

# **ESCUELA POLITÉCNICA NACIONAL**

## **ESCUELA DE FORMACIÓN TECNOLÓGICA**

### **DESARROLLO DE UN PORTAL WEB PARA EL INGRESO Y CONSULTA DE NOTAS DEL COLEGIO NACIONAL EXPERIMENTAL "AMAZONAS"**

**PROYECTO PREVIO A LA OBTENCIÓN DEL TÍTULO DE TECNÓLOGO EN  
ANÁLISIS DE SISTEMAS INFORMÁTICOS**

**HENRY HUMBERTO VARGAS GARCÍA**  
**henrynice\_007@hotmail.com**

**DIRECTORA: ING. MYRIAM PEÑAFIEL**  
**mpenafiel07@gmail.com**

**Quito, Octubre 2008**

## **CERTIFICACIÓN**

Certifico que el presente trabajo fue desarrollado por el Sr. Henry Humberto Vargas García, bajo mi supervisión.

**Ing. Myriam Peñafiel.**  
**DIRECTORA DE PROYECTO**

## **DECLARACIÓN**

Yo, Henry Humberto Vargas García, declaro bajo juramento que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentada para ningún grado o calificación profesional y que he consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración, cedo mis derechos de propiedad intelectual correspondientes a este trabajo a la Escuela Politécnica Nacional; según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.

---

Henry Humberto Vargas García

## **AGRADECIMIENTOS**

El presente trabajo, si bien ha requerido de esfuerzo y de dedicación de mi parte no hubiese sido posible sin la colaboración directa e indirecta de muchas personas, mil gracias a todas ellas.

Agradecer hoy y siempre a mi familia porque a pesar de no estar presentes físicamente, se que procuran mi bienestar desde mi ciudad, Guaranda, y esta claro que si no fuese por el esfuerzo realizado por ellos, mis estudios no hubiesen sido posible. A mis padres Humberto y Gloria, mi hermano Joffre, porque a pesar de la distancia, el ánimo, apoyo y alegría que me brindan me dan la fortaleza necesaria para seguir adelante.

## DEDICATORIA

El esfuerzo y la dedicación que he puesto en este proyecto, va Con mucho cariño principalmente a mis padres Humberto y Gloria que me dieron la vida y me enseñaron con ejemplo a ser una persona de bien, que pese a los problemas siempre tener fuerza para levantarse y seguir adelante. A mis más queridos amigos, pues su consejo, ha sido parte de este esfuerzo para conseguir un anhelo importante en mi vida.

# CONTENIDO

CAPITULO 1 .....	9
1. INTRODUCCIÓN.....	9
1.1 PLANTEAMIENTO DEL PROBLEMA.....	9
1.2 OBJETIVOS.....	10
1.2.1 OBJETIVO GENERAL .....	10
1.2.2 OBJETIVOS ESPECÍFICOS .....	10
1.3 JUSTIFICACIÓN.....	10
1.4 ALCANCE .....	10
1.5 LIMITACIONES.....	11
1.6 PRESUPUESTO.....	11
CAPITULO 2 .....	12
2. MARCO TEORICO .....	12
2.1 INGENIERÍA WEB .....	12
2.1.1 INTRODUCCION.....	12
2.1.2 EL PROCESO DE INGENIERÍA WEB.....	13
2.1.3 CONTROL Y GARANTÍA DE LA CALIDAD.....	14
2.1.4 CONTROL DE LA CONFIGURACIÓN .....	14
2.1.5 LA GESTIÓN DEL PROCESO.....	15
2.2 ARQUITECTURA DE LA INFORMACIÓN .....	15
2.2.1 DEFINICIÓN .....	15
2.2.2 ARQUITECTURA WEB .....	17
2.2.3 ARQUITECTURA DE LA INFORMACIÓN COMO PROCESO.....	18
2.2.4 ARQUITECTURA DE LA INFORMACIÓN COMO ABSTRACCIÓN... ..	19
2.2.5 EL ARQUITECTO DE INFORMACIÓN .....	19
2.2.6 LA EXPERIENCIA DE USUARIO .....	20
2.3 DISEÑO DE PAGINAS WEB .....	22
2.3.1 DISEÑO WEB APLICADO .....	23
2.3.2 ETAPAS .....	24
2.4 APLICACIONES WEB Y EL DESARROLLO DE CAPAS.....	24
2.4.1 ARQUITECTURAS DE TRES NIVELES O PROGRAMACIÓN POR CAPAS .....	25
2.4.2 SOFTWARE INTERMEDIO (Middleware).....	29
2.5 ASPECTOS METODOLÓGICOS.....	31
2.5.1 PARADIGMA ESPIRAL ORIENTADO A LA WEB .....	31
2.5.2 COMUNICACIÓN CON EL CLIENTE O FORMULACIÓN .....	31
2.5.3 PLANIFICACIÓN .....	32
2.5.4 ANÁLISIS DE RIESGOS.....	32
2.5.5 INGENIERÍA .....	32
2.5.6 GENERACIÓN DE PÁGINAS .....	32
2.5.7 EVALUACIÓN DEL CLIENTE .....	33
2.6 METODOLOGÍA OOHDM .....	33
2.6.1 INTRODUCCIÓN A OOHDM.....	33
2.6.2 FASES DE LA METODOLOGÍA OOHDM.....	34
2.7 UML (Unified Modeling Language).....	39
2.7.1 DIAGRAMAS PARA MODELAMIENTO.....	41
CAPITULO 3 .....	58

3. HERRAMIENTAS DE CONSTRUCCION WEB .....	58
3.1 CAPA DE PRESENTACIÓN .....	58
3.1.2 CAPA DE NEGOCIO .....	60
3.1.3 CAPA DE DATOS .....	63
3.1.4 MACROMEDIA FLASH MX 2004 .....	64
3.1.5 MACROMEDIA DREAMWEAVER MX 2004 .....	65
3.1.6 MACROMEDIA FIREWORKS MX 2004.....	66
CAPITULO 4 .....	68
4. CONCLUSIONES Y RECOMENDACIONES .....	68
4.1 CONCLUSIONES .....	68
4.2 RECOMENDACIONES .....	68
BIBLIOGRAFÍA .....	69
ANEXOS .....	71



# ***CAPITULO 1***

## **1. INTRODUCCIÓN**

### **1.1 PLANTEAMIENTO DEL PROBLEMA**

El COLEGIO NACIONAL EXPERIMENTAL “AMAZONAS” es una institución educativa fiscal mixta de nivel secundario, la misma que fue fundada en el año de 1961 con el propósito de formar a la juventud en la diversidad cultural, con capacidad de liderazgo, libertad y fortalecer la identidad nacional. Está ubicada al sur de la ciudad de Quito, en las calles Iturralde s/n y Lauro Guerrero.

Al momento los procesos administrativos y de registro académico, en la Institución Educativa, se lleva en forma manual, por lo que se presenta entre otros los siguientes problemas:

- Los datos personales de estudiantes y profesores se archivan en carpetas.
- El ingreso de los aportes se realiza a través de formularios, cuyos datos luego se transcriben a libros, que constituyen el único archivo de la institución.
- Para la generación de reportes trimestrales se usa una hoja electrónica y los datos se ingresan estudiante por estudiante desde los libros.

Esto causa una serie de inconvenientes y molestias para el personal administrativo, personal docente, estudiantes y padres de familia, entre los cuales citamos:

- El colegio no dispone de un mecanismo eficiente para manejar y hacer llegar la información a padres de familia y otros interesados.
- El personal encargado de emitir los reportes trimestrales no lo puede realizar a tiempo debido a la demora en la entrega de notas de algunos profesores y el retraso que conlleva la transcripción de los datos de los libros a los reportes trimestrales; por ello se debe postergar la entrega de éstos a los padres de familia. Este problema también afecta al momento de enviar la información al Ministerio de Educación y Cultura.

- La información de los estudiantes, representantes y profesores no están organizados y no se puede hacer una verificación de la información registrada en los libros.

- El acceso y transferencia de los datos no es confiable.
- Existe la posibilidad de pérdida de datos.

## **1.2 OBJETIVOS**

### **1.2.1 OBJETIVO GENERAL**

Apoyar la gestión administrativa; informativa y académica del COLEGIO NACIONAL EXPERIMENTAL "AMAZONAS", a través del desarrollo de una aplicación Web.

### **1.2.2 OBJETIVOS ESPECÍFICOS**

- Administrar perfiles de usuario para permitir el ingreso y control de usuarios con las respectivas restricciones de acceso al sistema.
- Registrar y actualizar datos personales de estudiantes y profesores.
- Permitir el ingreso de notas.
- Permitir la consulta de notas
- Generar de Reportes de Notas.

## **1.3 JUSTIFICACIÓN**

Este proyecto permitirá mejorar la administración académica del colegio Nacional Experimental Amazonas, respondiendo a las necesidades crecientes de la institución.

Esta aplicación facilitará la automatización de una gran parte de la administración académica del colegio, así como también se ofrecerá información oportuna a los interesados.

## **1.4 ALCANCE**

- Permite el ingreso de usuarios con las respectivas restricciones de acceso al sistema.
- Permite el ingreso de notas, alumnos, profesores, materias y asistencia.
- Permite generar de Reportes:

- Notas por alumno
- Notas por curso.
- Notas por Profesor y Materia
- Lista de alumnos por curso.

## 1.5 LIMITACIONES

- No contempla inscripciones previas al ingreso del colegio.
- No contempla ningún procedimiento para la matriculación.
- No maneja información financiera (cobro de matrículas).

## 1.6 PRESUPUESTO

	Costo por hora (\$)	Horas	Total (\$)
<b>Recursos Humanos</b>			
Diseñador	5	180	900
Programador	5	260	1300
<b>Software</b>			
Apache	-	-	0
PHP	-	-	0
MySQL	-	-	0
JavaScript	-	-	0
Macromedia	-	-	0
<b>Hardware</b>			
PC de Escritorio	-	-	-
<b>Otros</b>			
Impresiones y copias	-	-	165
Internet	0.75	110	82.25
		<b>Total :</b>	2447.25

Software	
Descripción	Costo
Apache	FREE
PHP	
MySQL	

JavaScript	
Macromedia	
Dreamweaver	

## ***CAPITULO 2***

### **2. MARCO TEORICO**

#### **2.1 INGENIERÍA WEB**

##### **2.1.1 INTRODUCCION**

La Ingeniería de la Web es la aplicación de metodologías sistemáticas, disciplinadas y cuantificables al desarrollo eficiente, operación y evolución de aplicaciones de alta calidad en la World Wide Web.

[http://es.wikipedia.org/wiki/Ingenier%C3%ADa\\_de\\_la\\_Web](http://es.wikipedia.org/wiki/Ingenier%C3%ADa_de_la_Web)

En 1998, Roger Pressman moderó una mesa redonda virtual con representantes la ingeniería software tradicional y del desarrollo software basado exclusivamente en Internet. El debate principalmente se centró en discutir si valía la pena aplicar un proceso de ingeniería a las aplicaciones con base en Internet, o qué características tenían éstas que justificaran el no utilizarlo. La conclusión general fue que aplicar un proceso de ingeniería nunca es una mala idea pero que éste debería adaptarse a los requerimientos de cambio continuo y rapidez siempre presentes en el proceso de desarrollo Web. De iniciativas como ésta y de otras como la organización de congresos y talleres especializados en el desarrollo para la Web, surge el nacimiento de una nueva disciplina denominada Ingeniería Web.

Escuetamente podemos “traducir” como el proceso utilizado para crear, implantar y mantener aplicaciones y sistemas Web de alta calidad.

Esta breve definición nos lleva a abordar un aspecto clave de cualquier proyecto como es determinar que tipo de proceso es más adecuado en función de las características del mismo.

## 2.1.2 EL PROCESO DE INGENIERÍA WEB

Características como inmediatez y evolución y crecimiento continuos, nos llevan a un proceso incremental y evolutivo, que permite que el usuario se involucre activamente, facilitando el desarrollo de productos que se ajustan mucho lo que éste busca y necesita. Según Presuman, las actividades que formarían parte del marco de trabajo incluirían las tareas abajo enumeradas. Dichas tareas serían aplicables a cualquier aplicación Web, independientemente del tamaño y complejidad de la misma.

Las actividades que forman parte del proceso son: *formulación, planificación análisis, modelización, generación de páginas, test y evaluación del cliente.*

**La Formulación.-** identifica objetivos y establece el alcance de la primera entrega.

**La Planificación.-** genera la estimación del coste general del proyecto, la evaluación de riesgos y el calendario del desarrollo y fechas de entrega. El Análisis especifica los requerimientos e identifica el contenido.

**La Modelización.-** se compone de dos secuencias paralelas de tareas. Una consiste en el diseño y producción del contenido que forma parte de la aplicación. La otra, en el diseño de la arquitectura, navegación e interfaz de usuario. Es importante destacar la importancia del diseño de la interfaz. Independientemente del valor del contenido y servicios prestados, una buena interfaz mejora la percepción que el usuario tiene de éstos.

**La Generación de páginas.-** se integra contenido, arquitectura, navegación e interfaz para crear estática o dinámicamente el aspecto más visible de las aplicaciones, las páginas.

**El Test.-** busca errores a todos los niveles: contenido, funcional, navegacional, rendimiento, etc. El hecho de que las aplicaciones residan en la red, y que ínter

operen en plataformas muy distintas, hace que el proceso de test sea especialmente difícil.

**Finalmente**, el resultado es sometido a la evaluación del cliente.

*Pressman, 813.*

### **2.1.3 CONTROL Y GARANTÍA DE LA CALIDAD**

Una de las tareas colaterales que forman parte del proceso es el Control y Garantía de la Calidad (CGC). Todas las actividades CGC de la ingeniería software tradicional como son: establecimiento y supervisión de estándares, revisiones técnicas formales, análisis, seguimiento y registro de informes, etc, son igualmente aplicables a la *Ingeniería Web*.

Sin embargo, en la *Web* toman especial relevancia para valorar la calidad aspectos como: Usabilidad, Funcionabilidad, Fiabilidad, Seguridad, Eficiencia y Mantenibilidad.

*Olsina, 266*

### **2.1.4 CONTROL DE LA CONFIGURACIÓN**

Establecer mecanismos adecuados de control de la configuración para la Ingeniería Web es uno de los mayores desafíos a los que esta nueva disciplina se enfrenta. La Web tiene características únicas que demandan estrategias y herramientas nuevas. Hay cuatro aspectos importantes a tener en cuenta en el desarrollo de tácticas de control de la configuración para la Web.

**Contenido:** Considerando la dinamicidad con la que el contenido se genera, es tarea compleja organizar racionalmente los objetos que forman la configuración y establecer mecanismos de control.

**Personal:** Cualquiera realiza cambios. Hay mucho personal no especializado que no reconoce la importancia que tiene el control del cambio.

**Escalabilidad:** Es común encontrar aplicaciones que de un día para otro crecen considerablemente. Sin embargo, las técnicas de control no escalan de forma adecuada.

**Política:** ¿Quién posee la información? ¿Quién asume la responsabilidad y coste de mantenerla?

### **2.1.5 LA GESTIÓN DEL PROCESO**

En un proceso tan rápido como es el proceso de Ingeniería Web, donde los tiempos de desarrollo y los ciclos de vida de los productos son tan cortos, ¿merece la pena el esfuerzo requerido por la gestión? La respuesta es que dada su complejidad es imprescindible. Entre los aspectos que añaden dificultad a la gestión destacamos:

- Alto porcentaje de contratación a terceros,
- El desarrollo incluye una gran variedad de personal técnico y no técnico trabajando en paralelo, el equipo de desarrollo debe dominar aspectos tan variados como, software basado en componentes, redes, diseño de arquitectura y navegación, diseño gráfico y de interfaces, lenguajes y estándares en Internet, test de aplicaciones Web, etc, lo que hace que el proceso de búsqueda y contratación de personal sea arduo.

<http://www.informandote.com/jornadasIngWEB/articulos/jiw01.pdf>

## **2.2 ARQUITECTURA DE LA INFORMACIÓN**

### **2.2.1 DEFINICIÓN**

El término "Arquitectura de la Información" fue utilizado por primera vez por Richard Saul Wurman en 1975, quién la define como:

*“El estudio de la organización de la información con el objetivo de permitir al usuario encontrar su vía de navegación hacia el conocimiento y la comprensión de la información”*

Si nos ceñimos exclusivamente a la Arquitectura de la Información en el campo de la Web, una de las definiciones que Louis Rosenfeld y Peter Morville ofrecen en su libro "Information Architecture for the World Wide Web 2nd Edition", puede sernos de más fácil comprensión:

*“El arte y la ciencia de estructurar y clasificar sitios web e intranets con el fin de ayudar a los usuarios a encontrar y manejar la información.”*

<http://www.nosolousabilidad.com/articulos/ai.htm>

En relación a la World Wide Web el Information Architecture Institute define la Arquitectura de la Información como:

- *El diseño estructural en entornos de información compartida.*
- *El arte y la ciencia de organizar y rotular sitios web, intranets, comunidades en línea y software para promover la usabilidad y encontrabilidad.*
- *Una comunidad emergente orientada a aplicar los principios del diseño y la arquitectura en el entorno digital.*

[http://es.wikipedia.org/wiki/Arquitectura\\_de\\_la\\_informaci%C3%B3n](http://es.wikipedia.org/wiki/Arquitectura_de_la_informaci%C3%B3n)

El concepto "Arquitectura de la Información" no solo engloba la actividad de organizar información, sino también el resultado de dicha actividad. La arquitectura de la información de un sitio web, como resultado de la actividad, comprende los sistemas de organización y estructuración de los contenidos, los sistemas de rotulado o etiquetado de dichos contenidos, y los sistemas de recuperación de información y navegación que provea el sitio web.

Independientemente de la velocidad, están los problemas de Navegación, de Organización, de Estructura, de Usabilidad, es decir de la Arquitectura de la información. En definitiva de Arquitectura Web.

El "diseño" cumple una importante función, pero no es la única que compone una web atractiva y funcional y, al parecer, algunos lo han olvidado. Un sitio web se

construye, no sólo se diseña. Hay que juntar diferentes piezas, que deben encajar convenientemente para que el usuario/navegante pueda, al menos, conocernos. Si no es así, difícilmente podrá convertirse en un posible cliente.

El principal "atractivo" de una web, debería residir en su contenido y funcionalidad. La Arquitectura Web pretende reunir todas las piezas y juntarlas en la forma más adecuada para construir el sitio que desea nuestro público objetivo.

La web debe construirse para el usuario. No para el diseñador, ni para el directivo que lo ha contratado, ni para su Consejo de Administración. Presentación, diseño, navegación, marketing, interactividad, estructura, flujo transaccional, información... son algunas de las piezas que debe contemplar el Arquitecto de la web, para su correcta construcción.

<http://www.webaprendiz.com/viewwp/notas/arquiweb/2001/06/id0001.htm>

### **2.2.2 ARQUITECTURA WEB**

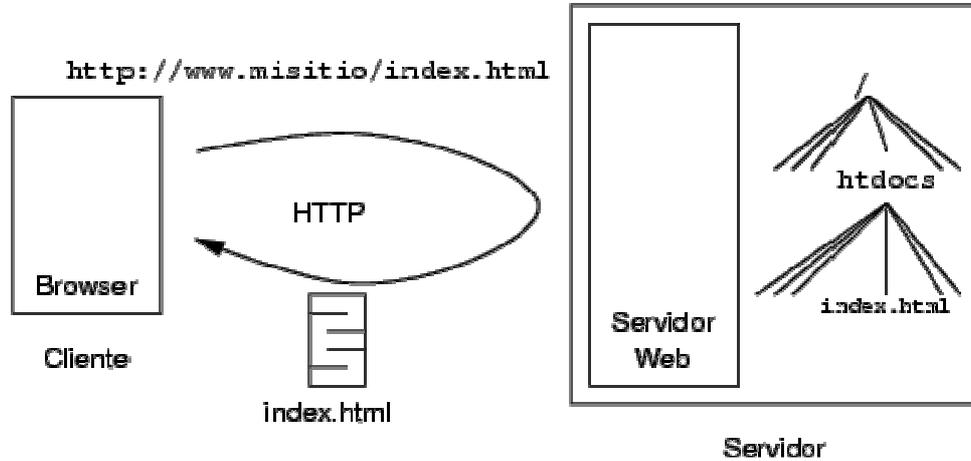
De la misma forma en que los arquitectos tradicionales diseñan y coordinan la construcción de edificios, los arquitectos Web diseñan y coordinan el desarrollo de sitios Web. Los sitios Web son una conjunción muy compleja de distintos sistemas integrados entre sí (Bases de datos, servidores, redes, componentes de backup y seguridad, etc...).

El resultado final será un sitio que pueda resolver las necesidades de negocios: Vender productos y servicios online y servir mejor a las necesidades de los clientes. Las aplicaciones deben ser diseñadas sobre de las necesidades de las personas que van a utilizarlas.

<http://www.pabloimpallari.com.ar/articulos/arquitecturaweb.php>

Para abrir una página Web en un navegador, normalmente se teclea el correspondiente URL o se emplea el hiperenlace oportuno. Una vez que se solicita esta petición mediante el protocolo HTTP y la recibe el servidor Web, éste localiza

la página Web en su sistema de ficheros y la envía de vuelta al navegador que la solicitó, según se muestra en la en la siguiente figura.



**Figura:** Arquitectura Web Básica

<http://www.infor.uva.es/~jvegas/cursos/buendia/pordocente/node18.html>

### 2.2.3 ARQUITECTURA DE LA INFORMACIÓN COMO PROCESO

La Arquitectura de la Información es un proceso cíclico, transversal, que se da a lo largo del diseño del sitio y en cada una de sus fases, para asegurarse de que los objetivos de su producción y del desarrollo de la interfaz se cumplen de manera efectiva.

El fin de la Arquitectura de la Información no es la búsqueda de una metodología de diseño universal sino la articulación de un conjunto de técnicas para ayudar al desarrollo y producción de los sitios.

Con el fin de que la asimilación de contenidos por parte del usuario sea eficiente y efectiva (accesible y usable) la Arquitectura de la Información como proceso en general, se encarga, durante el proceso de desarrollo de los sitios, de definir:

- El objeto, propósito y fines del sistema de información o sitio
- La definición del público objetivo y los estudios de la audiencia.
- El diseño de la interacción
- El diseño de la navegación
- La planificación, gestión y desarrollo de contenidos

- La facilidad de búsqueda y el diseño de la interfaz de búsqueda.
- La usabilidad.
- El feedback del resultado y el rediseño de la interfaz.

#### **2.2.4 ARQUITECTURA DE LA INFORMACIÓN COMO ABSTRACCIÓN**

La Arquitectura de la Información no trata de un conjunto de pasos o guías predefinidas, sino la omnipresente inteligencia que está detrás y a cargo de que interactividad, navegación y contenido sinérgicamente permitan una integración sistémica con el cerebro del usuario y se produzca un fenómeno de persuasión, conocimiento o información simbiótica, que se traspasa de un sistema a otro, según sean los objetivos del proyecto en general y los objetivos iniciales del usuario al interactuar con la interfaz.

De esta forma las acciones de buscabilidad, encontrabilidad y recuperabilidad de información se realizan en un contexto óptimo en ambos nodos de información (interfaz y usuario), hacen simbiosis e inician un proceso de comunicación que los enriquecerá de cualquier forma, la interfaz cumple con su objetivo y puede ser mejorada, y el usuario encuentra lo que busca y lo asimila.

#### **2.2.5 EL ARQUITECTO DE INFORMACIÓN**

Es la persona encargada de llevar a cabo y verificar el proceso de desarrollo del sitio trabajando estrechamente con los expertos en usabilidad para definir la interfaz de usuario. Está integrado en un equipo y se ocupa desde la fundamentación del proyecto hasta el rediseño del producto, verificando y testeando todas las fases hasta la obtención del resultado final.

Se trata de una nueva profesión surgida en un contexto social, cultural, económico y político fuertemente condicionado por las Nuevas Tecnologías de la Información, las cuales han modificado bruscamente en todos los niveles las formas de comunicación entre seres humanos, así como el medio en que perciben y asimilan información. Estos avances en telecomunicaciones, ciencia, y tecnología en

general han producido una cantidad ingente de conocimiento, nuevos conceptos, nuevas ideas, nuevos métodos, nuevos procesos, nuevas visiones, nuevos problemas, nuevas soluciones de tal forma que es necesario hacer llegar al usuario común estos nuevos conceptos y soluciones de manera simple, objetiva y precisa. Es aquí, entre otras instancias, en que interviene la Arquitectura de la Información.

Procesar y dosificar la enorme cantidad de información que se ha producido a causa de los descubrimientos, nuevas investigaciones en todos y nuevos campos, a causa de la revolución de Internet y ponerla de una manera clara, relevante y significativa a disposición del usuario común, se trata, entre otras cosas, de hacer comprensible lo abstracto de alguna forma.

