Get_Semana()
Descripción:	Método que devuelve la semana si par o impar
Nombre:	Datetime Get_Dia()
Descripción:	Método que devuelve el día de la semana.
Nombre:	string Get_Profesor ()
Descripción:	Método que devuelve el profesor.
Nombre:	string Get_Turno ()
Descripción:	Método que devuelve el turno.
Nombre:	string Get_Grupo ()
Descripción:	Método que devuelve Grupo.

Nombre: c_Acceso_Datos	
Tipo de clase: Controladora	
Para cada responsabilidad:	
Nombre:	Void ExecuteNonQuery(string conn, string SP, Object [] Param)
Descripción:	Método que se le pasa la conexión a la base datos, el nombre del procedimiento almacenado y un arreglo de valores que son los parámetros del procedimiento almacenado e inserta en la base datos.

Nombre:	Dataset ExecuteDataset(string conn, string SP, Object [] Param)
Descripción:	Método que se le pasa la conexión a la base datos, el nombre del procedimiento almacenado y un arreglo de valores que son los parámetros del procedimiento almacenado y devuelve una dataset con los valores devueltos por el procedimiento almacenado.

Nombre: c_Usuario	
Tipo de clase: Controladora	
Para cada responsabilidad:	
Nombre:	c_Usuario()
Descripción:	Método constructor de la clase.
Nombre:	Bool Valida(string User, string Pass)
Descripción:	Método que valida el usuario y la contraseña.
Nombre:	String Get_user()
Descripción:	Método que devuelve el usuario.
Nombre:	String Get_Nombreuser()
Descripción:	Método que devuelve el nombre de un usuario.
Nombre:	String Get_Tipouser()

Descripción:	Método que devuelve el tipo de usuario.
Nombre:	String Get_area()
Descripción:	Método que devuelve el área de un usuario.

Nombre: c_Informes	
Tipo de clase: Controladora	
Para cada responsabilidad:	
Nombre:	void Mostrar_AA(Dataset AA)
Descripción:	Método que se le pasa un dataset con los datos de un alumno ayudante y te los muestra.
Nombre:	void Mostrar_Prof (Dataset Prof)
Descripción:	Método que se le pasa un dataset con los datos de un profesor y te los muestra.
Nombre:	Dataset Get_AA (string User)
Descripción:	Método busca los datos de alumno ayudante según su usuario.
Nombre:	Dataset Get_Prof (string User)
Descripción:	Método busca los datos de profesor según su usuario.
Nombre:	Dataset List_Prof ()
Descripción:	Método que devuelve un dataset con todos los profesores.

Nombre:	Dataset List_AA ()
Descripción:	Método que devuelve un dataset con todos los alumnos ayudantes.

Nombre: c_Gestionar_Reportes	
Tipo de clase: Controladora	
Para cada responsabilidad:	
Nombre:	Dataset Get_Reportes(string Param)
Descripción:	Método que se le pasan una series de parámetros y devuelve una dataset con los valores del reporte.
Nombre:	void Mostrar(dataset Valores)
Descripción:	Método que muestra los valores de un reporte.

Nombre: c_Gestionar_Horarios	
Tipo de clase: Controladora	
Para cada responsabilidad:	
Nombre:	Dataset Get_Horarios(string Param)
Descripción:	Método que se le pasan una series de parámetros y devuelve una dataset con los valores del Horario.
Nombre:	void Mostrar (dataset Valores)

Descripción:	Método que muestra los valores de un Horario.
Nombre:	void Limpiar_Tabla (Tabla tb)
Nombre:	void Insert(string Grupo, string turno, string local, string profesor, string semana, string dia)
Descripción:	Método que inserta valores en la tabla horario.
Nombre:	void Delete(string Grupo, string turno, string local, string profesor, string semana, string dia)
Descripción:	Método que elimina valores en la tabla horario.
Nombre:	void Update(string Grupo, string turno, string local, string profesor, string semana, string dia)
Descripción:	Método que actualiza valores en la tabla horario.

Nombre: c_Gestionar_Noticia	
Tipo de clase: Controladora	
Para cada responsabilidad:	
Nombre:	void Insert (string Param)
Descripción:	Método que se le pasan una series de parámetros y devuelve una dataset con los valores del reporte.
Nombre:	void Mostrar(dataset Valores)
Descripción:	Método que muestra los valores de un reporte.

Nombre:	void Insert(string titulo,string noticia,datetime fecha, string usuario)
Descripción:	Método que Inserta una noticia
Nombre:	void Delete(int Id_Noticia)
Descripción:	Método que elimina una noticia.
Nombre:	void Update(string titulo,string string noticia,Datetime fecha,string usuario,int id_Noticia)
Descripción:	Método que actualiza una noticia

GLOSARIO

CLR (Common Language Runtime): Es el núcleo de la plataforma .NET. Es el motor encargado de gestionar la ejecución de las aplicaciones para ella desarrolladas y a las que ofrece numerosos servicios que simplifican su desarrollo y favorecen su fiabilidad y seguridad.

IIS (Internet Information Server): es una serie de servicios para los ordenadores que funcionan con Windows. Este servicio convierte a un ordenador en un servidor de Internet o Intranet es decir que en las computadoras que tienen este servicio instalado se pueden publicar páginas web tanto local como remotamente.

.NET Framework: componente de software que puede ser añadido al sistema operativo Windows. Provee un extenso conjunto de soluciones predefinidas para necesidades generales de la programación de aplicaciones, y administra la ejecución de los programas escritos específicamente con la plataforma. Esta solución es el producto principal en la oferta de Microsoft, y pretende ser utilizada por la mayoría de las aplicaciones creadas para la plataforma Windows.