Desarrollar y verificar procesos de producción o diseño de información con el fin de que el usuario pueda hacer recuperación de esa información de manera clara, precisa y sin ambigüedades desde cualquier plataforma o soporte; en especial hablamos de soportes multimediales e interactivos, aunque retomando a Shedroff en la práctica no debemos omitir ningún soporte por plano que este sea y hablar de experiencias de usuario.

Organizar, estructurar, sistematizar (Tufte), rotular, distribuir, diseñar estructuralmente sistemas de información (Baeza, Rivera, Velasco, 2003) con el fin de que el usuario pueda hacer de su experiencia de recuperación algo simple, agradable, eficaz y productivo.

### **2.2.6 LA EXPERIENCIA DE USUARIO**

Se entiende por Experiencia del Usuario el conjunto de factores y elementos que determinan una interacción satisfactoria del usuario con un entorno o dispositivo concretos. En ella influyen desde el diseño gráfico, la estética, hasta la articulación de la interfaz, la funcionalidad definida por el diseño de interacción y la extrapolación de principios del mundo del marketing, la psicología etc. Nathan

Shedroff extiende el concepto de Experiencia del usuario más allá de la Web planteando su Teoría Unificada del Diseño que articula en torno a los conceptos clave de diseño de información e interacción.

[http://es.wikipedia.org/wiki/Arquitectura\\_de\\_la\\_informaci%C3%B3n](http://es.wikipedia.org/wiki/Arquitectura_de_la_informaci%C3%B3n)

### **2.2.6.1 Diseño de información**

Hay tres áreas principales de donde provienen los practicantes del diseño de información: del diseño gráfico, de la documentación técnica (del inglés technical documentation) y de la informática. No es de extrañarse, pues el conjunto de estas tres disciplinas representan en gran medida lo que hace un diseñador de información. Explican (documentación técnica), recolectan datos y los transforman en información digerible (informática) y presentan información visualmente (diseño gráfico).

[http://es.wikipedia.org/wiki/Dise%C3%B1o\\_de\\_informaci%C3%B3n](http://es.wikipedia.org/wiki/Dise%C3%B1o_de_informaci%C3%B3n)

*Diseño gráfico.*- El diseño gráfico es una forma de comunicación visual. Se ocupa de organizar imagen y texto para comunicar un mensaje. Puede aplicarse a muchos medios de comunicación, ya sean impresos, digitales, audiovisuales, u otros. El diseño gráfico publicitario es una rama perteneciente al diseño gráfico, pero con objetivos comerciales definidos.

[http://es.wikipedia.org/wiki/Dise%C3%B1o\\_gr%C3%A1fico](http://es.wikipedia.org/wiki/Dise%C3%B1o_gr%C3%A1fico)

*Informática.*- La Informática es la ciencia del tratamiento automático de la información a través de un computador (llamado también ordenador o computadora).

<http://es.wikipedia.org/wiki/Inform%C3%A1tica>

*Documentación Técnica.*- La documentación técnica es un aspecto sumamente importante, tanto en el desarrollo de la aplicación como en el mantenimiento de la misma. Mucha gente no hace esta parte del desarrollo y no se da cuenta de que pierde la posibilidad de la reutilización de parte del programa en otras

aplicaciones, sin necesidad de conocerse el código al dedillo. La documentación de un programa empieza a la vez que la construcción del mismo y finaliza justo antes de la entrega del programa o aplicación al cliente. Así mismo, la documentación que se entrega al cliente tendrá que coincidir con la versión final de los programas que componen la aplicación. Una vez concluido el programa, los documentos que se deben entregar son una guía técnica, una guía de uso y de instalación.

<http://www.desarrolloweb.com/articulos/importancia-documentacion.html>

#### **2.2.6.2 Interacción**

Es el proceso que establece un usuario con un dispositivo, sistema u objeto determinado. De Entre otros factores en el diseño de interacción intervienen disciplinas como la usabilidad o la ergonomía.

<http://es.wikipedia.org/wiki/Interacci%C3%B3n>

### **2.3 DISEÑO DE PAGINAS WEB**

El diseño web es una actividad que consiste en la planificación, diseño e implementación de sitios web y páginas web. No es simplemente una aplicación del diseño convencional sobre internet ya que requiere tener en cuenta cuestiones tales como navegabilidad, interactividad, usabilidad, arquitectura de la información y la interacción de medios como el audio, texto, imagen y vídeo.

El diseño web no sólo aporta a la comunicación textual (contenidos) existente en internet una faceta visual, sino que obliga a pensar una mejor estructuración de los mismos en este soporte. La unión de un buen diseño con una jerarquía bien elaborada de contenidos aumenta la eficiencia de la web como canal de comunicación e intercambio de datos, que brinda posibilidades como el contacto directo entre el productor y el consumidor de contenidos, característica destacable del medio internet.

El diseño Web ha visto amplia aplicación en los sectores comerciales de internet especialmente en la World Wide Web. Asimismo, a menudo la Web se utiliza como medio de expresión plástica en sí. Artistas y creadores hacen de las páginas en Internet un medio más para ofrecer sus producciones y utilizarlas como un canal más de difusión de su obra.

### **2.3.1 DISEÑO WEB APLICADO**

El diseño de páginas Web es una amplia área de aplicación del diseño gráfico en la cual se integran conocimientos propios del diseño como son la composición, el uso de color y la tipografía con conocimientos técnicos del medio como son los lenguajes HTML (HiperText Markup Language) y CSS (Cascading Style Sheets), así como conocimientos sobre usabilidad, accesibilidad y organización de un sitio Web.

El diseño de páginas Web trata básicamente de realizar un documento con información hiperenlazado con otros documentos y asignarle una presentación para diferentes dispositivos de salida (en una pantalla de ordenador, en papel, en un teléfono móvil, etc).

Estos documentos o páginas Web pueden ser creados:

- Directamente el código HTML con un procesador de texto plano, como el mismo Bloc de Notas de Windows.
- Utilizando un programa WYSIWYG de creación de páginas como Adobe GoLive, Macromedia Dreamweaver, NVU o Microsoft Frontpage.
- Utilizando lenguajes de programación del lado del servidor como ASP.NET, JSP, PHP, Perl, etc para generar la página web
- Usando un sistema CMS (Web Content Management System) que permite separar el contenido del diseño y que permite al diseñador generar solamente las plantillas, de modo que el contenido sea administrado en un área separada. Algunos sistemas CMS son el PhpNuke, Plone y Mambo.

### **2.3.2 ETAPAS**

Para el diseño de páginas web debemos tener en cuenta dos etapas:

*La primera* es el diseño visual de la información que se desea editar. En esta etapa se trabaja en el papel distribuyendo el texto, los gráficos, los vínculos a otros documentos y otros objetos multimedia que se consideren pertinentes.

*La segunda*, una vez que se tiene este boceto se pasa a 'programar' la página web. Para esto, y fundamentalmente para manejar los vínculos entre documentos, se creó el lenguaje de marcación de hipertexto o HTML. Los enlaces que aparecen subrayados en este documento y otros de Wikipedia son ejemplos de hipertexto, puesto que al pulsar sobre ellos conducen a otras páginas con información relacionada.

El HTML consta de una serie de etiquetas que estructuran el texto y son presentados en forma de hipertexto por navegadores. Esto se puede hacer con un simple editor de textos (debe guardarse como texto plano, sin ningún tipo de formato). Aprender HTML es relativamente fácil, así que es sencillo crear páginas web de este modo. Esta era la única manera de generarlas hasta que aparecieron, a mediados de 1996, algunos editores visuales de HTML, como MS FrontPage y Macromedia Dreamweaver. Con estas herramientas no es absolutamente necesario aprender HTML (aunque sí aconsejable), con lo cual el desarrollador se concentra en lo más importante, el diseño del documento.

Todo esto teniendo en cuenta el nivel de programación en el diseño de las aplicaciones y del impacto visual que se quiere generar en el usuario.

### **2.4 APLICACIONES WEB Y EL DESARROLLO DE CAPAS**

Al hablar del desarrollo de aplicaciones Web resulta adecuado presentarlas dentro de las aplicaciones multinivel. Los sistemas típicos cliente/servidor pertenecen a la categoría de las aplicaciones de dos niveles. La aplicación reside en el cliente

mientras que la base de datos se encuentra en el servidor. En este tipo de aplicaciones el peso del cálculo recae en el cliente, mientras que el servidor hace la parte menos pesada, y eso que los clientes suelen ser máquinas menos potentes que los servidores.

<http://www.infor.uva.es/~jvegas/cursos/buendia/pordocente/node21.html>

### **2.4.1 ARQUITECTURAS DE TRES NIVELES O PROGRAMACIÓN POR CAPAS**

La programación por capas es un estilo de programación en la que el objetivo primordial es la separación de la lógica de negocios de la lógica de diseño, un ejemplo básico de esto es separar la capa de datos de la capa de presentación al usuario.

La ventaja principal de este estilo, es que el desarrollo se puede llevar a cabo en varios niveles y en caso de algún cambio sólo se ataca al nivel requerido sin tener que revisar entre código mezclado. Un buen ejemplo de este método de programación sería: Modelo de interconexión de sistemas abiertos

Además permite distribuir el trabajo de creación de una aplicación por niveles, de este modo, cada grupo de trabajo está totalmente abstraído del resto de niveles, simplemente es necesario conocer la API que existe entre niveles.

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

#### **2.4.1.1 Capas o Niveles**

El diseño más en boga actualmente es el diseño en tres niveles (o en tres capas).

*Capa de presentación:* es la que ve el usuario, presenta el sistema al usuario, le comunica la información y captura la información del usuario dando un mínimo de proceso (realiza un filtrado previo para comprobar que no hay errores de formato). Esta capa se comunica únicamente con la capa de negocio.

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

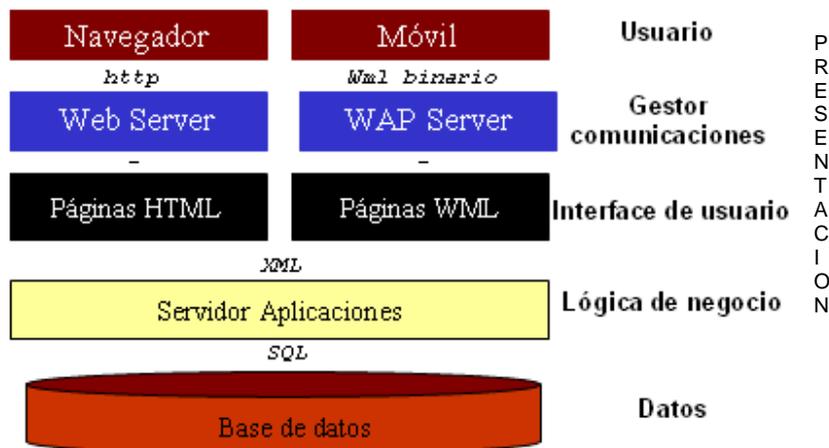
*Capa de datos:* es donde residen los datos. Está formada por uno o más gestores de bases de datos que realiza todo el almacenamiento de datos, reciben solicitudes de almacenamiento o recuperación de información desde la capa de negocio.

Todas estas capas pueden residir en un único ordenador (no sería lo normal), si bien lo más usual es que haya una multitud de ordenadores donde reside la capa de presentación (son los clientes de la arquitectura cliente/servidor). Las capas de negocio y de datos pueden residir en el mismo ordenador, y si el crecimiento de las necesidades lo aconseja se pueden separar en dos o mas ordenadores. Así, si el tamaño o complejidad de la base de datos aumenta, se puede separar en varios ordenadores los cuales recibirán las peticiones del ordenador en que resida la capa de negocio.

Si por el contrario fuese la complejidad en la capa de negocio lo que obligase a la separación, esta capa de negocio podría residir en uno o mas ordenadores que realizarían solicitudes a una única base de datos. En sistemas muy complejos se llega a tener una serie de ordenadores sobre los cuales corre la capa de datos, y otra serie de ordenadores sobre los cuales corre la base de datos. En una arquitectura de tres niveles, los términos "capas" y "niveles" no significan lo mismo ni son similares.

El término "capa" hace referencia a la forma como una solución es segmentada desde el punto de vista lógico:

Presentación/ Lógica de Negocio/ Datos.



En cambio, el término "nivel", corresponde a la forma en que las capas lógicas se encuentran distribuidas de forma física. Por ejemplo:

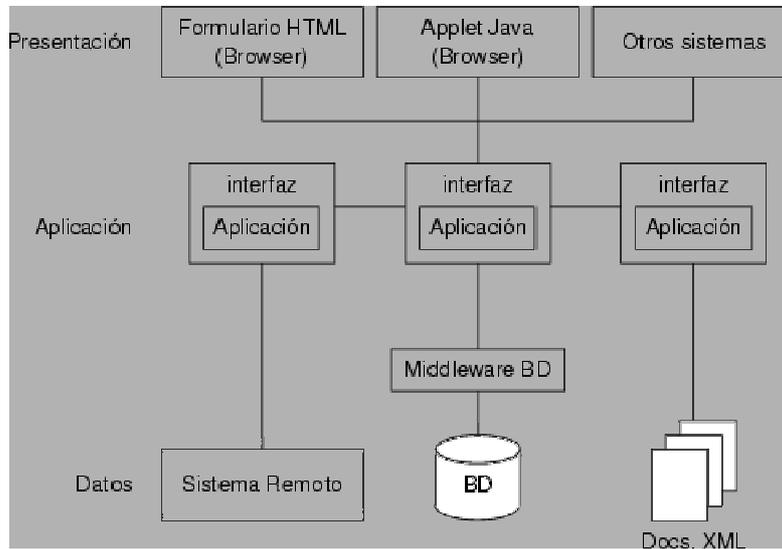
- Una solución de tres capas (presentación, lógica, datos) que residen en un solo ordenador (Presentación+lógica+datos). Se dice, que la arquitectura de la solución es de tres capas y un nivel.
- Una solución de tres capas (presentación, lógica, datos) que residen en dos ordenadores (presentación+lógica, lógica+datos). Se dice que la arquitectura de la solución es de tres capas y dos niveles.
- Una solución de tres capas (presentación, lógica, datos) que residen en tres ordenadores (presentación, lógica, datos). La arquitectura que la define es: solución de tres capas y tres niveles.

[http://es.wikipedia.org/wiki/Arquitectura\\_de\\_tres\\_niveles](http://es.wikipedia.org/wiki/Arquitectura_de_tres_niveles)

La capa intermedia es el código que el usuario invoca para recuperar los datos deseados. La capa de presentación recibe los datos y los formatea para mostrarlos adecuadamente. Esta división entre la capa de presentación y la de la lógica permite una gran flexibilidad a la hora de construir aplicaciones, ya que se pueden tener múltiples interfaces sin cambiar la lógica de la aplicación.

La tercera capa consiste en los datos que gestiona la aplicación. Estos datos pueden ser cualquier fuente de información como una base de datos o documentos XML.

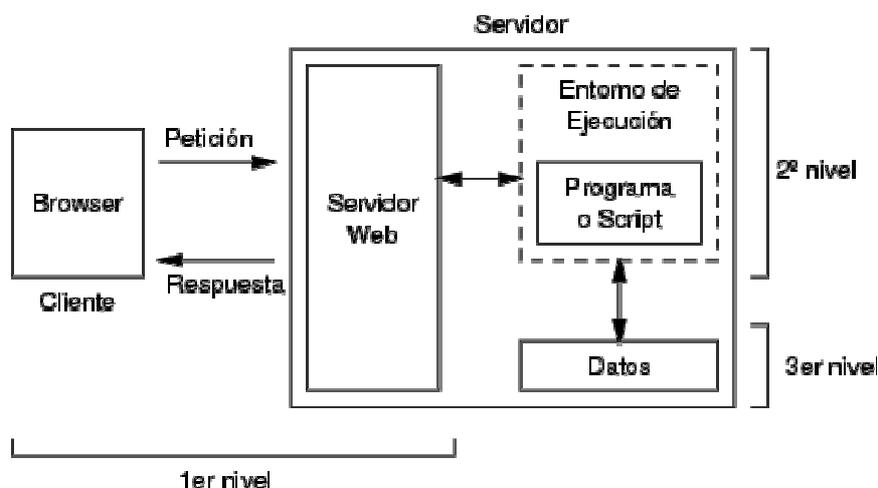
Convertir un sistema de tres niveles a otro multinivel es fácil ya que consiste en extender la capa intermedia permitiendo que convivan múltiples aplicaciones en lugar de una sola.



**Figura:** Arquitectura Multinivel.

La arquitectura de las aplicaciones Web suelen presentar un esquema de tres niveles. El primer nivel consiste en la capa de presentación que incluye no sólo el navegador, sino también el servidor web que es el responsable de dar a los datos un formato adecuado. El segundo nivel está referido habitualmente a algún tipo de programa o *script*. Finalmente, el tercer nivel proporciona al segundo los datos necesarios para su ejecución.

Una aplicación Web típica recogerá datos del usuario (primer nivel), los enviará al servidor, que ejecutará un programa (segundo y tercer nivel) y cuyo resultado será formateado y presentado al usuario en el navegador (primer nivel otra vez).

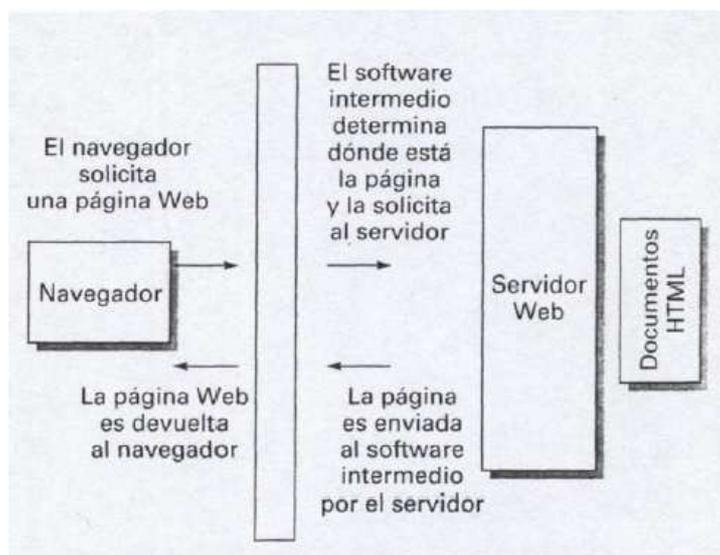


**Figura:** Arquitectura Web de tres niveles.

<http://www.infor.uva.es/~jvegas/cursos/buendia/pordocente/node21.html>

#### 2.4.2 SOFTWARE INTERMEDIO (Middleware)

Hasta el momento probablemente tengamos la impresión de que la comunicación entre cliente y servidor es directa. Desgraciadamente, esto no es verdad: normalmente existe por lo menos una capa de software entre ellos. Esta capa se llama software intermedio (*middleware*). Esta muestra la comunicación entre un cliente ejecutando un navegador como Internet Explorer y un servidor Web.



Fuente: <http://www.desarrolloweb.com/articulos/middleware>

Aquí el software intermedio que se encuentra entre el servidor Web y el cliente que ejecuta el navegador Web intercepta las peticiones que proceden del navegador. Si se hace una petición para una página Web entonces determina la localización del documento Web y envía una petición para esa página. El servidor responde a la petición y devuelve la página al software intermedio, quien la dirige al navegador que la visualizará en la pantalla del monitor que utiliza el cliente.

Existen dos categorías de software intermedio: el software intermedio general y el software intermedio de servicios.

El software intermedio general es el que está asociado a los servicios generales que requieren todos los clientes y servidores. El software típico que se utiliza como tal es:

- El software para llevar a cabo comunicaciones utilizando el protocolo TCP/IP y otros protocolos de red.
- El software del sistema operativo que por ejemplo, mantiene un almacenamiento distribuido de archivos.
- El software de autenticación, el cual comprueba que un usuario que desee utilizar un sistema distribuido pueda en efecto hacerlo.
- El software intermedio orientado a mensajes que gestiona el envío de mensajes desde clientes a servidores y viceversa.

El software intermedio de servicios es el software asociado a un servicio en particular. Entre los ejemplos típicos de este tipo de software se incluyen:

- Un software que permite a bases de datos diferentes conectarse a una red cliente/servidor.
- Un software específico de objetos distribuidos.
- Un software intermedio de red asociado al protocolo http
- Un software intermedio asociado a productos de seguridad específicos.

*(Enciclopedia Wikipedia, 2005 )*

<http://www.monografias.com/trabajos16/sistemas-distribuidos/sistemas-distribuidos.shtml#MIDDLE>

## **2.5 ASPECTOS METODOLÓGICOS**

### **2.5.1 PARADIGMA ESPIRAL ORIENTADO A LA WEB**

A medida que la evolución de las WebApps (Sistemas de aplicación basadas en Web) pasa de utilizar recursos estáticos de información controlada por el contenido a utilizar entornos de aplicación dinámicos controlados por el usuario, cada vez es más importante la necesidad de implementar una gestión sólida y unos principios de ingeniería. Para conseguir esto, es necesario desarrollar un marco de trabajo IWeb (Ingeniería Web) que acompañe a un modelo de proceso eficaz, popularizado por las actividades del marco de trabajo y por las tareas de Ingeniería.

El modelo en espiral trata de desarrollar incrementalmente el proyecto, dividiéndolo en muchos subproyectos. Uno de los puntos más importantes del proceso es concentrarse primero en los aspectos críticos del proyecto. La idea es definir e implementar las características más importantes primero, y con el conocimiento adquirido para hacerlo, volver hacia atrás y reimplementar las características siguientes en pequeños subproyectos.

El modelo en espiral orientado a la web se divide en un número de actividades estructurales, también llamadas regiones de tareas. Generalmente, existen entre tres y seis regiones de tareas.

### **2.5.2 COMUNICACIÓN CON EL CLIENTE O FORMULACIÓN**

Las tareas requeridas para establecer comunicación entre el desarrollador y el cliente. Actividad que identifica las metas y los objetivos de la WebApp, y establece el ámbito del primer incremento.

### **2.5.3 PLANIFICACIÓN**

La planificación estima el coste global del proyecto, evalúa los riesgos asociados con el esfuerzo del desarrollo, y define una planificación del desarrollo bien granulada para el incremento final de la WebApp, con una aplicación mas toscamente granulada para los incrementos subsiguientes. Son todos los requerimientos.

### **2.5.4 ANÁLISIS DE RIESGOS**

Establece los requerimientos técnicos para la WebApp e identifica los elementos del contenido que se van a incorporar. También se define los requisitos de diseño gráfico (estética). Es decir se identifica los datos y requisitos funcionales y de comportamiento para la aplicación web.

### **2.5.5 INGENIERÍA**

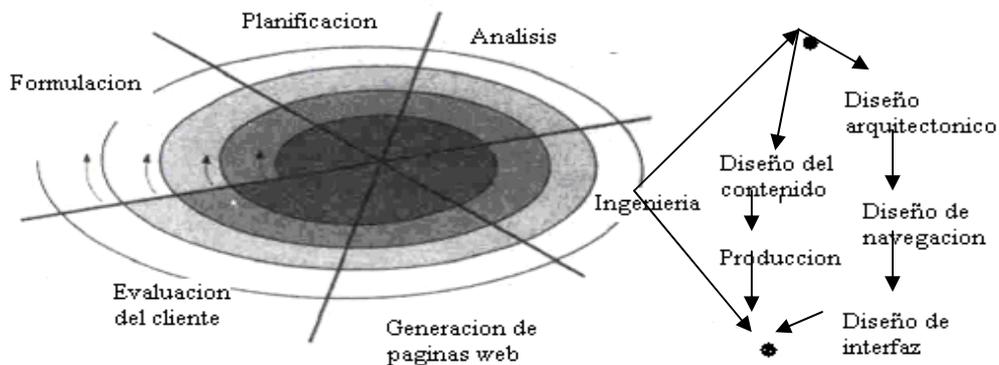
Aquí se incorpora dos tareas paralelas, el Diseño del Contenido y la Producción, son tareas llevadas a cabo por personas no técnicas del equipo IWeb. El objetivo de estas tareas es diseñar, producir y/o adquirir todo el contenido de texto, gráfico y video que se vayan a integrar en la WebApp. Al mismo tiempo se lleva a cabo un conjunto de tareas de diseño.

### **2.5.6 GENERACIÓN DE PÁGINAS**

La generación de páginas es una actividad de construcción que hace mucho uso de las herramientas automatizadas para la creación de la WebApp. El contenido definido en la actividad de ingeniería se fusiona con los diseños arquitectónicos, de navegación y de la interfaz para la elaboración de páginas Web ejecutables en HTML, XML y otros lenguajes orientados a procesos (java). Durante esta actividad también se lleva a cabo la integración con el software intermedio (Middleware) de componentes (es decir: COBRA, DCOM o JavaBEan). Las pruebas ejercitan la navegación, intentan descubrir los errores de los applets, guiones y formularios, y ayuda a asegurar que la WeApp funcionará correctamente en diferentes entornos (por ejemplo, con diferentes navegadores)

## 2.5.7 EVALUACIÓN DEL CLIENTE

En este punto es donde se solicitan cambios, tiene lugar la ampliación del ámbito.  
(Pressman, 2002)



**Figura:** El modelo de proceso Web.

## 2.6 METODOLOGÍA OOHDM

### 2.6.1 INTRODUCCIÓN A OOHDM

Las metodologías tradicionales de ingeniería de software, o las metodologías para sistemas de desarrollo de información, no contienen una buena abstracción capaz de facilitar la tarea de especificar aplicaciones hipermedia. El tamaño, la complejidad y el número de aplicaciones crecen en forma acelerada en la actualidad, por lo cual una metodología de diseño sistemática es necesaria para disminuir la complejidad y admitir evolución y usabilidad.

Producir aplicaciones en las cuales el usuario pueda aprovechar el potencial del paradigma de la navegación de sitios Web, mientras ejecuta transacciones sobre bases de información, es una tarea muy difícil de lograr. En primer lugar, la navegación posee algunos problemas. Una estructura de navegación robusta es una de las claves del éxito en las aplicaciones hipermedia. Si el usuario entiende dónde puede ir y cómo llegar al lugar deseado, es una buena señal de que la aplicación ha sido bien diseñada.

Construir la interfaz de una aplicación Web es también una tarea compleja; no sólo se necesita especificar cuáles son los objetos de la interfaz que deberían ser implementados, sino también la manera en la cual estos objetos interactuarán con el resto de la aplicación. En hipermedia existen requerimientos que deben ser satisfechos en un entorno de desarrollo unificado.

Por un lado, la navegación y el comportamiento funcional de la aplicación deberían ser integrados. Por otro lado, durante el proceso de diseño se debería poder desacoplar las decisiones de diseño relacionadas con la estructura navegacional de la aplicación, de aquellas relacionadas con el modelo del dominio.

[http://www.unab.edu.co/editorialunab/revistas/rcc/pdfs/r22\\_art5\\_c.pdf](http://www.unab.edu.co/editorialunab/revistas/rcc/pdfs/r22_art5_c.pdf)

## **2.6.2 FASES DE LA METODOLOGÍA OOHDM**

La metodología aplicada para el desarrollo del proyecto y propuesta, es la OOHDM (Object – Oriented Hypermedia Design Method, Método de Diseño Hipermedia Orientado a Objeto). La misma fue seleccionada ya que satisface los requerimientos de la aplicación, especialmente en lo referente al desarrollo Web.

Captura de Requerimientos.

Diseño Conceptual.

Diseño Navegacional.

Diseño Abstracto de Interfaz.

Implementación.

Pruebas.

<http://edutec2004.lmi.ub.es/pdf/157.pdf>

### **Captura de requerimientos**

La herramienta en la cual se fundamenta esta fase son los diagramas de casos de usos, los cuales son diseñados por escenarios con la finalidad de obtener de manera clara los requerimientos y acciones del sistema.

Según German (2003) primero que todo es necesaria la recopilación de requerimientos. En este punto, se hace necesario identificar los actores y las tareas que ellos deben realizar. Luego, se determinan los escenarios para cada tarea y tipo de actor.

Los casos de uso que surgen a partir de aquí, serán luego representados mediante los Diagramas de Interacción de Usuario (UIDs), los cuales proveen de una representación gráfica concisa de la interacción entre el usuario y el sistema durante la ejecución de alguna tarea. Con este tipo de diagramas se capturan los requisitos de la aplicación de manera independiente de la implementación.

Ésta es una de las fases más importantes, debido a que es aquí donde se realiza la recogida de datos, para ello se deben de proporcionar las respuestas a las siguientes interrogantes:

- ¿Cuáles son los tópicos principales que serán atendidos?
- ¿Cómo los tópicos están relacionados entre sí?
- ¿Qué categoría de usuarios serán atendidos?
- ¿Cuáles son las tareas principales que serán abordadas?
- ¿Qué tareas corresponden a qué categoría de usuarios?
- ¿Los recursos disponibles son competitivos con la información levantada?

Con las preguntas mencionadas anteriormente, se puede recaudar de cierta manera las bases necesarias para la construcción de una aplicación hipertextual exitosa, sin embargo mientras mayor sea el nivel de profundidad de la recolección de datos, mayor probabilidad de realizar una aplicación adecuada a las necesidades de los usuarios.

### **Diseño conceptual**

Durante esta actividad se construye un esquema conceptual representado por los objetos del dominio, las relaciones y colaboraciones existentes establecidas entre ellos. En las aplicaciones hipertextuales convencionales, cuyos componentes de

hipermedia no son modificados durante la ejecución, se podría usar un modelo de datos semántico estructural (como el modelo de entidades y relaciones). De este modo, en los casos en que la información base pueda cambiar dinámicamente o se intenten ejecutar cálculos complejos, se necesitará enriquecer el comportamiento del modelo de objetos.

En OOHDM, el esquema conceptual está construido por clases, relaciones y subsistemas. Las clases son descritas como en los modelos orientados a objetos tradicionales. Sin embargo, los atributos pueden ser de múltiples tipos para representar perspectivas diferentes de las mismas entidades del mundo real.

Se usa notación similar a UML (Lenguaje de Modelado Unificado).

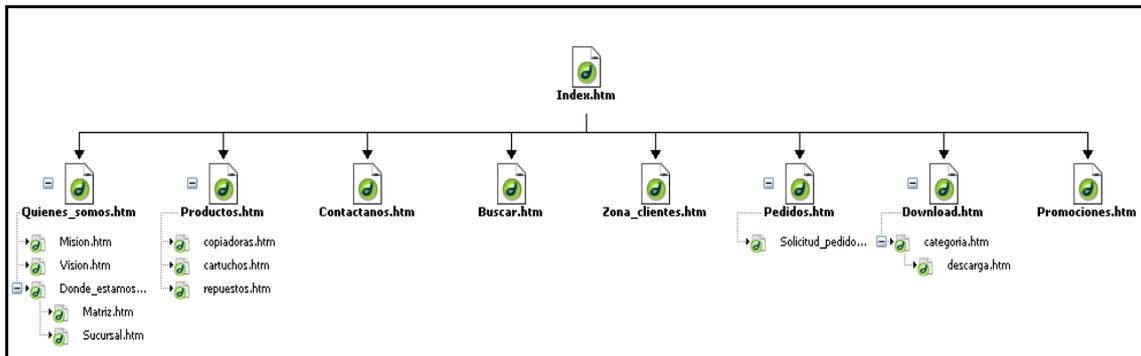
<http://edutec2004.lmi.ub.es/pdf/157.pdf>

### **Diseño navegacional**

La primera generación de aplicaciones web fue pensada para realizar navegación a través del espacio de información, utilizando un simple modelo de datos de hipermedia. En OOHDM, la navegación es considerada un paso crítico en el diseño aplicaciones. Un modelo navegacional es construido como una vista sobre un diseño conceptual, admitiendo la construcción de modelos diferentes de acuerdo con los diferentes perfiles de usuarios. Cada modelo navegacional provee una vista subjetiva del diseño conceptual.

El diseño de navegación es expresado en dos esquemas: el esquema de clases navegacionales y el esquema de contextos navegacionales. En OOHDM existe un conjunto de tipos predefinidos de clases navegacionales: nodos, enlaces y estructuras de acceso. La semántica de los nodos y los enlaces son las tradicionales de las aplicaciones hipermedia, y las estructuras de acceso, tales como índices o recorridos guiados, representan los posibles caminos de acceso a los nodos.

### **GRÁFICO 2 .6**



La principal estructura primitiva del espacio navegacional es la noción de contexto navegacional. Un contexto navegacional es un conjunto de nodos, enlaces, clases de contextos, y otros contextos navegacionales (contextos anidados). Pueden ser definidos por comprensión o extensión, o por enumeración de sus miembros.

Los contextos navegacionales juegan un rol similar a las colecciones y fueron inspirados sobre el concepto de contextos anidados. Organizan el espacio navegacional en conjuntos convenientes que pueden ser recorridos en un orden particular y que deberían ser definidos como caminos para ayudar al usuario a lograr la tarea deseada.

Los nodos son enriquecidos con un conjunto de clases especiales que permiten de un nodo observar y presentar atributos (incluidos las anclas), así como métodos (comportamiento) cuando se navega en un particular contexto.

[http://www.unab.edu.co/editorialunab/revistas/rcc/pdfs/r22\\_art5\\_c.pdf](http://www.unab.edu.co/editorialunab/revistas/rcc/pdfs/r22_art5_c.pdf)

### **Diseño abstracto de interfaz**

Una vez que las estructuras navegacionales son definidas, se deben especificar los aspectos de interfaz. Esto significa definir la forma en la cual los objetos navegacionales pueden aparecer, cómo los objetos de interfaz activarán la navegación y el resto de la funcionalidad de la aplicación, qué transformaciones de la interfaz son pertinentes y cuándo es necesario realizarlas.

Una clara separación entre diseño navegacional y diseño de interfaz abstracta permite construir diferentes interfaces para el mismo modelo navegacional, dejando un alto grado de independencia de la tecnología de interfaz de usuario.

El aspecto de la interfaz de usuario de aplicaciones interactivas (en particular las aplicaciones web) es un punto crítico en el desarrollo que las modernas metodologías tienden a descuidar. En OOHDM se utiliza el diseño de interfaz abstracta para describir la interfaz del usuario de la aplicación de hipermedia.

El modelo de interfaz ADVs (Vista de Datos Abstracta) especifica la organización y comportamiento de la interfaz, pero la apariencia física real o de los atributos, y la disposición de las propiedades de las ADVs en la pantalla real son hechas en la fase de implementación.

[http://www.unab.edu.co/editorialunab/revistas/rcc/pdfs/r22\\_art5\\_c.pdf](http://www.unab.edu.co/editorialunab/revistas/rcc/pdfs/r22_art5_c.pdf)

### **Implementación**

En esta fase, el diseñador debe implementar el diseño. Según Schwabe y Rossi. (1998), hasta ahora todos los modelos fueron construidos en forma independiente de la plataforma de implementación; en esta fase es tenido en cuenta el entorno particular en el cual se va a correr la aplicación.

Al llegar a esta fase, el primer paso que debe realizar el diseñador es definir los ítems de información que son parte del dominio del problema. Debe identificar también, cómo son organizados los ítems de acuerdo con el perfil del usuario y su tarea; decidir qué interfaz debería ver y cómo debería comportarse.

A fin de implementar todo en un entorno web, el diseñador debe decidir además qué información debe ser almacenada. Es de especial importancia el hacer notar que hoy en día, hay muchos y varios ambientes de implementación, con características distintas. Es claro, por ejemplo, que no se puede usar el mismo conjunto de líneas de acción en la traducción de un proyecto OOHDM para un documento HTML que para un programa en Macromedia Flash.

### **Pruebas**

En este caso a medida que se codifiquen las tareas, se van a ir aplicando las pruebas unitarias a cada una de ellas, con el fin de ir constatando el buen

funcionamiento de los módulos, para luego poder ensamblar el sistema y así realizar las pruebas definitivas con el cliente.

### **Ventajas de OOHDM**

De acuerdo con Silva y Mercerat (2001) OOHDM como metodología de desarrollo de aplicaciones de hipermedia, proporciona ventajas como:

- La recuperación de la información puede realizarse sin problemas.
- Se pueden crear enlaces entre nodos cualesquiera.
- La modularidad y la consistencia se potencian.
- Marco idóneo para la autoría en colaboración.
- Soporte a diferentes modos de acceso a la información.

En la actualidad, el desarrollo de software empleando patrones de proyecto, se encuentra en crecimiento según Gamma, Helm, Johnson y Vlissides (1995), sin embargo, su potencial se encuentra inexplorado en el campo de hipermedios, especialmente a la hora de describir las arquitecturas para la navegación e interface en aplicaciones de hipermedia.

OOHDM según Schwabe y Rossi (1998) se torna diferente y superior a otras metodologías de desarrollo de aplicaciones de hipermedia al ofrecer la ventaja de patrones de proyecto poderosos como primitivas para la construcción del modelo navegacional de una aplicación hipermedia.

Puede decirse con base en todas las ventajas antes mencionadas que OOHDM toma en cuenta las crecientes necesidades de analistas y programadores de aplicaciones hipermedia, y se presenta como una técnica ideal de desarrollo para la producción de aplicaciones evolutivas de alta calidad.

## **2.7 UML (Unified Modeling Language)**

El Lenguaje de Modelado Unificado (UML) es la sucesión de una serie de métodos de análisis y diseño orientadas a objetos que aparecen a fines de los 80's y principios de los 90s. Directamente unifica los métodos de Booch, Rumbaugh (OMT), y Jacobson, y algo más.

UML es llamado un lenguaje de modelado, no un método. Los métodos consisten de ambos de un lenguaje de modelado y de un proceso.

<http://agamenon.uniandes.edu.co/~pfigueroa/soo/uml>

UML Es un lenguaje de modelado visual que se usa para especificar, visualizar, construir y documentar artefactos de un sistema de software. Se usa para entender, diseñar, configurar, mantener y controlar la información sobre los sistemas a construir.

UML capta la información sobre la estructura estática y el comportamiento dinámico de un sistema. Un sistema se modela como una colección de objetos discretos que interactúan para realizar un trabajo que finalmente beneficia a un usuario externo.

El lenguaje de modelado pretende unificar la experiencia pasada sobre técnicas de modelado e incorporar las mejores prácticas actuales en un acercamiento estándar.

UML no es un lenguaje de programación. Las herramientas pueden ofrecer generadores de código de UML para una gran variedad de lenguaje de programación, así como construir modelos por ingeniería inversa a partir de programas existentes.

Es un lenguaje de propósito general para el modelado orientado a objetos. UML es también un lenguaje de modelamiento visual que permite una abstracción del sistema y sus componentes.

Existían diversos métodos y técnicas Orientadas a Objetos, con muchos aspectos en común pero utilizando distintas notaciones, se presentaban inconvenientes

para el aprendizaje, aplicación, construcción y uso de herramientas, etc., además de pugnas entre enfoques, lo que generó la creación del UML como estándar para el modelamiento de sistemas de software principalmente, pero con posibilidades de ser aplicado a todo tipo de proyectos.

<http://www.creangel.com/uml/intro.php>

## **2.7.1 DIAGRAMAS PARA MODELAMIENTO**

### **2.7.1.1 Diagramas de Caso de Uso**

Objeto es una entidad discreta con límites bien definidos y con identidad, es una unidad atómica que encapsula estado y comportamiento. La encapsulación en un objeto permite una alta cohesión y un bajo acoplamiento. El Objeto es reconocido también como una instancia de la clase a la cual pertenece. La encapsulación presenta tres ventajas básicas:

- Se protegen los datos de accesos indebidos
- El acoplamiento entre las clases se disminuye
- Favorece la modularidad y el mantenimiento

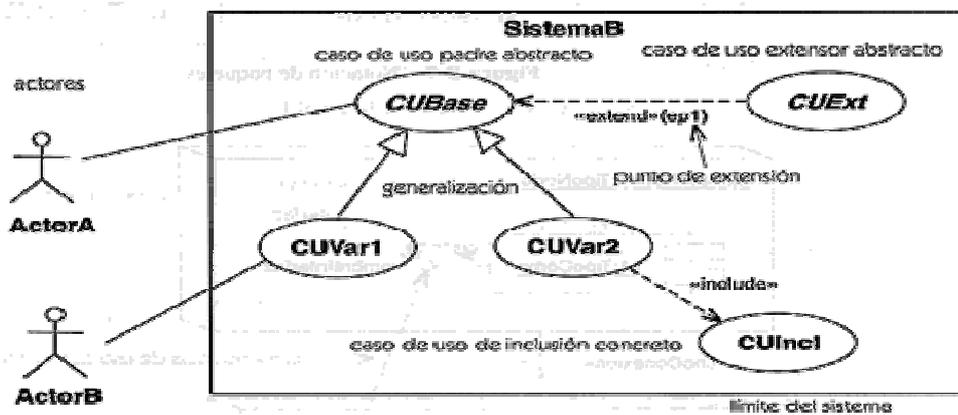
Un objeto se puede ver desde dos perspectivas relacionadas: como una entidad de un determinado instante de tiempo que posee un valor específico (Un objeto puede caracterizar una entidad física -coche-) y como un poseedor de identidad que tiene distintos valores a lo largo del tiempo (abstracta-ecuación matemática).

El Modelado de Objetos permite representar el ciclo de vida de los objetos a través de sus interacciones. En UML, un objeto se representa por un rectángulo con un nombre subrayado.

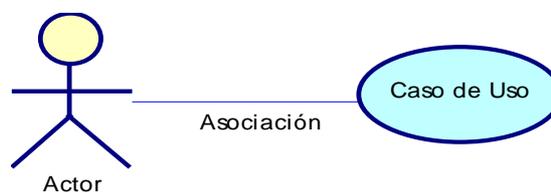
- Objeto = Identidad + Estado + Comportamiento.

- El estado está representado por los valores de los atributos.
- Un atributo toma un valor en un dominio concreto.

La regla general para la notación de instancias consiste en utilizar el mismo símbolo geométrico que el descriptor. En la instancia se muestran los posibles valores pero las propiedades compartidas sólo se pone de manifiesto en el descriptor. La notación canónica es un rectángulo con tres compartimientos. En el primero va el nombre del objeto, en el segundo sus atributos y en el tercero sus operaciones. Este último puede ser omitido si así se prefiere.



**Notación del Diagrama de Caso de uso.-** Un diagrama de caso de uso es una gráfica de actores, un conjunto de casos de uso, posiblemente algunas interfaces y las relaciones entre estos elementos. Las relaciones son asociaciones entre los actores y los casos de uso generalizaciones entre los actores y generalizaciones, extensiones e inclusiones entre los casos de uso.



**Relaciones de los Casos de uso.-** Hay varias relaciones estándar entre los casos de uso o entre los actores y los casos de uso.

- Asociación ( \_\_\_\_ ). La participación de un actor en el Caso de uso, i. e. instancias de el actor e instancias de el caso de uso se comunican entre si. **Esta es la única relación entre los actores y los Casos de uso.**
- Extensión ( \_\_e\_\_ ). Una relación de extensión entre el caso de uso A y el caso de uso B indica que una instancia de el caso de uso B puede ser aumentada (Dependiendo de ciertas condiciones especificadas en la extensión) por el comportamiento de especificado en el Caso de uso B. El comportamiento es insertado el punto definido como punto de extensión del Caso de uso B, el cual es referenciado por la relación externa.
- Generalización ( ← ) ( Una generalización de un caso de uso A hacia el caso de uso B indica que A es una especialización de B
- Inclusión ( \_\_i\_\_ ). Una relación de inclusión del caso de uso A hacia el Caso de uso B indica que una instancia del caso de uso A también contendrá el comportamiento especificado por B. El comportamiento es incluido en el punto definido en A.

<http://www.geocities.com/txmetsb/#Que es un Use Case>

### 2.7.1.2 Diagrama de Clases

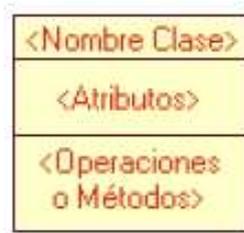
Un diagrama de clases sirve para visualizar las relaciones entre las clases que involucran el sistema, las cuales pueden ser asociativas, de herencia, de uso y de contenido.

Un diagrama de clases esta compuesto por los siguientes elementos:

- Clase: atributos, métodos y visibilidad.
- Relaciones: Herencia, Composición, Agregación, Asociación y Uso.

*Clase*.- Es la unidad básica que encapsula toda la información de un Objeto (un objeto es una instancia de una clase). A través de ella podemos modelar el entorno en estudio (una Casa, un Auto, una Cuenta Corriente, etc.).

En UML, una clase es representada por un rectángulo que posee tres divisiones:



En donde:

**Superior:** Contiene el nombre de la Clase

**Intermedio:** Contiene los atributos (o variables de instancia) que caracterizan a la Clase (pueden ser private, protected o public).

**Inferior:** Contiene los métodos u operaciones, los cuales son la forma como interactúa el objeto con su entorno (dependiendo de la visibilidad: private, protected o public).

**Atributos:** Los atributos o características de una Clase pueden ser de tres tipos, los que definen el grado de comunicación y visibilidad de ellos con el entorno, estos son:

**Public** (+, 

**Private** (-, 

**Protected** (#, 

**Métodos:** Los métodos u operaciones de una clase son la forma en como ésta interactúa con su entorno, éstos pueden tener las características:

**Public** (+,

**Private** (-, ): Indica que el método sólo será accesible desde dentro de la clase (sólo otros métodos de la clase lo pueden acceder).

**Protected** (#, ): Indica que el método no será accesible desde fuera de la clase, pero si podrá ser accesado por métodos de la clase además de métodos de las subclases que se deriven (ver herencia).

<http://www.clikear.com/manuales/uml/diagramasestructuraestatica.asp>

**Relaciones entre Clases:** Ahora ya definido el concepto de Clase, es necesario explicar como se pueden interrelacionar dos o más clases (cada uno con características y objetivos diferentes).

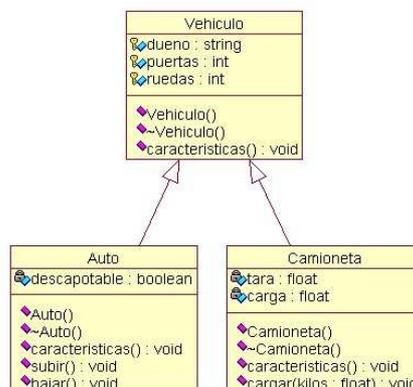
Antes es necesario explicar el concepto de cardinalidad de relaciones:

En UML, la cardinalidad de las relaciones indica el grado y nivel de dependencia, se anotan en cada extremo de la relación y éstas pueden ser:

- **uno o muchos:** 1..\* (1..n)
- **0 o muchos:** 0..\* (0..n)
- **número fijo:** m (m denota el número).

**Herencia (Especialización/Generalización):** 

Indica que una subclase hereda los métodos y atributos especificados por una Super Clase, por ende la Subclase además de poseer sus propios métodos y atributos, poseerá las características y atributos visibles de la Super Clase (public y protected), ejemplo:



En la figura se especifica que Auto y Camión heredan de Vehículo, es decir, Auto posee las Características de Vehículo (Precio, VelMax, etc) además posee algo particular que es Descapotable, en cambio Camión también hereda las características de Vehículo (Precio, VelMax, etc) pero posee como particularidad propia Acoplado, Tara y Carga.

Cabe destacar que fuera de este entorno, lo único "visible" es el método Características aplicable a instancias de Vehículo, Auto y Camión, pues tiene definición pública, en cambio atributos como Descapotable no son visibles por ser privados.

**Agregación:** 

Para modelar objetos complejos, no bastan los tipos de datos básicos que proveen los lenguajes: enteros, reales y secuencias de caracteres. Cuando se requiere componer objetos que son instancias de clases definidas por el desarrollador de la aplicación, tenemos dos posibilidades:

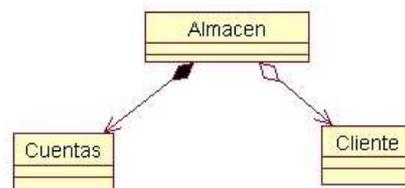
**Por Valor:** Es un tipo de relación estática, en donde el tiempo de vida del objeto incluido esta condicionado por el tiempo de vida del que lo incluye. Este tipo de relación es comúnmente llamada:

*Composición* (el Objeto base se construye a partir del objeto incluido, es decir, es "parte/todo").

*Por Referencia:* Es un tipo de relación dinámica, en donde el tiempo de vida del objeto incluido es independiente del que lo incluye. Este tipo de relación es comúnmente llamada

*Agregación* (el objeto base utiliza al incluido para su funcionamiento).

Un Ejemplo es el siguiente:



En donde se destaca que:

Un almacén posee Clientes y Cuentas (los rombos van en el objeto que posee las referencias).

Cuando se destruye el Objeto almacén también son destruidos los objetos Cuenta asociados, en cambio no son afectados los objetos Cliente asociados.

La composición (por Valor) se destaca por un rombo relleno.

La agregación (por Referencia) se destaca por un rombo transparente.

La flecha en este tipo de relación indica la navegabilidad del objeto referenciado.

Cuando no existe este tipo de particularidad la flecha se elimina.

### Asociación:

La relación entre clases conocida como Asociación, permite asociar objetos que colaboran entre si. Cabe destacar que no es una relación fuerte, es decir, el tiempo de vida de un objeto no depende del otro.

Ejemplo:



Un cliente puede tener asociadas muchas Ordenes de Compra, en cambio una orden de compra solo puede tener asociado un cliente.

### Dependencia o Instanciación (uso):

Representa un tipo de relación muy particular, en la que una clase es instanciada (su instanciación es dependiente de otro objeto/clase). Se denota por una flecha punteada.

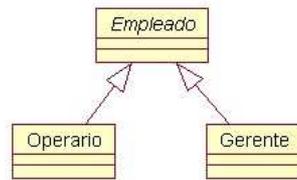
El uso más particular de este tipo de relación es para denotar la dependencia que tiene una clase de otra, como por ejemplo una aplicación grafica que instancia una ventana (la creación del Objeto Ventana esta condicionado a la instanciación proveniente desde el objeto Aplicación):



Cabe destacar que el objeto creado (en este caso la Ventana gráfica) no se almacena dentro del objeto que lo crea (en este caso la Aplicación).

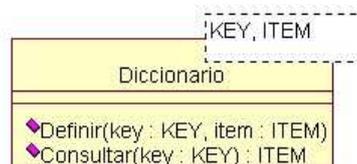
### Casos Particulares:

#### Clase Abstracta:



Una clase abstracta se denota con el nombre de la clase y de los métodos con letra "itálica". Esto indica que la clase definida no puede ser instanciada pues posee métodos abstractos (aún no han sido definidos, es decir, sin implementación). La única forma de utilizarla es definiendo subclases, que implementan los métodos abstractos definidos.

### Clase parametrizada:



Una clase parametrizada se denota con un subcuadro en el extremo superior de la clase, en donde se especifican los parámetros que deben ser pasados a la clase para que esta pueda ser instanciada. El ejemplo más típico es el caso de un Diccionario en donde una llave o palabra tiene asociado un significado, pero en este caso las llaves y elementos pueden ser genéricos.

<http://www.dcc.uchile.cl/~psalinas/uml/modelo.html>

### 2.7.1.3 Diagrama de Interacción

Los diagramas de interacción, representan la forma en como un Cliente (Actor) u Objetos (Clases) se comunican entre si en petición a un evento. Esto implica recorrer toda la secuencia de llamadas, de donde se obtienen las responsabilidades claramente.

Dichos diagramas pueden ser obtenidos de dos partes, desde el Diagrama Estático de Clases o el de Casos de Uso (son diferentes).

Los componentes de un diagrama de interacción son:

- Un Objeto o Actor.
- Mensaje de un objeto a otro objeto.

- Mensaje de un objeto a si mismo.

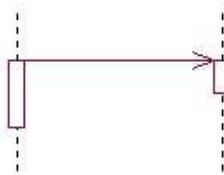
**Objeto/Actor:**



El rectángulo representa una instancia de un Objeto en particular, y la línea punteada representa las llamadas a métodos del objeto.

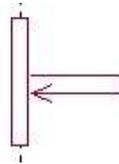
<http://www.dcc.uchile.cl/~psalinas/uml/interaccion.html>

**Mensaje a Otro Objeto:**



Se representa por una flecha entre un objeto y otro, representa la llamada de un método (operación) de un objeto en particular.

**Mensaje al Mismo Objeto:**



No solo llamadas a métodos de objetos externos pueden realizarse, también es posible visualizar llamadas a métodos desde el mismo objeto en estudio.

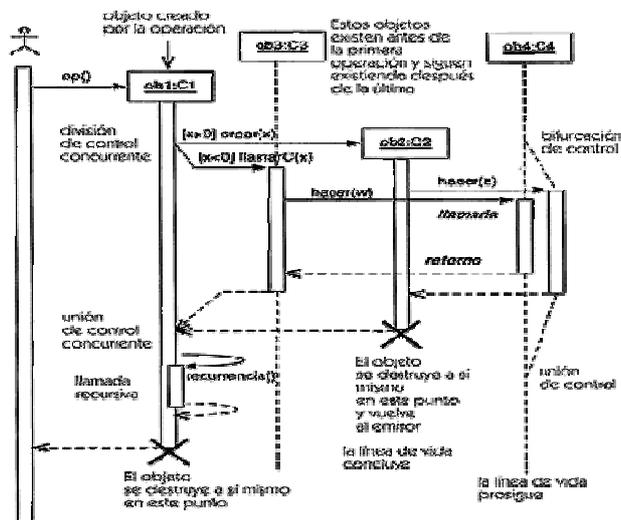
Hay dos tipos de diagrama de interacción, ambos basados en la misma información, pero cada uno enfatizando un aspecto particular: Diagramas de Secuencia y Diagramas de Colaboración.

**Diagrama de Secuencia.-** Un diagrama de Secuencia muestra una interacción ordenada según la secuencia temporal de eventos. En particular, muestra los objetos participantes en la interacción y los mensajes que intercambian ordenados según su secuencia en el tiempo. El eje vertical representa el tiempo, y en el eje

horizontal se colocan los objetos y actores participantes en la interacción, sin un orden prefijado. Cada objeto o actor tiene una línea vertical, y los mensajes se representan mediante flechas entre los distintos objetos. El tiempo fluye de arriba abajo. Se pueden colocar etiquetas (como restricciones de tiempo, descripciones de acciones, etc.) bien en el margen izquierdo o bien junto a las transiciones o activaciones a las que se refieren.

El diagrama de Secuencia Muestra las interacciones entre los objetos organizadas en una secuencia temporal. En particular muestra los objetos participantes en la interacción y la secuencia de mensajes intercambiados.

<http://www.clikear.com/manuales/uml/diagramainteraccion.asp>

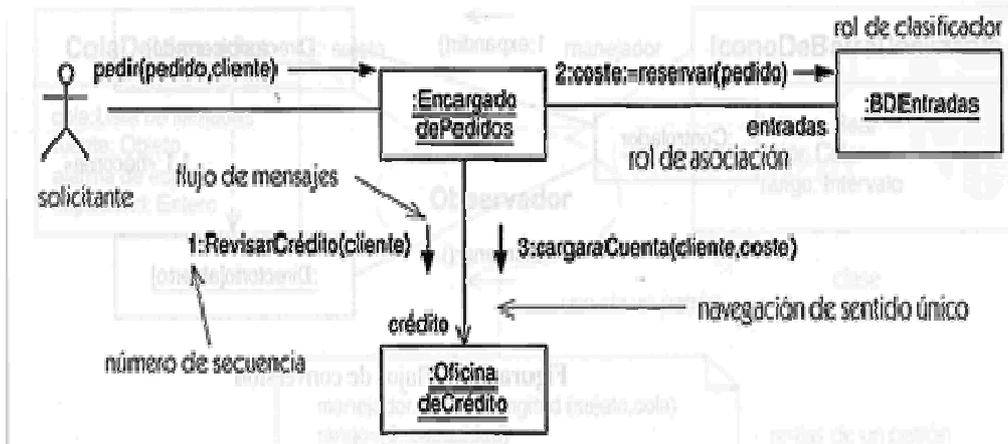


<http://www.creangel.com/uml/secuencia.php#>

**Diagrama de Colaboración.-** Un Diagrama de Colaboración muestra una interacción organizada basándose en los objetos que toman parte en la interacción y los enlaces entre los mismos (en cuanto a la interacción se refiere). A diferencia de los Diagramas de Secuencia, los Diagramas de Colaboración muestran las relaciones entre los roles de los objetos. La secuencia de los mensajes y los flujos de ejecución concurrentes deben determinarse explícitamente mediante números de secuencia.

Un Diagrama de Colaboración muestra interacciones organizadas alrededor de los roles. A diferencia de los diagramas de secuencia, los diagramas de colaboración muestran explícitamente las relaciones de los roles.

<http://www.dcc.uchile.cl/~psalinas/uml/interaccion.html>



<http://www.creangel.com/uml/colaboracion.php#>

#### 2.7.1.4 Diagrama de Estados

Un Diagrama de Estados muestra la secuencia de estados por los que pasa bien un caso de uso, bien un objeto a lo largo de su vida, o bien todo el sistema. En él se indican qué eventos hacen que se pase de un estado a otro y cuáles son las respuestas y acciones que genera.

En cuanto a la representación, un diagrama de estados es un grafo cuyos nodos son estados y cuyos arcos dirigidos son transiciones etiquetadas con los nombres de los eventos. Un estado se representa como una caja redondeada con el nombre del estado en su interior. Una transición se representa como una flecha desde el estado origen al estado destino.

La caja de un estado puede tener 1 o 2 compartimentos.

- En el primer compartimento aparece el nombre del estado.
- El segundo compartimento es opcional, y en él pueden aparecer acciones de entrada, de salida y acciones internas.

Una acción de entrada aparece en la forma `entrada/acción_asociada` donde `acción_asociada` es el nombre de la acción que se realiza al entrar en ese estado.

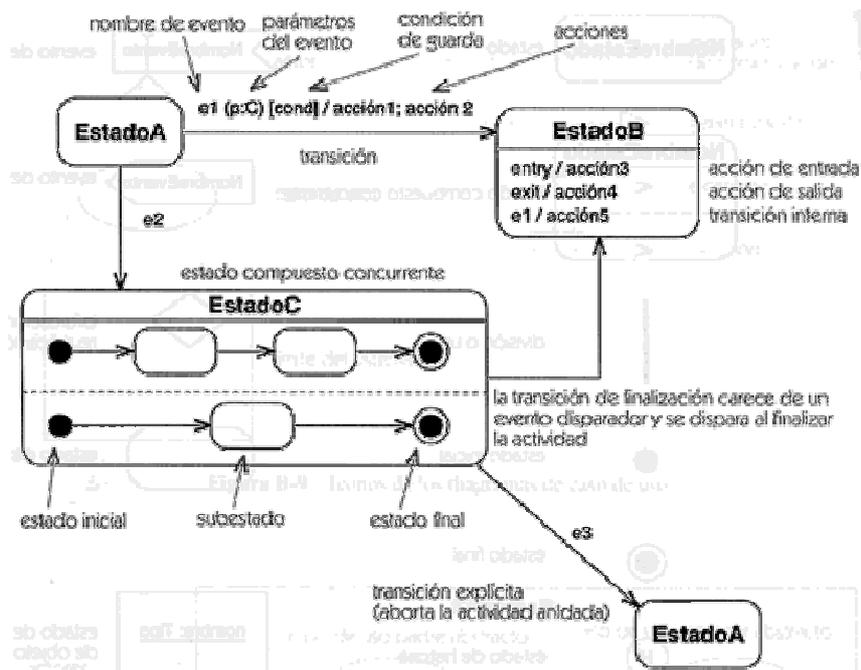
Cada vez que se entra al estado por medio de una transición la acción de entrada se ejecuta.

Una acción de salida aparece en la forma salida/acción\_asociada. Cada vez que se sale del estado por una transición de salida la acción de salida se ejecuta.

<http://es.tldp.org/Tutoriales/doc-modelado-sistemas-UML/multiple-html/x98.html>

Una acción interna es una acción que se ejecuta cuando se recibe un determinado evento en ese estado, pero que no causa una transición a otro estado. Se indica en la forma nombre\_de\_evento/acción\_asociada.

Un diagrama de estados puede representar ciclos continuos o bien una vida finita, en la que hay un estado inicial de creación y un estado final de destrucción (finalización del caso de uso o destrucción del objeto).

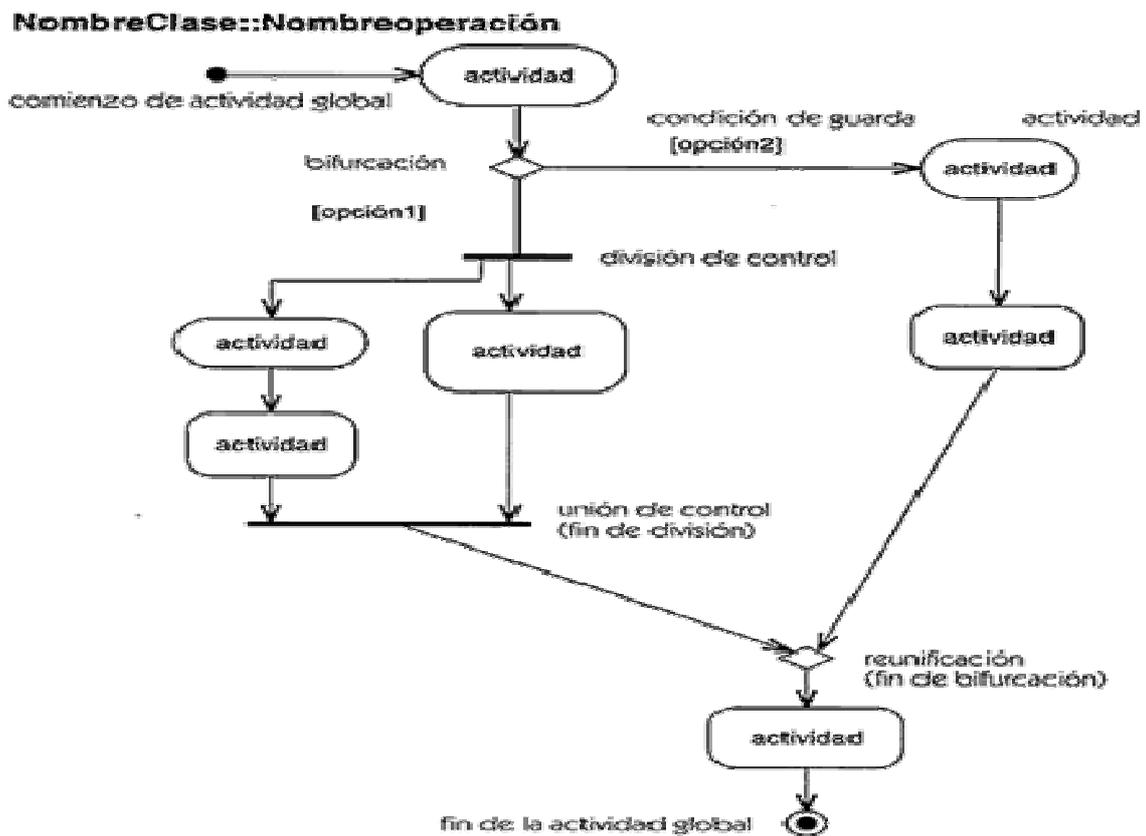


El estado inicial se muestra como un círculo sólido y el estado final como un círculo sólido rodeado de otro círculo. En realidad, los estados inicial y final son pseudoestados, pues un objeto no puede "estar" en esos estados, pero nos sirven para saber cuáles son las transacciones inicial y final(es).

### 2.7.1.5 Diagrama de Actividades

Un diagrama de actividad muestra la realización de operaciones para conseguir un objetivo. Presentan una visión simplificada de lo que ocurre en un proceso, mostrando los pasos que se realizan, constituyéndose en uno de los diagramas que modelan los aspectos dinámicos del sistema.

Los diagramas de actividad son una extensión de los diagramas de estado. Los diagramas de estado resaltan los estados y muestran las actividades que dan lugar a cambios de estado, mientras que los diagramas de actividad resaltan justamente las actividades.



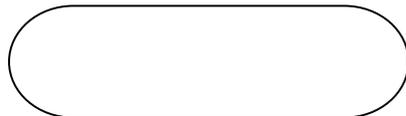
Básicamente un diagrama de actividades contiene:

- Estados de acción

- Estados de actividad
- Inicio y Terminación
- Transiciones
- Bifurcaciones
- Calles
- Señales.

**Estado de Acción.-** La representación es un rectángulo con las puntas redondeadas, en cuyo interior se representa una acción. La idea central es la siguiente:

*“Un estado que represente una acción es atómico, lo que significa que su ejecución se puede considerar instantánea y no puede ser interrumpida”*



**Estados de Actividad.-** Un estado de actividad representa la realización de una o varias tareas que causa un cambio en el sistema. La representación es un rectángulo con las puntas redondeadas

Un estado de actividad puede descomponerse en más sub-actividades representadas a través de otros diagramas de actividades. Además estos estados sí pueden ser interrumpidos y tardan un cierto tiempo en completarse.

En los estados de actividad podemos encontrar otros elementos adicionales como son: acciones de entrada (entry) y de salida (exit) del estado en cuestión.

[http://www.embarcadero.com/support/what\\_is\\_uml.asp](http://www.embarcadero.com/support/what_is_uml.asp)

**Inicio y Terminación.-** Un flujo de control debe empezar en algún sitio y terminar El inicio se representa mediante un círculo completamente relleno.

● Inicio del flujo

○ Término de flujo

Mientras que el término se representa con un círculo relleno dentro de una circunferencia.

**Transiciones.-** Las transiciones reflejan el paso de un estado a otro, bien sea de actividad o de acción. Esta transición se produce como resultado de la finalización del estado del que parte la transición.

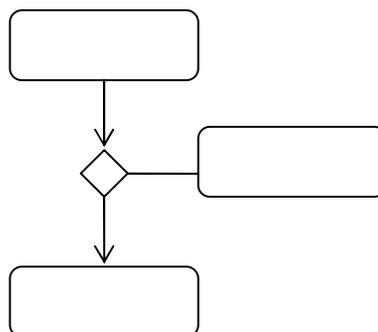
Como todo flujo de control debe empezar y terminar en algún momento, podemos indicar esto utilizando dos disparadores de inicio y fin.

Se representa mediante una línea dirigida que va de la actividad que finaliza hacia la actividad a ejecutar. Las flechas que indican la transición no deben ser etiquetadas pues son disparadas automáticamente por la finalización de la primera actividad.

**Bifurcaciones (Decisiones).-** En los diagramas de actividad se pueden incluir caminos alternativos según se cumpla alguna condición. Estas condiciones se representan mediante un rombo al cual llega la transición de la actividad origen y del cual salen las múltiples transiciones a cada actividad destino, cada una con una condición diferente.

Se pueden incluir la palabra else esto indica que ninguna de las expresiones de guarda es verdad.

Una expresión de guarda es aquella que puede tomar un valor de verdadero o falso. Las alternativas de la bifurcación han de ser excluyentes y contemplar todos los casos ya que de otro modo la ejecución quedaría interrumpida.



<http://www.monografias.com/trabajos5/insof/insof.shtml>

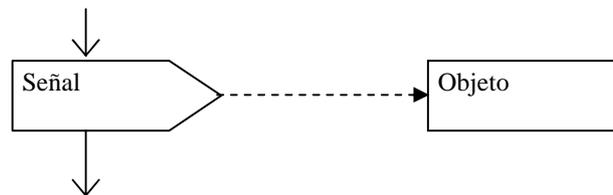
**Barras de Sincronización.-** Al modelar flujos de procesos podemos encontrar actividades que se pueden realizar simultáneamente. A una barra de sincronización se la representa como una línea vertical u horizontal gruesa.

Las barras de sincronización muestran actividades concurrentes es decir que la ejecución de procesos es al mismo tiempo; de forma paralela.

**Calles.-** Cuando se modelan flujos de trabajo de organizaciones, es especialmente útil dividir los estados de actividades en grupos, cada grupo tiene un nombre concreto y se denominan calles. Cada calle representa a la parte de la organización responsable de las actividades que aparecen en esa calle.

**Señales.-** Algunos objetos pueden enviar señales para la realización de alguna actividad. Estas señales pueden ser de dos tipos: señal de envío y señal de recibo, aunque no son indispensables para la construcción de diagramas de actividades y pueden evitarse.

**Señal de envío.-** El envío es una señal, desde nuestro diagrama de actividades hacia un objeto, puede ser mostrado como un pentágono convexo que luce como un rectángulo con un triángulo apuntando hacia fuera. El significado de esta señal es mostrada dentro del símbolo.



Esta señal se coloca dentro de dos transiciones una entrando desde una actividad y otra saliendo hacia otra actividad. Opcionalmente se puede dibujar una flecha de línea discontinua desde el pentágono convexo, hacia el objeto receptor de la señal.

**Señal de recibo.-** El recibo de una señal, por nuestro diagrama de actividades desde un objeto puede ser mostrado como un pentágono cóncavo que luce como un rectángulo con un triángulo entrante en uno de sus lados. El significado de esta señal es mostrado dentro del símbolo.



Esta señal se coloca dentro de dos transiciones una entrando desde una actividad y la otra saliendo hacia otra actividad. Opcionalmente se puede dibujar una flecha de línea discontinua desde el objeto que envía la señal hacia el pentágono cóncavo

<http://lucas.hispalinux.es/Tutoriales/doc-modelado-sistemas-UML/multiple-html/index.html>

## ***CAPITULO 3***

### **3. HERRAMIENTAS DE CONSTRUCCIÓN WEB**

En el presente capítulo se describirá las características mas importantes de las software utilizado en el desarrollo del Sitio Web, dicho software proporcionan una combinación potente como herramientas visuales para el desarrollo de aplicaciones Web.

#### **3.1 CAPA DE PRESENTACIÓN**

##### **3.1.1 JAVASCRIPT**

JavaScript es un lenguaje interpretado orientado a las páginas web basado en el paradigma prototipo, con una sintaxis semejante a la del lenguaje Java.

El lenguaje fue inventado por Brendan Eich en la empresa Netscape Communications, que es la que fabricó los primeros navegadores de Internet comerciales. Apareció por primera vez en el producto de Netscape llamado Netscape Navigator 2.0.

Tradicionalmente, se venía utilizando en páginas web HTML, para realizar tareas y operaciones en el marco de la aplicación cliente servidor.

Los autores inicialmente lo llamaron Mocha y más tarde LiveScript pero fue rebautizado como JavaScript en un anuncio conjunto entre Sun Microsystems y Netscape, el 4 de diciembre de 1995.

En 1997 los autores propusieron JavaScript para que fuera adoptado como estándar de la the European Computer Manufacturers' Association ECMA, que a pesar de su nombre no es europeo sino internacional, con sede en Ginebra. En junio de 1997 fue adoptado como un estándar ECMA, con el nombre de ECMAScript. Poco después también lo fue como un estándar ISO.

JScript es la implementación de ECMAScript de Microsoft, muy similar al JavaScript de Netscape, pero con ciertas diferencias en el modelo de objetos del navegador que hacen a ambas versiones con frecuencia incompatibles.

Para evitar estas incompatibilidades, el World Wide Web Consortium diseñó el estándar Document Object Model (DOM, ó Modelo de Objetos del Documento en castellano), que incorporan Konqueror, las versiones 6 de Internet Explorer y Netscape Navigator, Opera versión 7, y Mozilla desde su primera versión.

<http://es.wikipedia.org/wiki/JavaScript>

### **3.1.1.1 HTML**

El HTML, acrónimo inglés de HyperText Markup Language (lenguaje de marcas hipertextuales), es un lenguaje de marcación diseñado para estructurar textos y presentarlos en forma de hipertexto, que es el formato estándar de las páginas web. Gracias a Internet y a los navegadores del tipo Internet Explorer, Opera, Firefox o Netscape, el HTML se ha convertido en uno de los formatos más populares que existen para la construcción de documentos y también de los más fáciles de aprender.

HTML es una aplicación de SGML conforme al estándar internacional ISO 8879. XHTML es una reformulación de HTML 4 como aplicación XML 1.0, y que supone la base para la evolución estable de este lenguaje. Además XHTML permite la compatibilidad con los agentes de usuario que ya admitían HTML 4 siguiendo un conjunto de reglas.

#### ***Historia del estándar***

En 1989 existían dos técnicas que permitían vincular documentos electrónicos, por un lado los hipervínculos (links) y por otro lado un poderoso lenguaje de etiquetas denominado SGML. Por entonces un usuario conocedor de ambas opciones, Tim Berners-Lee físico nuclear del Centro Europeo de Investigaciones Nucleares da a conocer a la prensa que estaba trabajando en un sistema que permitirá acceder a ficheros en línea, funcionando sobre redes de computadoras o máquinas electrónicas basadas en el protocolo TCP/IP.

Principios de 1990, Tim Berners-Lee define por fin el HTML como un subconjunto del conocido SGML y crea algo más valioso aun, el World Wide Web. En 1991, Tim Berners-Lee crea el primer navegador de HTML que funcionaría en modo texto y para UNIX.

Los trabajos para crear un sucesor del HTML, posteriormente llamado 'HTML+', comenzaron a finales de 1993. El HTML+ se diseñó originalmente para ser un superconjunto del HTML que permitiera evolucionar gradualmente desde el formato HTML anterior. A la primera especificación formal de HTML+ se le dio, por lo tanto, el número de versión 2.0 para distinguirla de esos "estándares no oficiales" previos. Los trabajos sobre HTML+ continuaron, pero nunca se convirtió en un estándar.

El borrador del estándar HTML 3.0 fue propuesto por el recién formado W3C en marzo de 1995. Con él se introdujeron muchas nuevas capacidades, tales como facilidades para crear tablas, hacer que el texto fluyese alrededor de las figuras y mostrar elementos matemáticos complejos. Aunque se diseñó para ser compatible con HTML 2.0, era demasiado complejo para ser implementado con la tecnología de la época y, cuando el borrador del estándar expiró en septiembre de 1995, se abandonó debido a la carencia de apoyos de los fabricantes de navegadores web. El HTML 3.1 nunca llegó a ser propuesto oficialmente, y el estándar siguiente fue el HTML 3.2, que abandonaba la mayoría de las nuevas características del HTML 3.0 y, a cambio, adoptaba muchos elementos desarrollados inicialmente por los navegadores web Netscape y Mosaic. La posibilidad de trabajar con fórmulas matemáticas que se había propuesto en el HTML 3.0 pasó a quedar integrada en un estándar distinto llamado MathML.

El HTML 4.0 también adoptó muchos elementos específicos desarrollados inicialmente para un navegador web concreto, pero al mismo tiempo comenzó a limpiar el HTML señalando algunos de ellos como 'desaprobados'.

Ya no va a haber nuevas versiones del HTML. Sin embargo, la herencia del HTML se mantiene en XHTML, que se basa en XML.

[http://es.wikipedia.org/wiki/HTML#Historia\\_del\\_est.C3.A1ndar](http://es.wikipedia.org/wiki/HTML#Historia_del_est.C3.A1ndar)

### **3.1.2 CAPA DE NEGOCIO**

#### **3.1.2.1 PHP**

PHP es un lenguaje de programación usado generalmente para la creación de contenido para sitios web. El nombre es el acrónimo recursivo de "PHP: Hypertext

Preprocessor" (inicialmente PHP Tools, o, Personal Home Page Tools), y se trata de un lenguaje interpretado usado para la creación de aplicaciones para servidores, o creación de contenido dinámico para sitios web. Últimamente también para la creación de otro tipo de programas incluyendo aplicaciones con interfaz gráfica usando la librería GTK+.

### *Visión general*

El fácil uso y la similitud con los lenguajes más comunes de programación estructurada, como C y Perl, permiten a la mayoría de los programadores experimentados crear aplicaciones complejas con una curva de aprendizaje muy suave. También les permite involucrarse con aplicaciones de contenido dinámico sin tener que aprender todo un nuevo grupo de funciones y prácticas.

Debido al diseño de PHP, también es posible crear aplicaciones con una interfaz gráfica para el usuario (también llamada GUI), utilizando la extensión PHP-GTK. También puede ser usado desde la línea de órdenes, de la misma manera como Perl o Python pueden hacerlo, esta versión de PHP se llama PHP CLI (Command Line Interface).

Su interpretación y ejecución se da en el servidor, en el cual se encuentra almacenado el script, y el cliente sólo recibe el resultado de la ejecución. Cuando el cliente hace una petición al servidor para que le envíe una página web, generada por un script PHP, el servidor ejecuta el intérprete de PHP, el cual procesa el script solicitado que generará el contenido de manera dinámica, pudiendo modificar el contenido a enviar, y regresa el resultado al servidor, el cual se encarga de regresárselo al cliente. Además es posible utilizar PHP para generar archivos PDF, Flash, así como imágenes en diferentes formatos, entre otras cosas.

Permite la conexión a diferentes tipos de servidores de bases de datos tales como MySQL, Postgres, Oracle, ODBC, DB2, Microsoft SQL Server, Firebird y SQLite; lo cual permite la creación de Aplicaciones web muy robustas.

PHP también tiene la capacidad de ser ejecutado en la mayoría de los sistemas operativos tales como UNIX (y de ese tipo, como Linux), Windows y Mac OS X, y

puede interactuar con los servidores de web más populares ya que existe en versión CGI, módulo para Apache, e ISAPI.

El modelo PHP puede ser visto como una alternativa al sistema de Microsoft que utiliza ASP.NET/C#/VB.NET, a ColdFusion de la compañía Macromedia, a JSP/Java de Sun Microsystems, y al famoso CGI/Perl. Aunque su creación y desarrollo se da en el ámbito de los sistemas libres, bajo la licencia GNU, existe además un compilador comercial llamado Zend Optimizer.

### ***Historia***

PHP fue originalmente diseñado en Perl, seguidos por la escritura de un grupo de CGI binarios escritos en el lenguaje C por el programador Danés-Canadiense Rasmus Lerdorf en el año 1994 para mostrar su currículum vitae y guardar ciertos datos, como la cantidad de tráfico que su página web recibía. El 8 de junio del 1995 fue publicado "Personal Home Page Tools" después de que Lerdorf lo combinara con su propio Form Interpreter para crear PHP/FI.

Dos programadores israelíes del Technion, Zeev Suraski y Andi Gutmans, reescribieron el analizador sintáctico (parser en inglés) en el año 1997 y crearon la base del PHP 3, cambiando el nombre del lenguaje a la forma actual. Inmediatamente comenzaron experimentaciones públicas de PHP 3 y fue lanzado oficialmente en junio del 1998.

Para 1999, Suraski y Gutmans reescribieron el código de PHP, produciendo lo que hoy se conoce como Zend Engine o motor Zend. También conformaron Zend Technologies en Ratmat Gan, Israel. En mayo de 2000 PHP 4 fue lanzado bajo el poder del motor Zend Engine 1.0. El 13 de julio de 2004, fue lanzado PHP 5, utilizando el motor Zend Engine II (o Zend Engine 2). La versión más reciente de PHP es la 5.1.6, que incluye el novedoso PDO (Objetos de Datos de PHP o PHP Data Objects) y mejoras utilizando todas las ventajas que provee el nuevo Zend Engine 2.

### ***Ventajas de PHP***

- Es un lenguaje multiplataforma.
- Capacidad de conexión con la mayoría de los manejadores de base de datos que se utilizan en la actualidad.

- Leer y manipular datos desde diversas fuentes, incluyendo datos que pueden ingresar los usuarios desde formularios HTML.
- Capacidad de expandir su potencial utilizando la enorme cantidad de módulos (llamados ext's o extensiones).
- Es libre, por lo que se presenta como una alternativa de fácil acceso para todos.
- Permite las técnicas de Programación Orientada a Objetos.
- Nos permite crear los formularios para la Web

<http://es.wikipedia.org/wiki/PHP>

### **3.1.2.2 SERVIDOR WEB APACHE**

Apache es uno de los servidores Web mas potentes del mercado, ofreciendo una perfecta combinación entre estabilidad y sencillez.

Las principales características de Apache son:

- Funcionalidad en múltiples plataformas ( Unix / Linux), Windows y otras).
- Elaborado índice de directorios.
- Soporte del último protocolo http 1.1.
- Sencilla administración basada en la configuración de un único archivo.
- Soporte para CGI (Common Gateway Interface) y FastCGI

Apache presenta entre otras características mensajes de error altamente configurables, bases de datos de autenticación y negociado de contenido, pero fue criticado por la falta de una interfaz gráfica que ayude en su configuración.

En la actualidad (2006), Apache es el servidor HTTP más usado, siendo el servidor HTTP del 68% de los sitios web en el mundo y creciendo aún su cuota de mercado (estadísticas históricas y de uso diario proporcionadas por Netcraft.

[http://es.wikipedia.org/wiki/Servidor\\_web\\_Apache](http://es.wikipedia.org/wiki/Servidor_web_Apache)

### **3.1.3 CAPA DE DATOS**

#### **3.1.3.1 MySQL**

MySQL es uno de los Sistemas Gestores de bases de Datos (SQL) más populares desarrolladas bajo la filosofía de código abierto.

La desarrolla y mantiene la empresa MySQL AB pero puede utilizarse gratuitamente y su código fuente está disponible libremente.

Inicialmente, MySQL carecía de elementos considerados esenciales en las bases de datos relacionales, tales como integridad referencial y transacciones. A pesar de ello, atrajo a los desarrolladores de páginas web con contenido dinámico, justamente por su simplicidad; aquellos elementos faltantes fueron llenados por la vía de las aplicaciones que la utilizan.

Poco a poco los elementos faltantes en MySQL están siendo incorporados tanto por desarrollos internos, como por desarrolladores de software libre. Entre las características disponibles en las últimas versiones se puede destacar:

- Amplio subconjunto del lenguaje SQL.
- Algunas extensiones son incluidas igualmente.
- Disponibilidad en gran cantidad de plataformas y sistemas.
- Diferentes opciones de almacenamiento según si se desea velocidad en las operaciones o el mayor número de operaciones disponibles.
- Transacciones y claves foráneas.
- Conectividad segura.
- Replicación.
- Búsqueda e indexación de campos de texto.

<http://es.wikipedia.org/wiki/MySQL>

#### **3.1.4 MACROMEDIA FLASH MX 2004**

Flash es la plataforma de software de predominancia indiscutible, usada por más de un millón de profesionales y con una presencia en más del 97% de los equipos de escritorio con conexión a Internet en todo el mundo, así como en una amplia gama de dispositivos.

Macromedia Flash MX 2004 le permite a los diseñadores y desarrolladores integrar video, texto, audio y gráficos en experiencias dinámicas que le permiten al cliente imbuirse en su vivencia y que producen resultados superiores para marketing y presentaciones interactivas, aprendizaje electrónico e interfaces de usuario de aplicaciones. Los Timeline Effects permiten a los diseñadores añadir transiciones comunes como difusiones y sombras sin necesidad de scripting. Comportamientos (“Behaviours”) pre-definidos, eliminan la necesidad de un scripting complejo de navegación. El compilador de alto desempeño mejora significativamente el nivel de reproducción del sonido, incluso con contenido creado para versiones anteriores de Flash Player. Mejoras en Flash Player 7 entregan un desempeño adicional que ofrecen la mejor experiencia para el usuario final. Flash MX 2004 ofrece integración única a Macromedia Dreamweaver, Fireworks y las otras herramientas de diseño

Flash MX 2004 también incluye Elementos MX para Flash, componentes de aplicación que facilitan a los desarrolladores el crear rápidamente grandes experiencias con una imagen visual consistente. (*Horizonteweb.com, 2005*)

El nuevo lenguaje ActionScript 2.0 permite a los desarrolladores añadir interactividad sofisticada a su contenido. Nuevo soporte para Hojas de Estilo en Cascada –CSS- permite a los usuarios entregar sitios híbridos que mezclan Flash y HTML con una imagen visual consistente.

Los Elementos MX para Flash incluyen plantillas para proyectos comunes como presentaciones y anuncios para la Web, así como componentes útiles para la interfase de usuario como los “data grid”, un “player” para multimedia y un “accordion pane”.

<http://www.horizonteweb.com/revision/flashMX.htm>

### **3.1.5 MACROMEDIA DREAMWEAVER MX 2004**

Dreamweaver MX 2004 es la opción profesional para la creación de sitios y aplicaciones web. Proporciona una combinación potente de herramientas visuales

de disposición, características de desarrollo de aplicaciones y soporte para la edición de código. Gracias a las robustas características para la integración y diseño basado en CSS, Dreamweaver permite que los diseñadores y desarrolladores web creen y manejen cualquier sitio web con toda facilidad.

Dreamweaver MX 2004 incluye potentes controles basados en normas para asegurar un diseño de alta calidad. Un entorno de diseño construido en torno a las hojas de estilo en cascada (CSS) hace posible un desarrollo más rápido y más eficiente de sitios profesionales creados con código limpio.

Dreamweaver MX 2004 es abierto y ampliable y cuenta con los productos y las tecnologías que usted usa, para darle la libertad y la flexibilidad de elegir la tecnología que más le conviene, ahora y en el futuro. Ya que permite desarrollar sitios web de HTML, XHTML, XML, ASP, ASP.NET, JSP, PHP y Macromedia ColdFusion. Además Dreamweaver MX 2004 ofrece un FTP seguro para codificar completamente todas las transferencias de archivos y evitar el acceso no autorizado a los datos, contenido de archivos, nombres de usuario y contraseñas.  
<http://www.horizonteweb.com/revision/dreamweaverMX.htm>

### **3.1.6 MACROMEDIA FIREWORKS MX 2004**

Fireworks MX 2004 tiene las herramientas que los profesionales del web necesitan para crear de todo, desde botones gráficos sencillos hasta sofisticados efectos de rollover. Permite Importar, editar e integrar fácilmente todos los principales formatos gráficos, incluidos imágenes vectoriales y de mapas de bits. Con Fireworks MX 2004 podemos exportar fácilmente imágenes de Fireworks a Flash, Dreamweaver y aplicaciones de terceros.

Fireworks MX 2004 permite crear rápidamente gráficos web de alta calidad e interactividad compleja. Fireworks ofrece todas las herramientas familiares que usted necesita: edición de fotos robusta, control de texto preciso y creación de imágenes profesional.

Fireworks proporciona un conjunto de herramientas completo y profesional para una producción rápida y actualizaciones fáciles con lo cual se reduce el tiempo necesario para el diseño y el desarrollo de imágenes, aun al trabajar con imágenes grandes.

<http://www.horizonteweb.com/revision/fireworks.htm>

## ***CAPITULO 4***

### **4. CONCLUSIONES Y RECOMENDACIONES**

#### **4.1 CONCLUSIONES**

- El utilizar la metodología OOHDM como técnica de diseño. Ha permitido dedicar un tiempo importante en las fases previas a la implementación. Esta inversión de tiempo está ampliamente justificada no sólo porque simplifica el proceso de desarrollo, sino que también permitirá el mantenimiento de la aplicación.
- OOHDM propone un conjunto de tareas que en principio pueden involucrar mayores costos de diseño, pero que a mediano y largo plazo reducen notablemente los tiempos de desarrollo al tener como objetivo principal la reusabilidad de diseño, y así simplificar la evolución y el mantenimiento de toda aplicaron web.
- EL software de Open Source fue muy eficaz y útil en el desarrollo de este Portal WEB, no solo por el lenguaje de programación al momento de utilizar PHP, sino que el fácil manejo de creación tablas e información con Mysql, a esto se puede agregar y recalcar la confiabilidad, seguridad el este presta,
- Con el desarrollo de este Portal Web, se fortaleció conocimientos en ámbitos de Diseño, Programación, Seguridades y algo muy importante es que permitió ampliar la visión y los objetivos Profesionales.

#### **4.2 RECOMENDACIONES**

- Se recomienda que la persona encargada de la administración del sitio actualice continuamente la información de software; que cumpla las funciones de un webmaster.

- Es necesario capacitar a las personas encargadas de la administración del sitio y de notas residentes en el sistema
- Se deben instalar seguridades como fireware, antivirus, antispy, etc que protejan al sitio de intrusos y ataques. Así evitar que la información, tanto de Base de Datos y el Portal, sea alterada.

## ***BIBLIOGRAFÍA***

- Pressman Roger, Ingeniería de Software, MC. Graw-Hill, 2002.
- Pressman Roger, Ingeniería del software. Un enfoque práctico, Madrid, McGraw-Hill / Interamericana de España, 1997
- S. Murugesan, Y. Deshpande , S. Hansen, A. Ginige. “Web Engineering : A New Discipline for Development of Web - Based Systems.” Lecture Notes in Computer Science 2016 Springer, 2001.
- L. Olsina, G. Lafuente, G. Rossi. “Specifying Quality Characteristics and Attributes for Websites.” Lecture Notes in Computer Science 2016 Springer, 2001.
- Gamma, Y., Helm, R., Johnson R. y Vlissides, J. (1995). Design Patterns: Elements of reusable object-oriented software. USA: Addison Wesley.

### **Internet**

- German, D. (2003). The Object Oriented Hypermedia Design Method. [Documento en línea]. Disponible: <http://www.telemidia.puc-rio.br/oohdm/oohdm.html>.
- Sánchez, M. (s. f.). Interfaz de Usuario en el Desarrollo de un Simulador de Conducción [Documento en línea] Disponible: [http://www.cc3.net/sanchez/Sanchez\\_M.htm](http://www.cc3.net/sanchez/Sanchez_M.htm).
- Schwabe, D. y Rossi, G. (1998). Developing Hypermedia Applications using OOHDm [Documento en línea]. <http://www.oohdm.inf.puc-rio.br:8668/space/pessoas+ligadas+ao+OOHDm/ExOOHDm.pdf>.

- Schwabe, D., Rossi, G. y Simone, J. (s. f.). Systematic Hypermedia Application Design with OOHDM. [Documento en línea]. <http://wwwx.cs.unc.edu/~barman/HT96/P52/section1.html>.
- Silva, D. y Mercerat, B. (2001). Construyendo aplicaciones web con una metodología de diseño orientada a objetos. [Documento en línea]. Disponible: [www.unab.edu.co/editorialunab/revistas/rcc/pdfs/r22\\_art\\_c.pdf](http://www.unab.edu.co/editorialunab/revistas/rcc/pdfs/r22_art_c.pdf).

[http://es.wikipedia.org/wiki/Ingenier%C3%ADa\\_de\\_la\\_Web](http://es.wikipedia.org/wiki/Ingenier%C3%ADa_de_la_Web)

<http://www.informandote.com/jornadasIngWEB/articulos/jiw01.pdf>

<http://www.nosolousabilidad.com/articulos/ai.htm>

<http://www.webaprendiz.com/viewp/notas/arquiweb/2001/06/id0001.htm>

<http://www.pabloimpallari.com.ar/articulos/arquitecturaweb.php>

<http://www.infor.uva.es/~jvegas/cursos/buendia/pordocente/node18.html>

[http://es.wikipedia.org/wiki/Arquitectura\\_de\\_la\\_informaci%C3%B3n](http://es.wikipedia.org/wiki/Arquitectura_de_la_informaci%C3%B3n)

<http://www.desarrolloweb.com/articulos/importancia-documentacion.html>

<http://edutec2004.lmi.ub.es/pdf/157.pdf>

[http://www.unab.edu.co/editorialunab/revistas/rcc/pdfs/r22\\_art5\\_c.pdf](http://www.unab.edu.co/editorialunab/revistas/rcc/pdfs/r22_art5_c.pdf)

<http://agamenon.uniandes.edu.co/~pfigueroa/soo/uml>

<http://www.geocities.com/txmetsb/#Que es un Use Case>

<http://www.clikear.com/manuales/uml/diagramasestructuraestatica.asp>

<http://www.dcc.uchile.cl/~psalinas/uml/modelo.html>

<http://www.dcc.uchile.cl/~psalinas/uml/interaccion.html>

<http://www.creangel.com/uml/secuencia.php#>

<http://www.creangel.com/uml/colaboracion.php#>

<http://es.tldp.org/Tutoriales/doc-modelado-sistemas-UML/multiple-html/x98.html>

<http://es.tldp.org/Tutoriales/doc-modelado-sistemas-UML/multiple-html/x277.html>

[http://www.embarcadero.com/support/what\\_is\\_uml.asp](http://www.embarcadero.com/support/what_is_uml.asp)

<http://www.monografias.com/trabajos5/insof/insof.shtml>

<http://lucas.hispalinux.es/Tutoriales/doc-modelado-sistemas-UML/multiple-html/index.html>

<http://es.wikipedia.org/wiki/JavaScript>

[http://es.wikipedia.org/wiki/HTML#Historia\\_del\\_est.C3.A1ndar](http://es.wikipedia.org/wiki/HTML#Historia_del_est.C3.A1ndar)

<http://es.wikipedia.org/wiki/PHP>

[http://es.wikipedia.org/wiki/Servidor\\_web\\_Apache](http://es.wikipedia.org/wiki/Servidor_web_Apache)

<http://es.wikipedia.org/wiki/MySQL>

<http://www.horizonteweb.com/revision/flashMX.htm>

<http://www.horizonteweb.com/revision/dreamweaverMX.htm>

<http://www.horizonteweb.com/revision/fireworks.htm>

# ***ANEXOS***

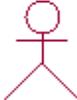
# MANUAL TÉCNICO

ANALISIS Y REQUERIMIENTOS.....	74
MODELO ESTÁTICO	75
1.1 IDENTIFICACION DE ACTORES .....	76
1.2 DIAGRAMAS DE CASOS DE USO .....	77
1.2.1 DIAGRAMAS DE CASOS DE USO POR ACTIVIDADES.....	77
1.2.2 DIAGRAMAS DE CASOS DE USO POR ACTORES .....	81
1.2.3 DICCIONARIO DE CASOS DE USO .....	85
1.3 DIAGRAMA DE CLASES.....	97
1.4 DIAGRAMA DE OBJETOS.....	98
MODELO DINÁMICO	101
2.1 DIAGRAMAS DE INTERACCIÓN.....	102
2.1.1 DIAGRAMAS DE SECUENCIA .....	102
2.1.2 DIAGRAMAS DE COLABORACIÓN.....	106
2.1.3 DIAGRAMAS DE ESTADOS.....	111
2.1.4 DIAGRAMAS DE ACTIVIDADES.....	113
MODELO ARQUITECTÓNICO	119
MODELO NAVEGACIONAL	121

# **ANALISIS Y REQUERIMIENTOS**

# **MODELO ESTÁTICO**

## 1.1 IDENTIFICACION DE ACTORES

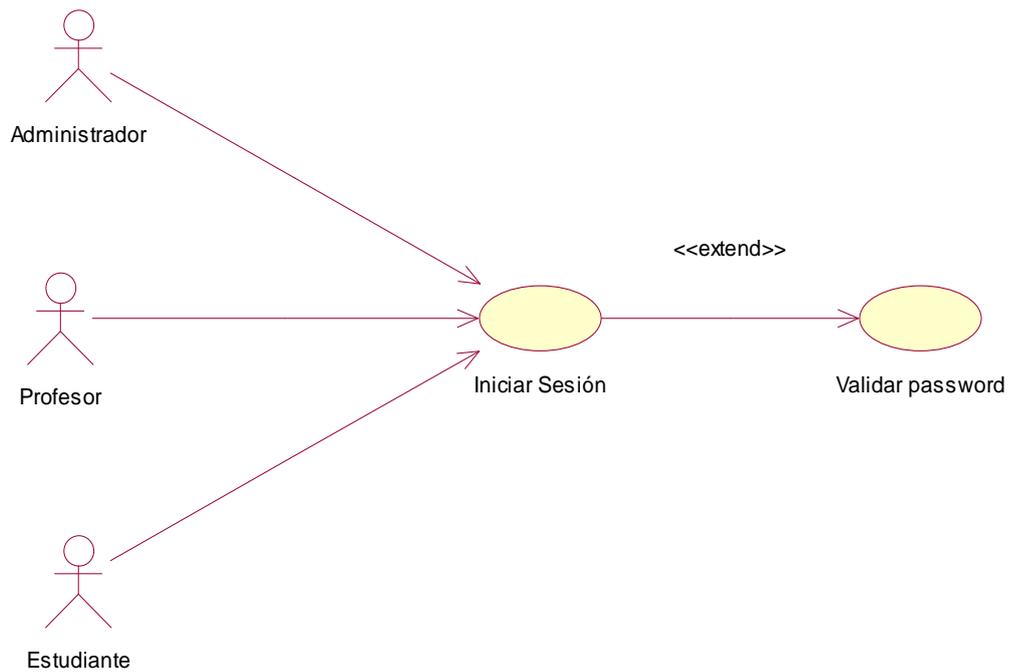
ACTORES	FUNCIÓN
 <p data-bbox="333 822 489 846">Administrador</p>	<ul data-bbox="635 506 1369 1323" style="list-style-type: none"><li>• Iniciar sesión en el sistema.</li><li>• Validar password.</li><li>• Ingresar/Actualizar/Borrar datos personales del estudiante.</li><li>• Ingresar/Actualizar/Borrar datos personales del profesor.</li><li>• Ingresar/Actualizar/Borrar datos de especialidad.</li><li>• Ingresar/Actualizar/Borrar datos de materia.</li><li>• Ingresar/Actualizar/Borrar datos de curso.</li><li>• Ingresar/Actualizar calificaciones.</li><li>• Consultar/Imprimir calificaciones por alumno.</li><li>• Consultar/Imprimir calificaciones por curso y materia.</li><li>• Consultar /Imprimir lista de profesores.</li><li>• Consultar/Imprimir lista de alumnos por curso.</li></ul>
 <p data-bbox="365 1523 458 1547">Profesor</p>	<ul data-bbox="635 1404 1082 1547" style="list-style-type: none"><li>• Iniciar sesión en el sistema.</li><li>• Cambiar contraseña profesor.</li><li>• Ingresar calificaciones.</li></ul>
 <p data-bbox="349 1803 467 1827">Estudiante</p>	<ul data-bbox="635 1693 1098 1834" style="list-style-type: none"><li>• Iniciar sesión en el sistema.</li><li>• Cambiar contraseña estudiante.</li><li>• Consultar calificaciones.</li></ul>

 Internauta	<ul style="list-style-type: none"><li>• Registrar internauta.</li><li>• Ver información.</li></ul>
---	--

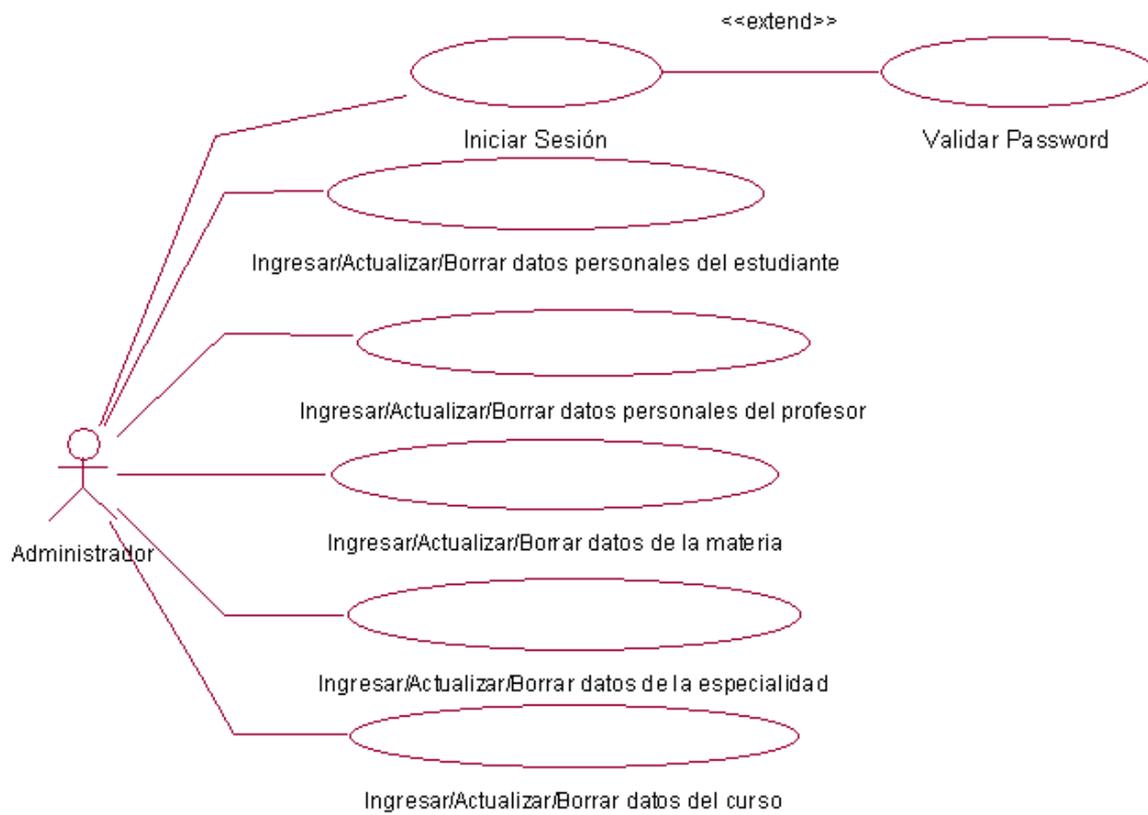
## 1.2 DIAGRAMAS DE CASOS DE USO

### 1.2.1 DIAGRAMAS DE CASOS DE USO POR ACTIVIDADES

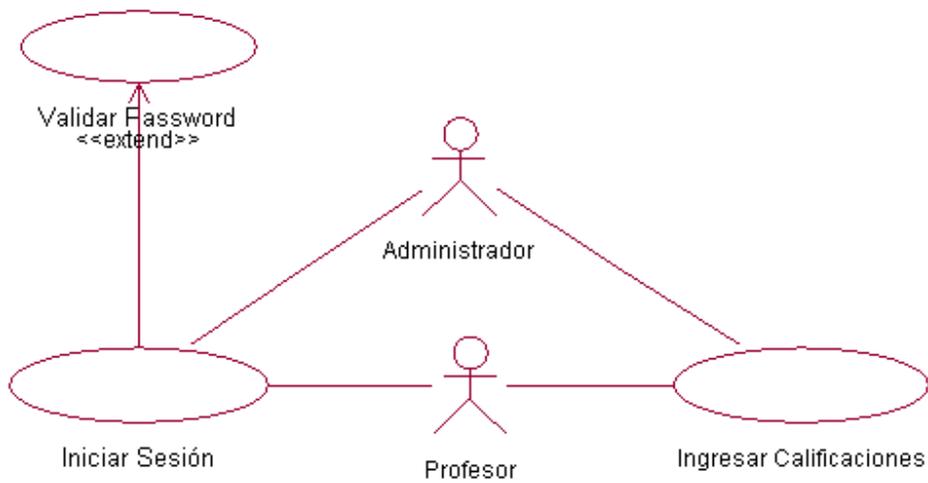
#### 1. INICIO DE SESIÓN



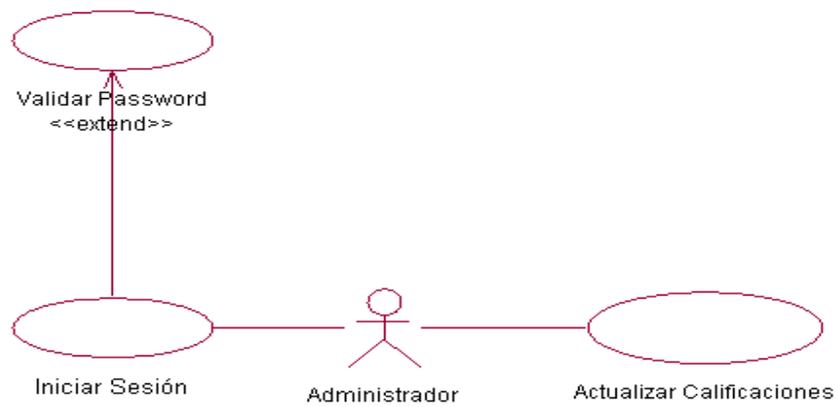
## 2. INGRESAR/ACTUALIZAR/BORRAR DATOS



## 3. INGRESAR CALIFICACIONES



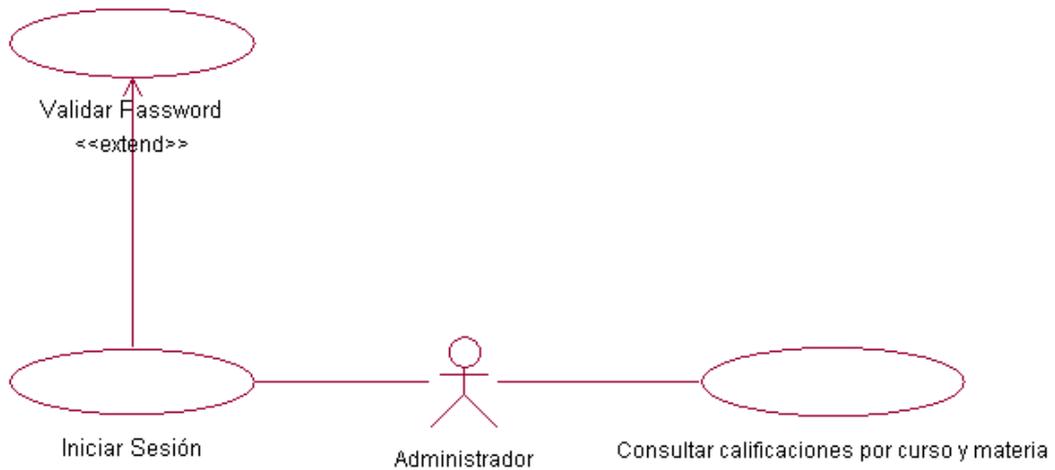
#### 4. ACTUALIZAR CALIFICACIONES



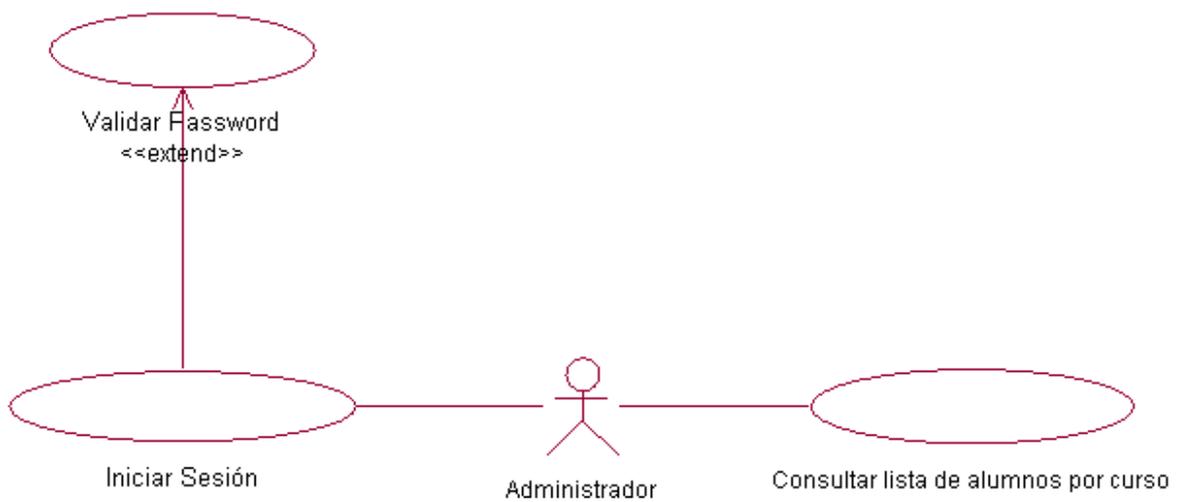
#### 5. CONSULTAR CALIFICACIONES POR ALUMNO



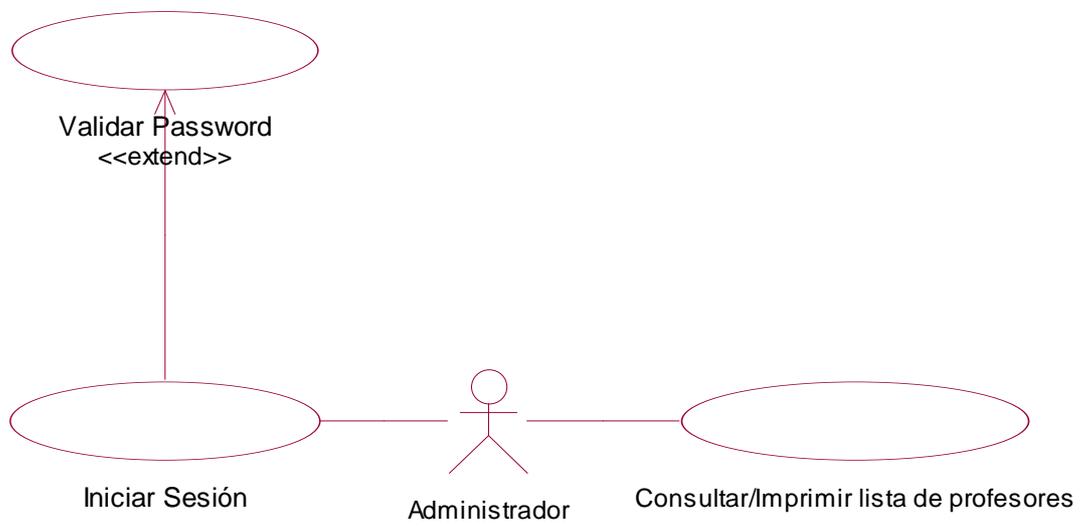
## 6. CONSULTAR CALIFICACIONES POR CURSO Y MATERIA



## 7. CONSULTAR/IMPRIMIR LISTA DE ALUMNOS POR CURSO

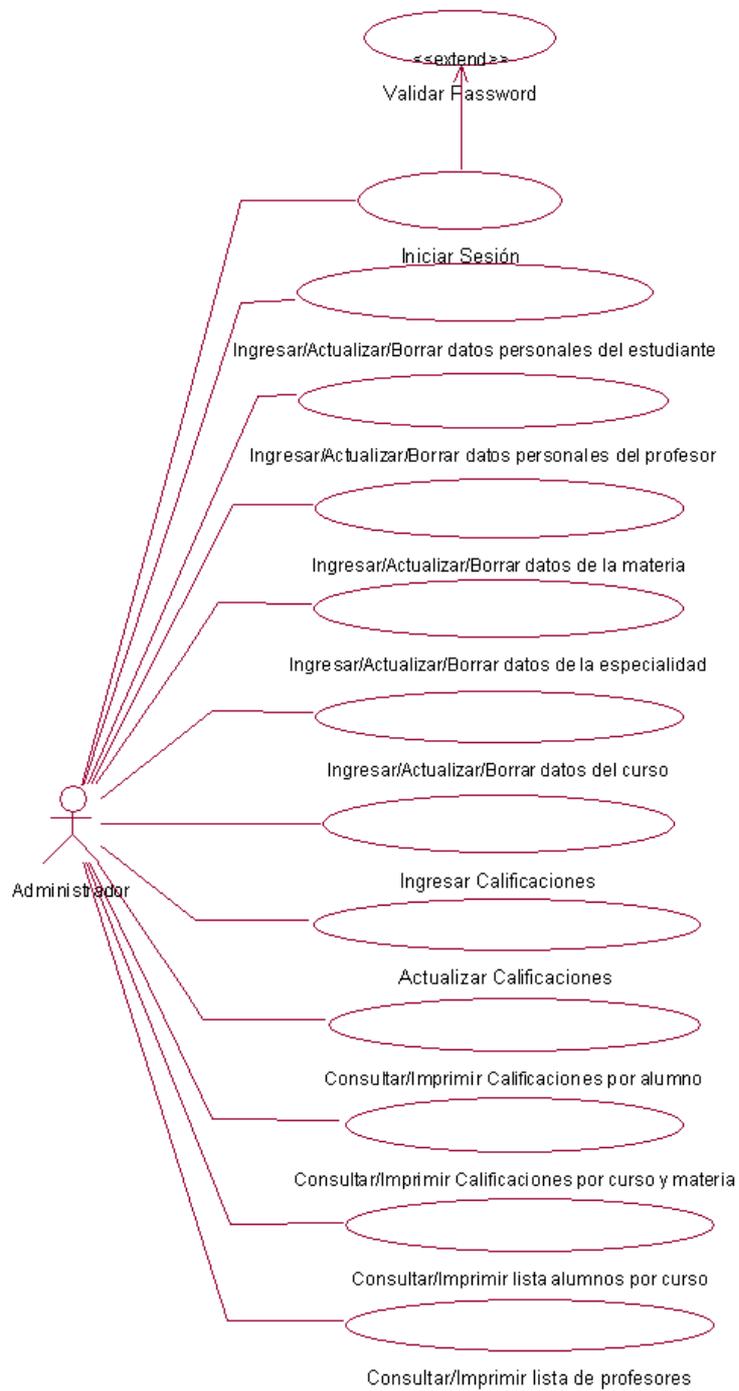


## 8. CONSULTAR/IMPRIMIR LISTA DE PROFESORES

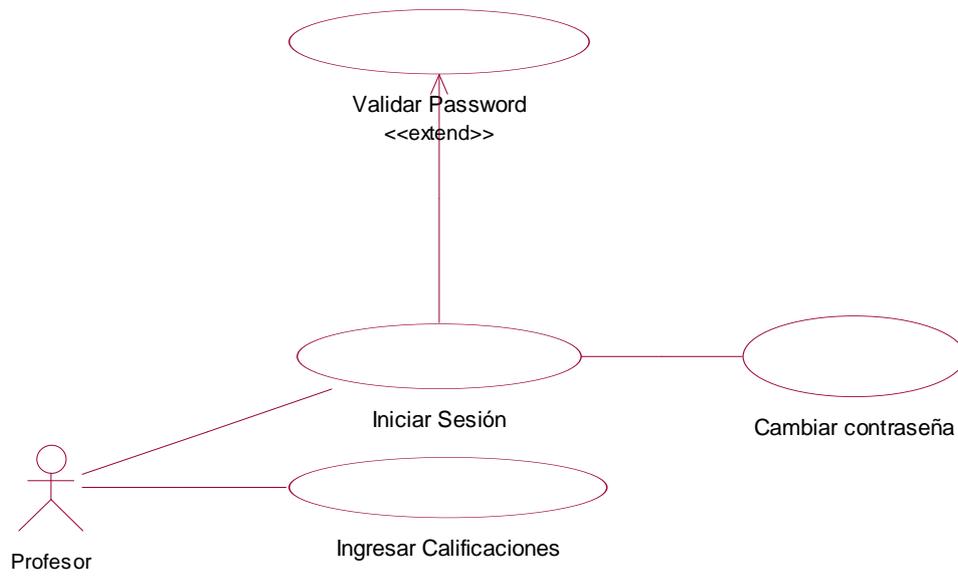


## 1.2.2 DIAGRAMAS DE CASOS DE USO POR ACTORES

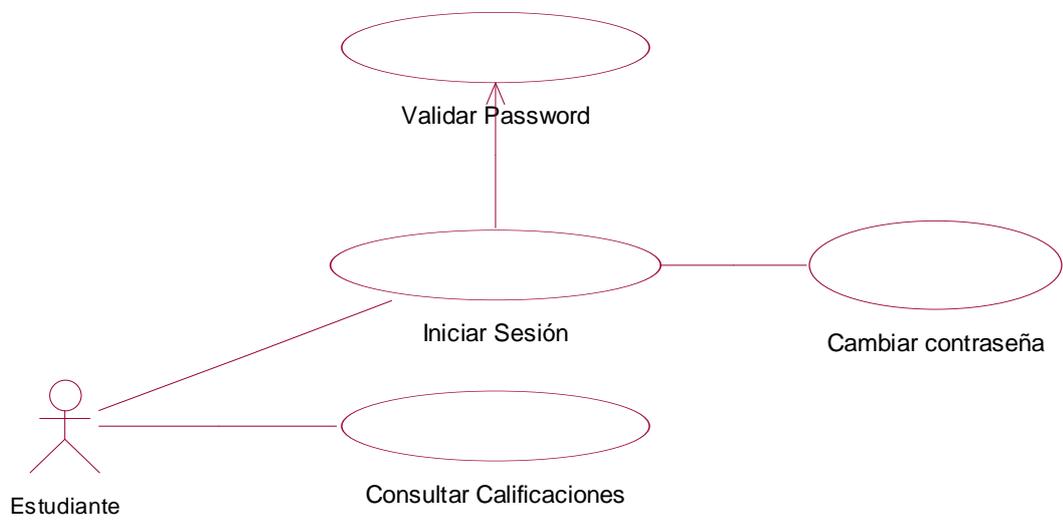
### 1. ADMINISTRADOR



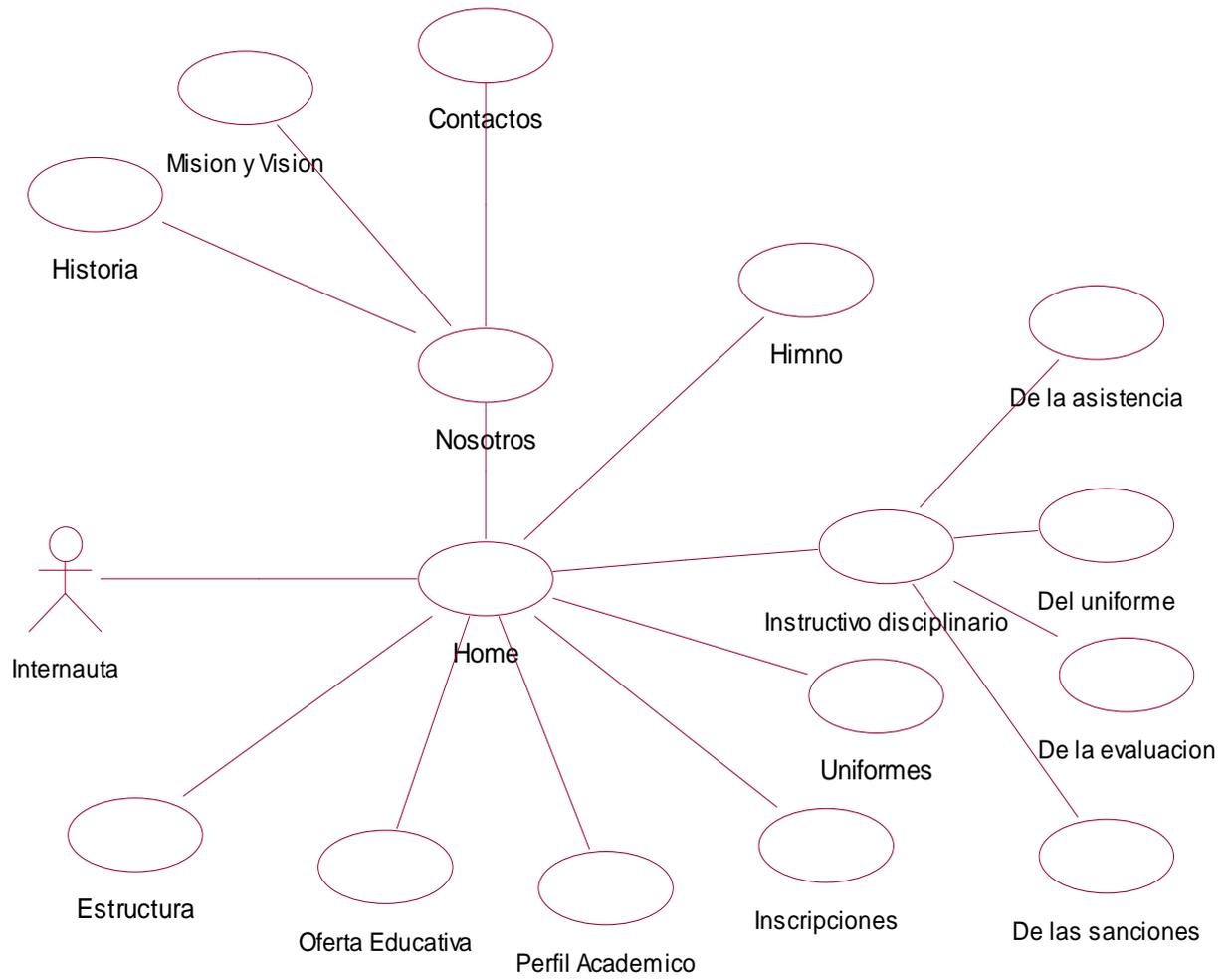
## 2. PROFESOR



### 3. ESTUDIANTE



## 4. INTERNAUTA



## 1.2.3 DICCIONARIO DE CASOS DE USO

### ADMINISTRADOR

#### Inicia sesión en el sistema.

ACCION	REACCION
Ingresar el nombre de usuario del sistema y contraseña.	Llamar a la función que valida usuario y contraseña. Si la respuesta es positiva inicia la sesión para el usuario dado y crea las variables de sesión: año lectivo en curso, grupo del usuario, código de estudiante/profesor asociado con el usuario de sistema. En caso de recibir un usuario o contraseña inválidos despliega un mensaje de error.

#### Validar contraseña.

ACCION	REACCION
Se llama a la función de validación que recibe como parámetros un código de usuario y una contraseña	Valida al usuario y contraseña ingresados buscando en la base de datos. Retorna verdadero o falso.

**Ingresar/ Actualizar/Borrar datos personales del estudiante.**

<b>ACCION</b>	<b>REACCION</b>
El usuario estudiante realiza la petición de ingreso	El sistema despliega un formulario.
El usuario estudiante llena el formulario y pide guardar.	Los datos son validados por el sistema en la tabla del estudiante y despliega los cursos para escoger en cual se va a inscribir, caso contrario indica error.
El usuario estudiante escoge el curso, paralelo, especialidad y solicita guardar.	El sistema guarda los datos y envía mensaje de datos guardados y el user del estudiante, a continuación ya se muestra la lista de los estudiantes donde ya consta el recién ingresado.
El administrador llena o edita los datos entregados por un usuario estudiante del sistema y ordena guardar.	La información se guarda en la base de datos, en la tabla correspondiente y se crea un usuario del sistema. Se despliega un mensaje indicando si la operación fue exitosa o de lo contrario un mensaje de error.

<b>ACCION</b>	<b>REACCION</b>
El administrador ingresa al usuario estudiante.	El sistema despliega los nombres de las estudiantes que se encuentran como usuarios.
El administrador escoge un usuario estudiante y ordena abrir.	El sistema devuelve un formulario parcialmente lleno con los datos del estudiante.
El administrador con la información del estudiante previamente enviada; escoge la curso, paralelo, especialidad y ordena guardar.	El sistema guarda los datos envía mensaje de datos guardados y la contraseña del estudiante, a continuación ya se muestra la lista de los estudiantes donde ya consta el recién ingresado.
<b>ACCIÓN</b>	<b>REACCIÓN</b>
El administrador escoge el estudiante para realizar la actualización de los datos	Se despliega los datos del estudiante.
El administrador actualiza los datos del estudiante y pide guardar caso contrario pide cancelar.	Se muestra un mensaje pidiendo confirmación.
El administrador ordena aceptar	Se muestra un mensaje de que los datos han sido actualizados.
<b>ACCIÓN</b>	<b>REACCIÓN</b>

El administrador escoge el estudiante y pide borrar.	Se muestra un mensaje pidiendo confirmación.
El administrador ordena aceptar.	Los datos del estudiante son borrados.

### **Ingresar/Actualizar/Borrar datos personales del profesor.**

<b>ACCIÓN</b>	<b>REACCIÓN</b>
El usuario profesor realiza la petición de ingreso.	El sistema despliega un formulario.
El usuario profesor llena el formulario y pide guardar.	Se validan los datos, para luego ser guardados en el sistema. Se envía un mensaje de datos guardados y el user del profesor a continuación ya se muestran la lista de los profesor donde ya consta el recién ingresado, caso contrario muestra mensaje de error.
<b>ACCIÓN</b>	<b>REACCIÓN</b>
El administrador escoge el profesor para realizar la actualización respectiva.	Se despliega los datos del profesor.
El administrador actualiza los datos del profesor y pide guardar caso contrario pide cancelar.	Se muestra un mensaje pidiendo confirmación.

El administrador ordena aceptar.	Se muestra un mensaje de que los datos han sido actualizados.
<b>ACCIÓN</b>	<b>REACCIÓN</b>
El administrador selecciona el profesor que desea borrar.	Si el profesor existe y esta dictando materias se muestra un mensaje de que el profesor no se puede borrar, caso contrario muestra un mensaje pidiendo confirmación.
El administrador ordena aceptar.	El profesor es borrado de la base

#### **Ingresar/Actualizar/Borrar datos de especialidad.**

<b>ACCIÓN</b>	<b>REACCIÓN</b>
El administrador requiere ingresar nueva especialidad.	El sistema despliega un formulario.
El administrador llena el formulario y pide guardar.	Se validan los datos y a continuación el sistema envía mensaje de datos guardados caso contrario sale mensaje de error.
<b>ACCIÓN</b>	<b>REACCIÓN</b>
El administrador escoge la especialidad que se va a actualizar.	Se despliega los datos de la especialidad.
El administrador actualiza los datos de la especialidad y pide guardar caso contrario pide cancelar.	Se muestra un mensaje pidiendo confirmación.

El administrador ordena aceptar.	Se muestra un mensaje de que los datos han sido actualizados.
<b>ACCIÓN</b>	<b>REACCIÓN</b>
El administrador escoge la especialidad y pide borrar.	Se muestra un mensaje pidiendo confirmación.
El administrador ordena aceptar.	La especialidad y sus datos son borrados de la base.

**Ingresar/Actualizar/Borrar datos de materia.**

<b>ACCIÓN</b>	<b>REACCIÓN</b>
El administrador requiere ingresar nueva materia.	El sistema despliega un formulario.
El administrador llena el formulario y pide guardar.	Los datos son validados por el sistema y a continuación el sistema envía mensaje de datos guardados caso contrario sale mensaje de error.
<b>ACCIÓN</b>	<b>REACCIÓN</b>
El administrador selecciona la especialidad que va a actualizar.	Se despliega los datos de la especialidad.
El administrador actualiza los datos de la materia y pide guardar caso contrario	Se muestra un mensaje pidiendo confirmación.

pide cancelar	
El administrador ordena aceptar.	Se muestra un mensaje de que los datos han sido actualizados.
<b>ACCIÓN</b>	<b>REACCIÓN</b>
El administrador selecciona la materia y pide borrar.	Se muestra un mensaje pidiendo confirmación.
El administrador ordena aceptar.	La materia es borrada de la base.

### **Ingresar/Actualizar/Borrar datos de curso.**

<b>ACCIÓN</b>	<b>REACCIÓN</b>
El administrador requiere ingresar curso.	El sistema despliega un formulario.
El administrador llena el formulario y pide guardar.	Se validan los datos y a continuación el sistema envía mensaje de datos guardados caso contrario sale mensaje de error.
<b>ACCIÓN</b>	<b>REACCIÓN</b>
El administrador actualiza los datos del curso y pide guardar caso contrario pide cancelar.	Se muestra un mensaje pidiendo confirmación.

El administrador ordena aceptar.	Se muestra un mensaje de que los datos han sido actualizados.
<b>ACCIÓN</b>	<b>REACCIÓN</b>
El administrador selecciona el curso y pide borrar.	Se muestra un mensaje pidiendo confirmación.
El administrador ordena aceptar.	El curso y los datos son borrados de la base.

### **Ingresar/Actualizar calificaciones**

<b>ACCIÓN</b>	<b>REACCIÓN</b>
El administrador ingresa los datos de calificaciones y ordena guardar.	Los datos ingresados se guardan en la base de datos en la tabla correspondiente. Se despliega un mensaje para el usuario indicando una operación exitosa o un error.
El administrador selecciona los datos de año lectivo, especialidad, curso,	Se despliega la lista de los estudiantes con sus calificaciones de acuerdo con

materia y ordena guardar	los parámetros.
<b>ACCIÓN</b>	<b>REACCIÓN</b>
El administrador escoge el estudiante del cual se van a actualizar sus calificaciones	Se despliegan las calificaciones del estudiante con opción a ser actualizadas
El administrador actualiza las calificaciones y ordena cambiar.	Se muestra un mensaje de confirmación.
El administrador acepta.	Se presenta mensaje de calificación actualizada.

### **Consultar/Imprimir calificaciones por alumno.**

<b>ACCIÓN</b>	<b>REACCIÓN</b>
El administrador pide generar calificaciones por estudiante	Se muestra una pantalla donde se escoge el estudiante.
El administrador selecciona el estudiante del cual se van a consultar las calificaciones.	Se realiza una consulta sobre la base de datos con los parámetros ingresados y se despliegan las calificaciones del estudiante.

<b>ACCIÓN</b>	<b>REACCIÓN</b>
El administrador ordena imprimir.	El sistema genera un reporte del estudiante con todas sus calificaciones.

**Consultar/Imprimir calificaciones por curso y materia.**

El administrador pide generar nómina de estudiantes por materia.	Se despliega una pantalla donde se muestra tipo de curso, nivel, materia, año lectivo.
El administrador escoge la materia y ordena generar.	Si existen datos se muestra la nómina de estudiantes, caso contrario se muestra un mensaje de que no existen estudiantes en esa materia.
<b>ACCIÓN</b>	<b>REACCIÓN</b>
El administrador ordena imprimir.	El sistema genera un reporte de los estudiantes.

**Consultar /Imprimir lista de profesores.**

<b>ACCIÓN</b>	<b>REACCIÓN</b>
El administrador pide generar nómina de profesores	Se realiza una consulta sobre la base de datos con los parámetros ingresados y se despliegan los datos de los profesores, teléfonos y materias que dictan en el presente año lectivo.

<b>ACCIÓN</b>	<b>REACCIÓN</b>
El administrador ordena imprimir.	El sistema genera una lista de los profesores.

**Consultar/Imprimir lista de alumnos por curso.**

<b>ACCIÓN</b>	<b>REACCIÓN</b>
El administrador selecciona año lectivo, curso y especialidad. Luego ordena consultar	Se realiza una consulta sobre la base de datos con los parámetros ingresados y se despliega una lista con los nombres de los estudiantes.
El administrador ordena imprimir.	El sistema genera una lista de los estudiantes del curso seleccionado.

**PROFESOR**

**Iniciar sesión en el sistema.**

<b>ACCIÓN</b>	<b>REACCIÓN</b>
El profesor ingresa login y password.	El sistema valida estas variables y si son correctas inicia sesión. Se muestra la pantalla con las materias que dicta.

**Cambiar contraseña profesor.**

<b>ACCIÓN</b>	<b>REACCIÓN</b>
El profesor requiere cambiar contraseña.	El sistema presenta una ventana que pide contraseña anterior y nueva.
El profesor ingresa contraseñas y	Si las contraseñas son válidas se

solicita cambiar.	cambia, caso contrario presenta mensaje de error.
-------------------	---

### Ingresar calificaciones.

ACCIÓN	REACCIÓN
El profesor escoge la materia.	Si las calificaciones ya fueron ingresadas no permiten hacer cambios, caso contrario se despliegan los estudiantes con opción a ingresar sus notas.
El profesor ingresa notas y pide guardar.	Si las calificaciones son válidas se guardan, caso contrario emite mensaje de error.

### ESTUDIANTE

#### Iniciar sesión en el sistema.

ACCIÓN	REACCIÓN
El estudiante ingresa login y password.	El sistema valida estas variables y si son correctas inicia sesión.

#### Cambiar contraseña estudiante.

ACCIÓN	REACCIÓN
El estudiante requiere cambiar contraseña.	El sistema presenta una ventana que pide contraseña anterior y nueva.
El estudiante ingresa contraseñas y	Si las contraseñas son válidas se

solicita cambiar.	cambia, caso contrario presenta mensaje de error.
-------------------	---

### Consultar calificaciones.

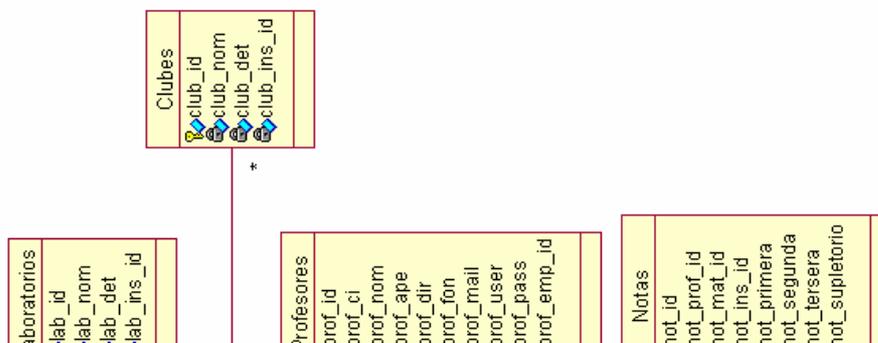
ACCIÓN	REACCIÓN
El usuario requiere ver calificaciones.	El sistema despliega las materias que toma el estudiante con sus calificaciones.

## INTERNAUTA

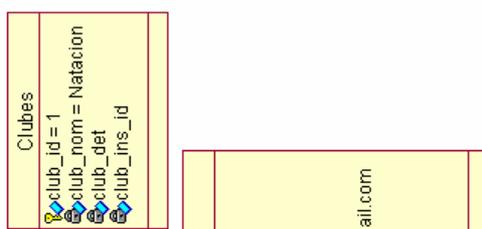
### Registrar internauta.

ACCIÓN	REACCIÓN
El internauta pide registrarse.	La página despliega un formulario de registro.
El internauta llena formulario y pide enviar.	Si los datos son válidos se envían, caso contrario muestra mensaje de error.

## 1.3 DIAGRAMA DE CLASES



## 1.4 DIAGRAMA DE OBJETOS





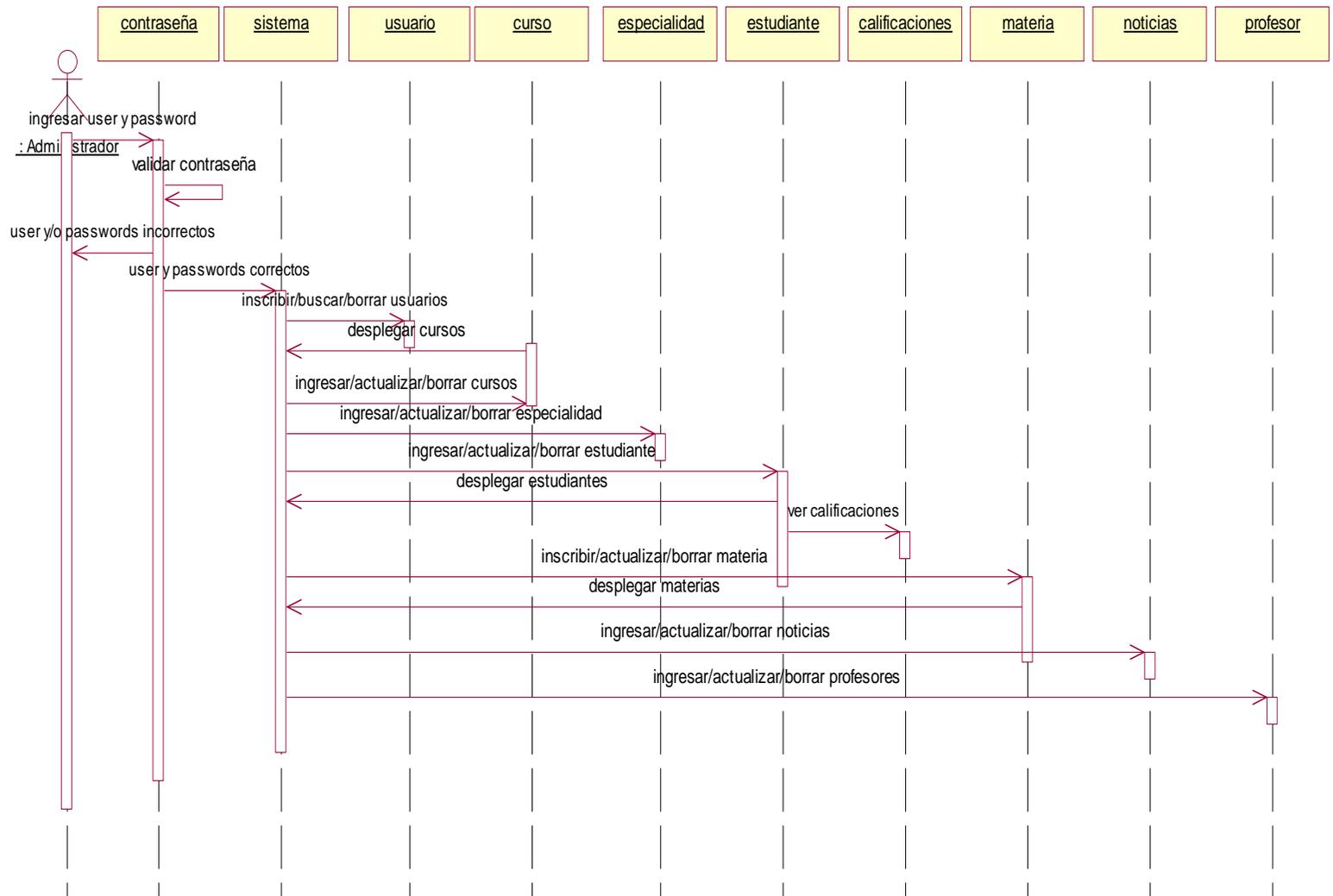
**DISEÑO**

# **MODELO DINÁMICO**

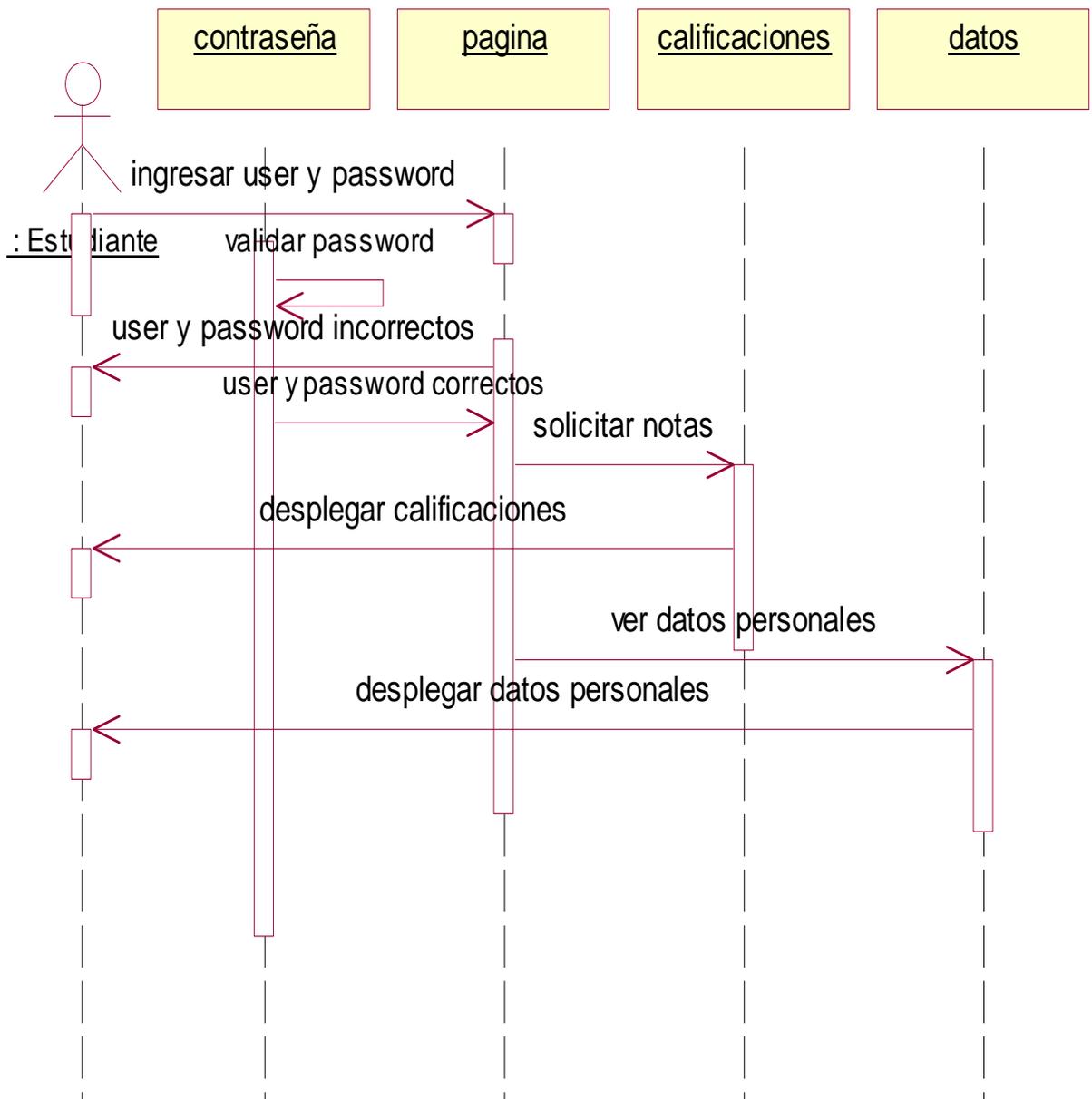
## 2.1 DIAGRAMAS DE INTERACCIÓN

### 2.1.1 DIAGRAMAS DE SECUENCIA

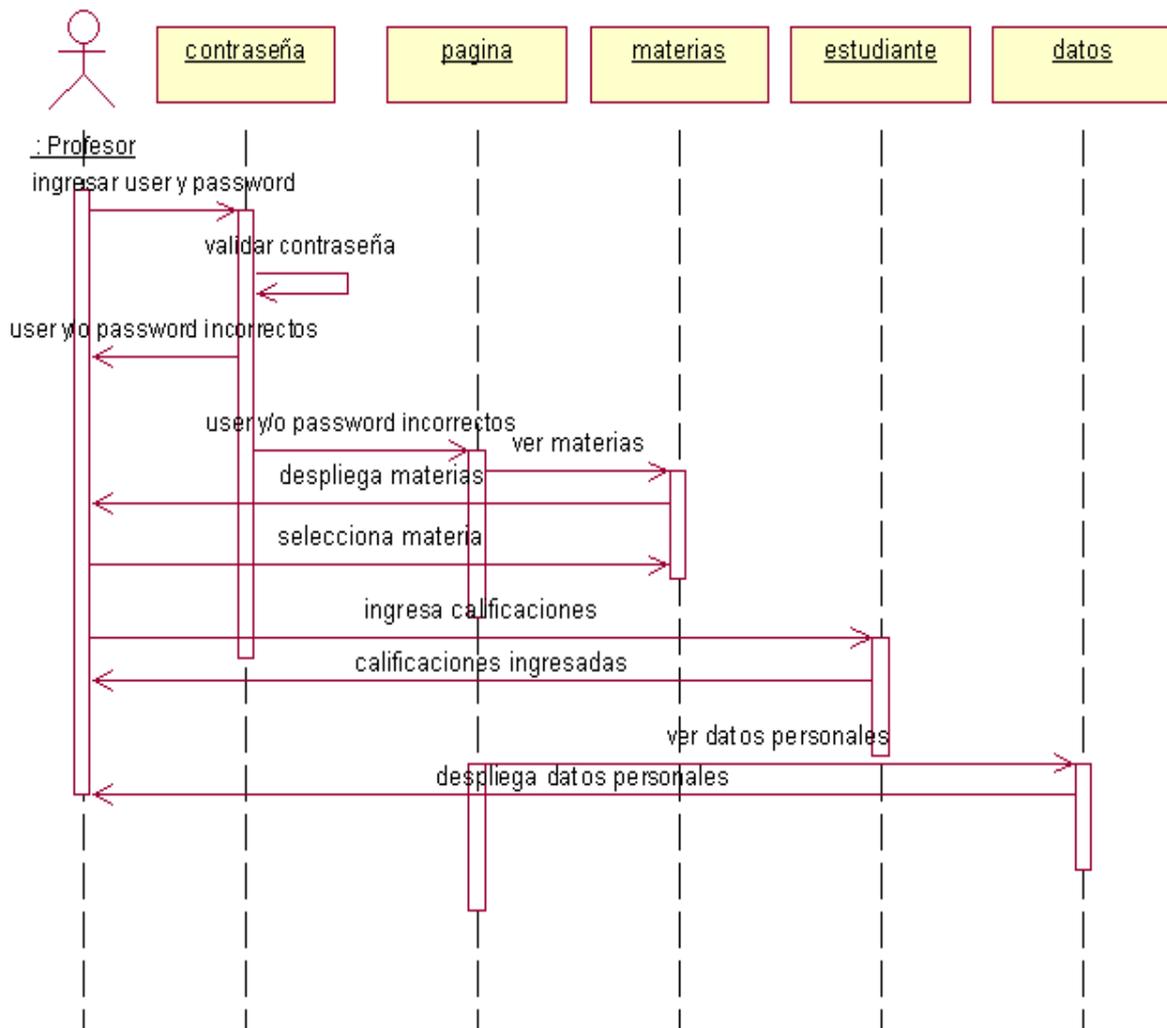
#### ADMINISTRADOR



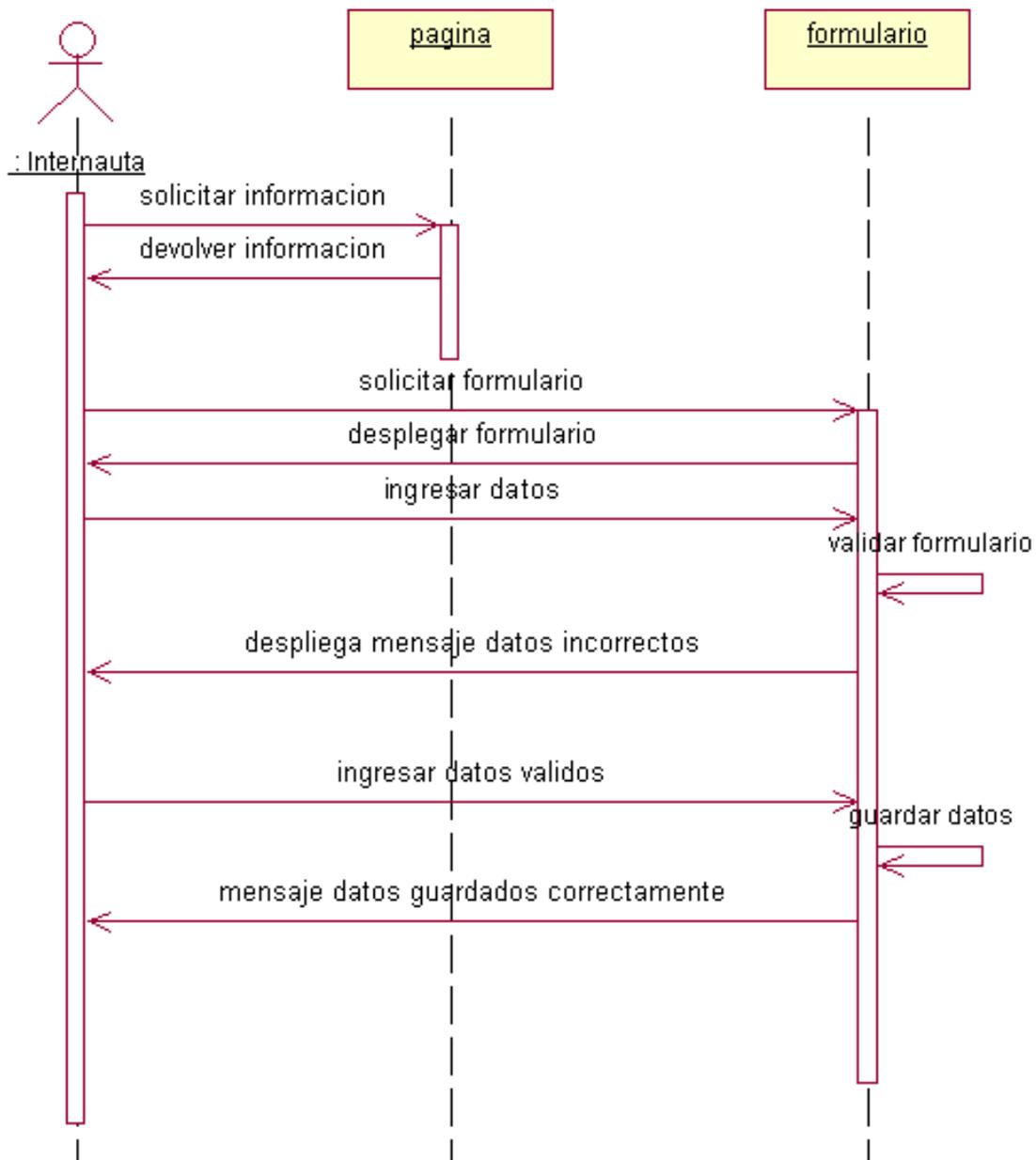
## ESTUDIANTE



## PROFESOR

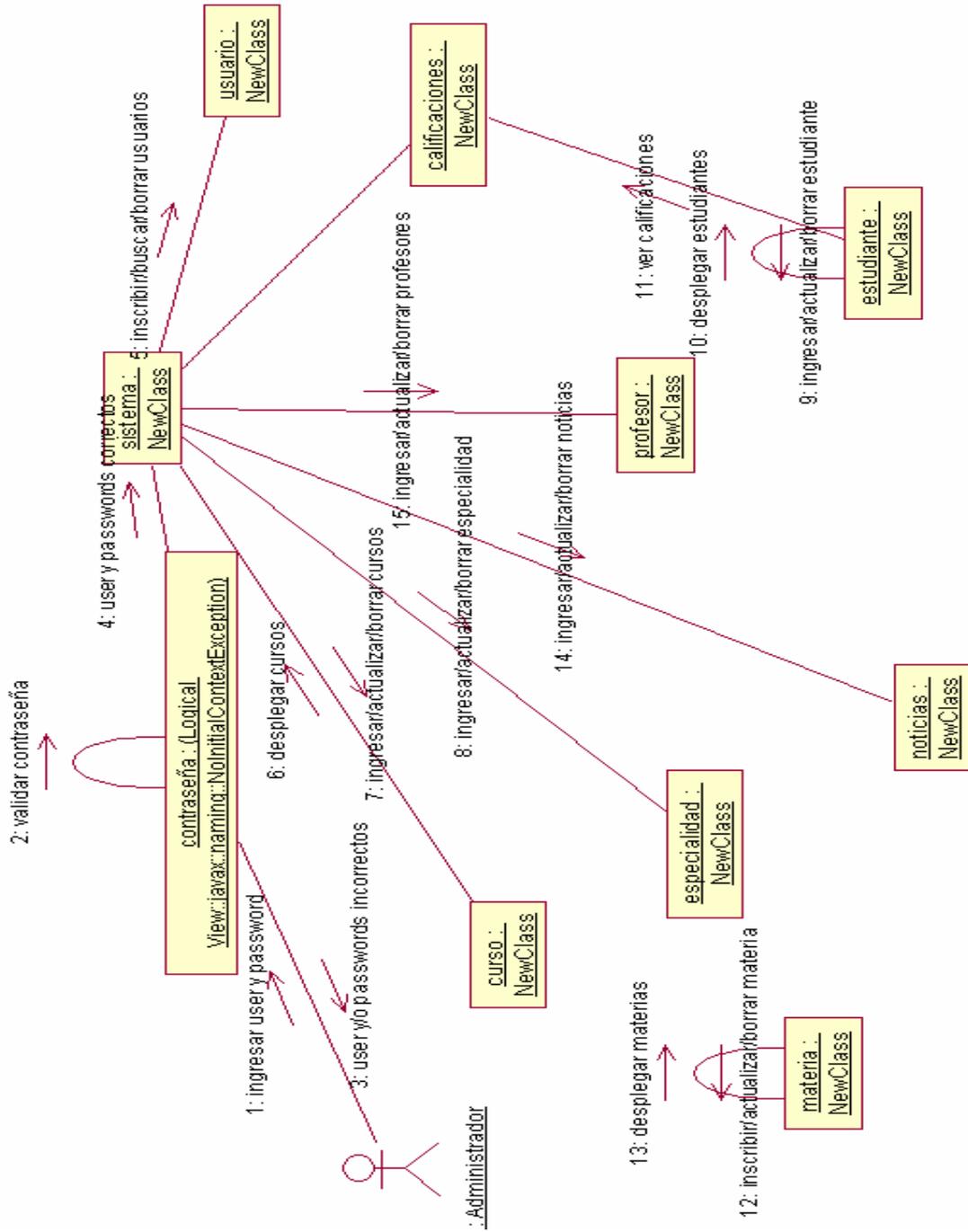


**INTERNAUTA**

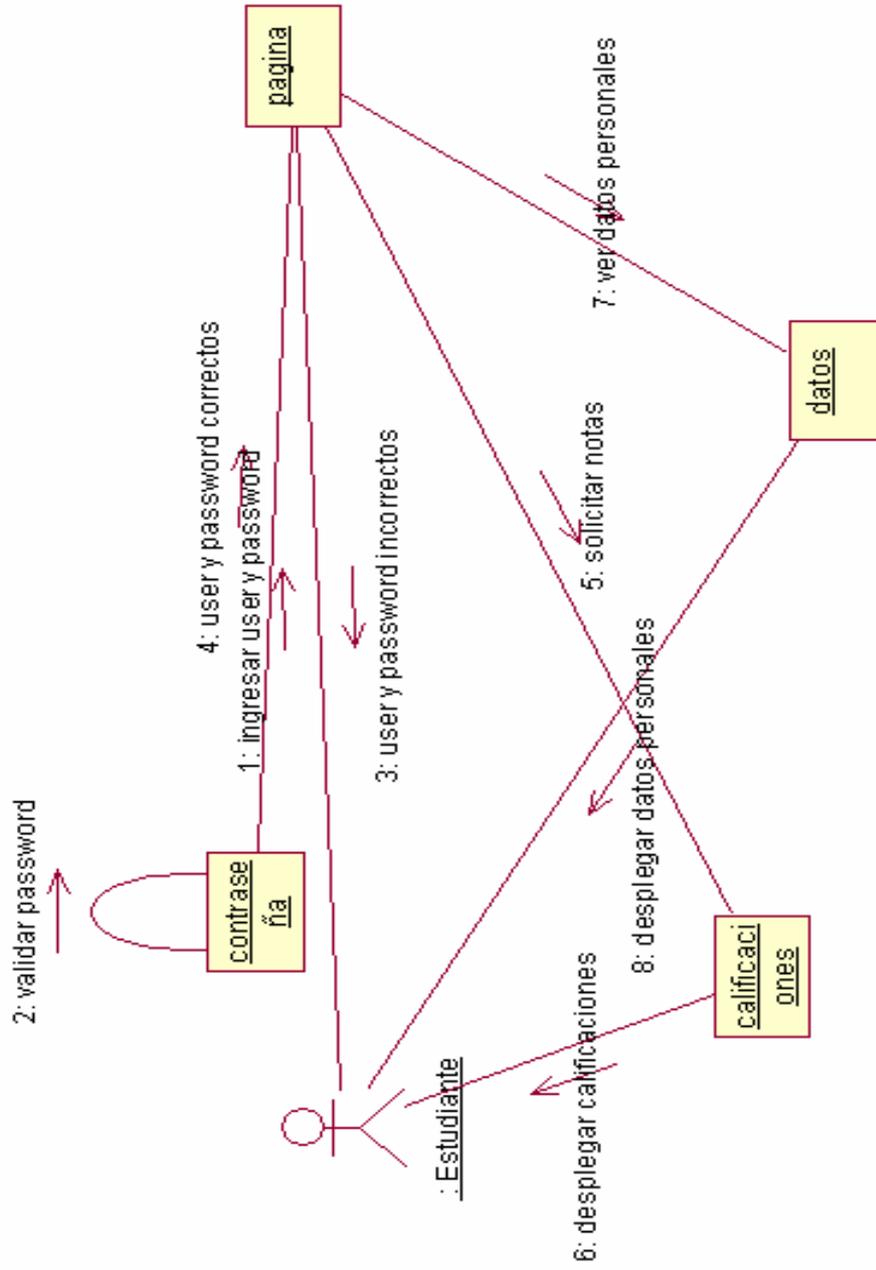


## 2.1.2 DIAGRAMAS DE COLABORACIÓN

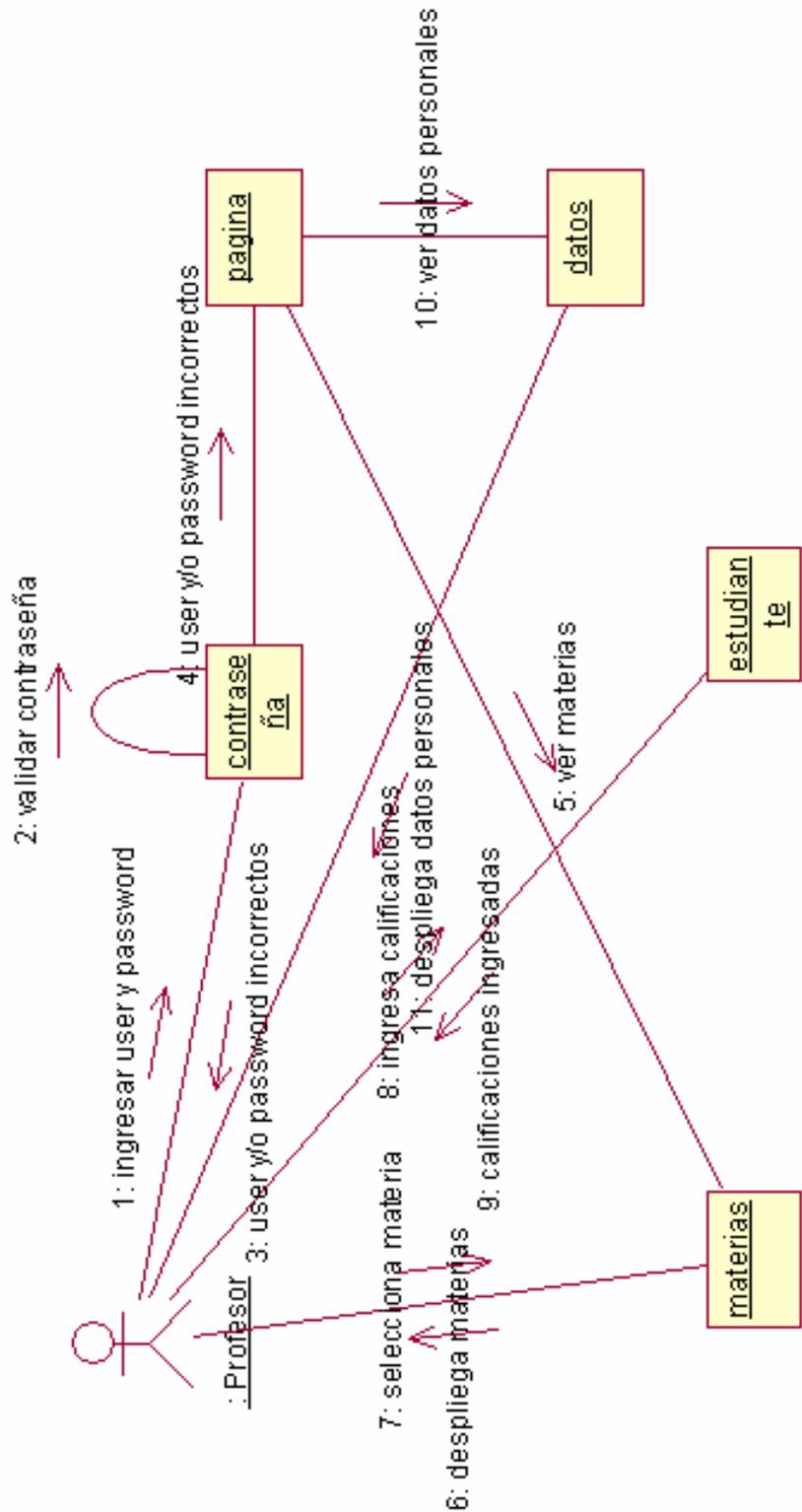
# ADMINISTRADOR



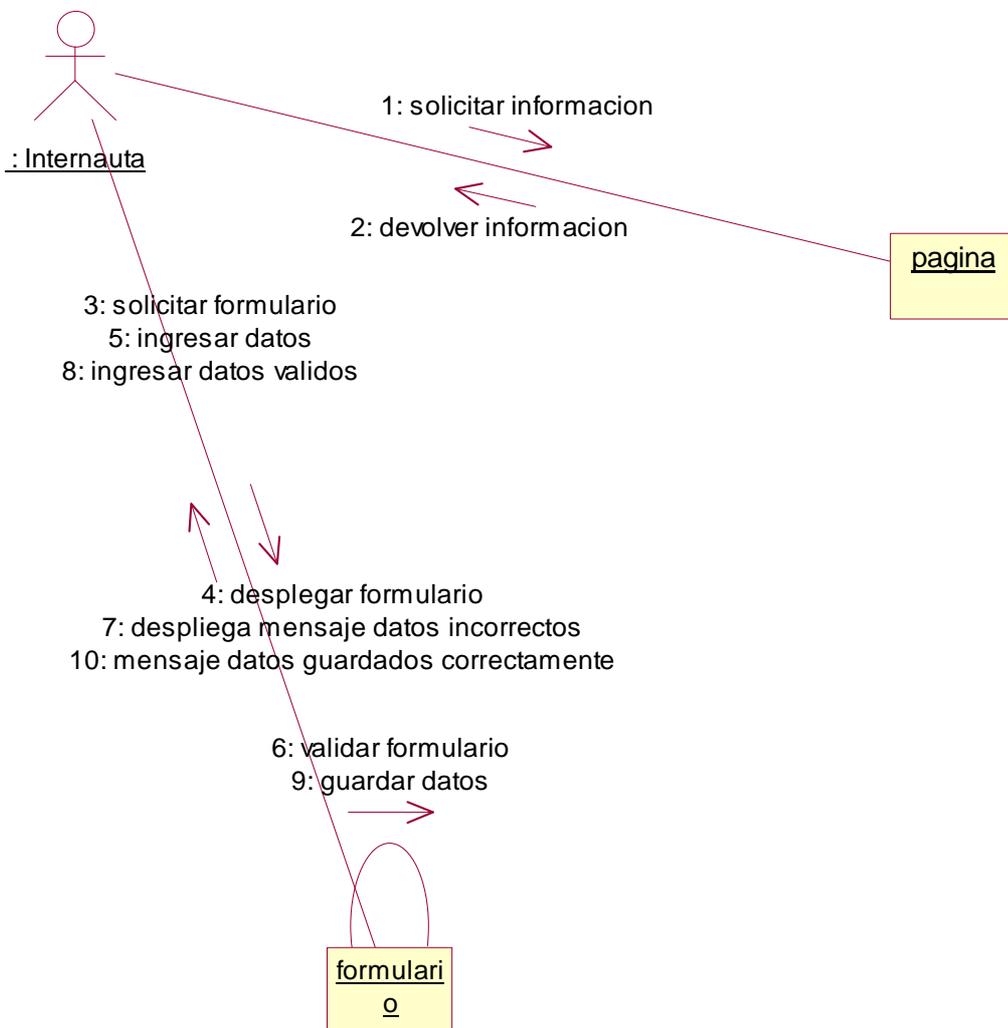
# ESTUDIANTE



# PROFESOR

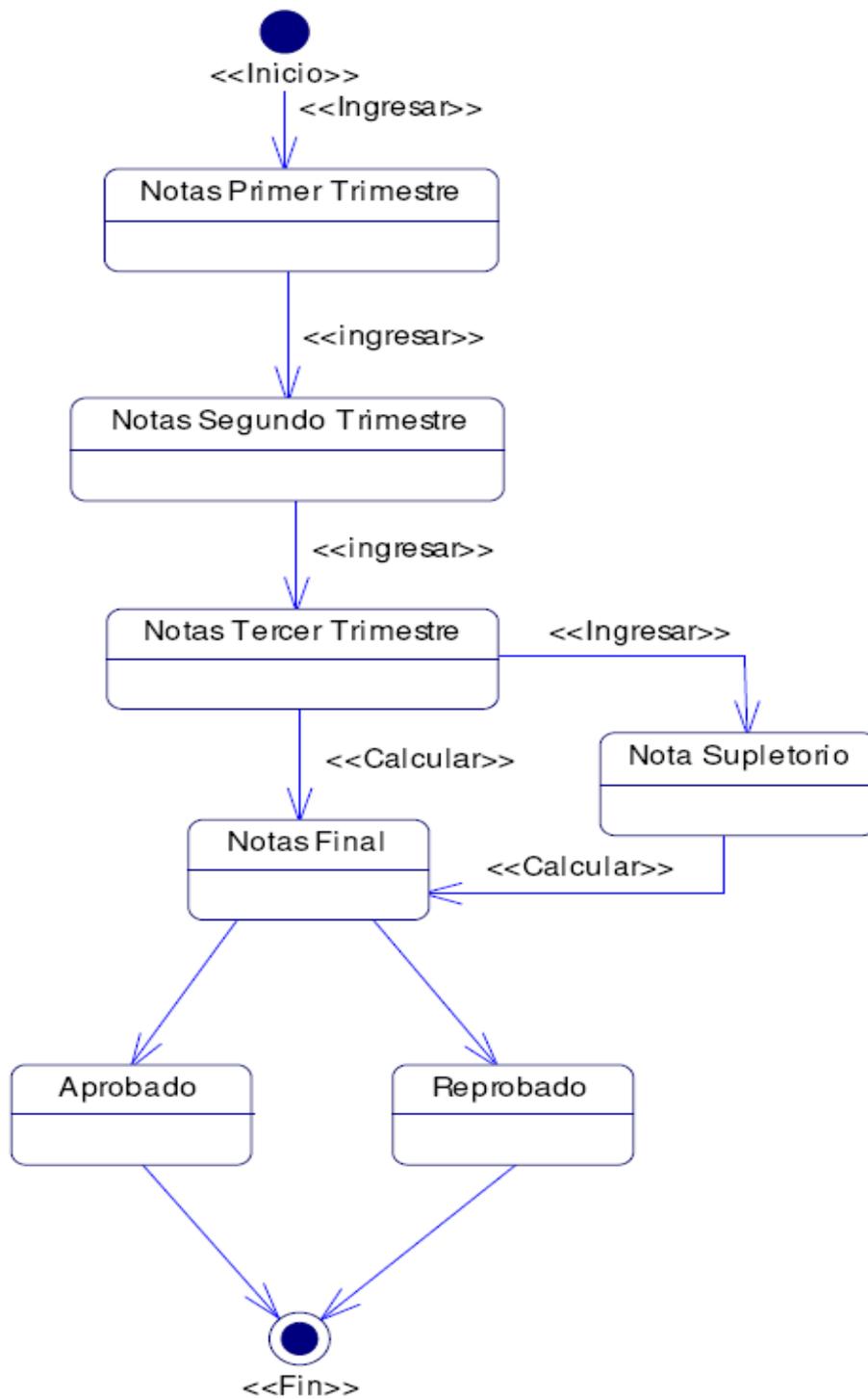


# INTERNAUTA

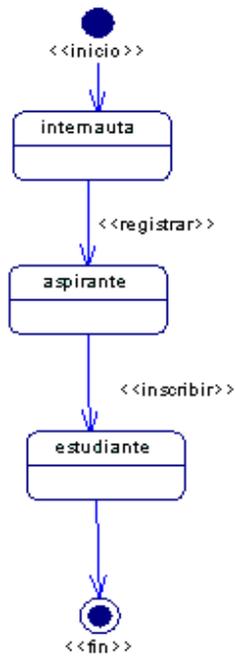


### **2.1.3 DIAGRAMAS DE ESTADOS**

#### **DIAGRAMA DE ESTADO CALIFICACIONES**

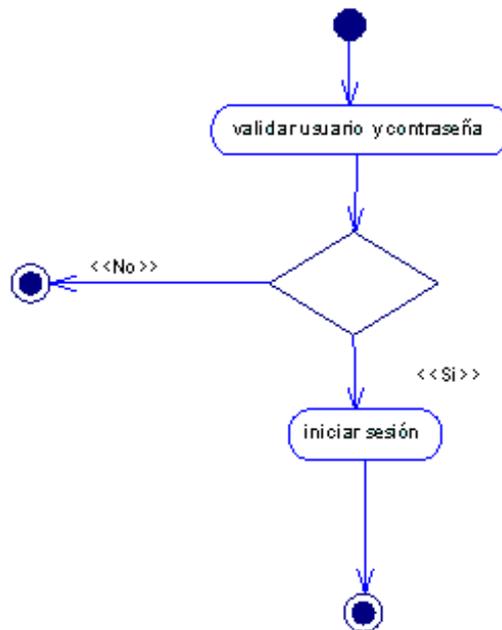


**DIAGRAMA DE ESTADOS USUARIO**

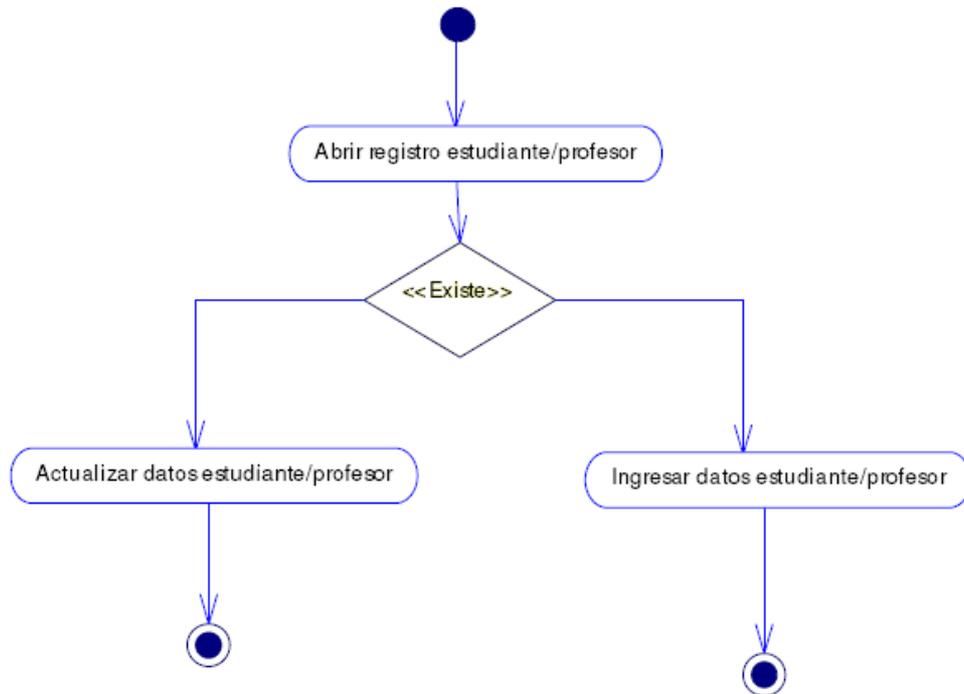


## 2.1.4 DIAGRAMAS DE ACTIVIDADES

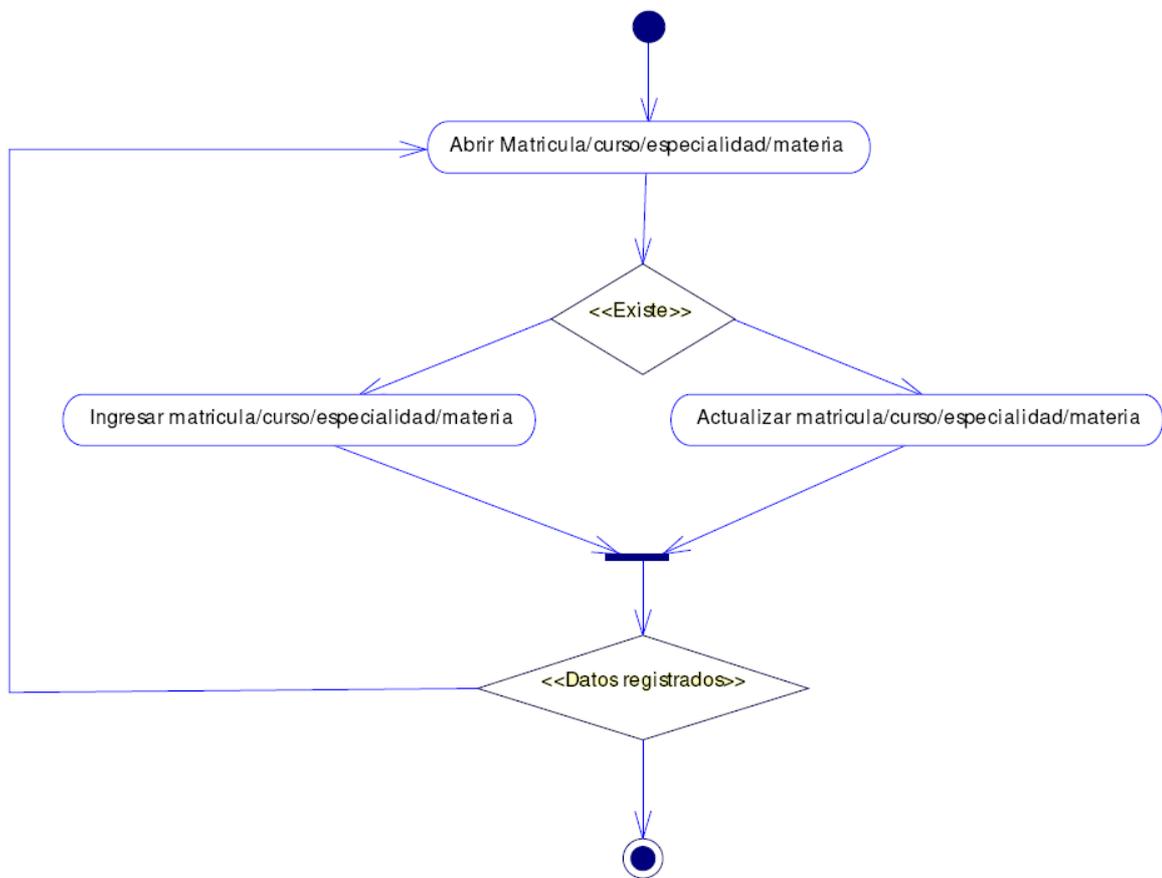
### INICIO DE SESIÓN



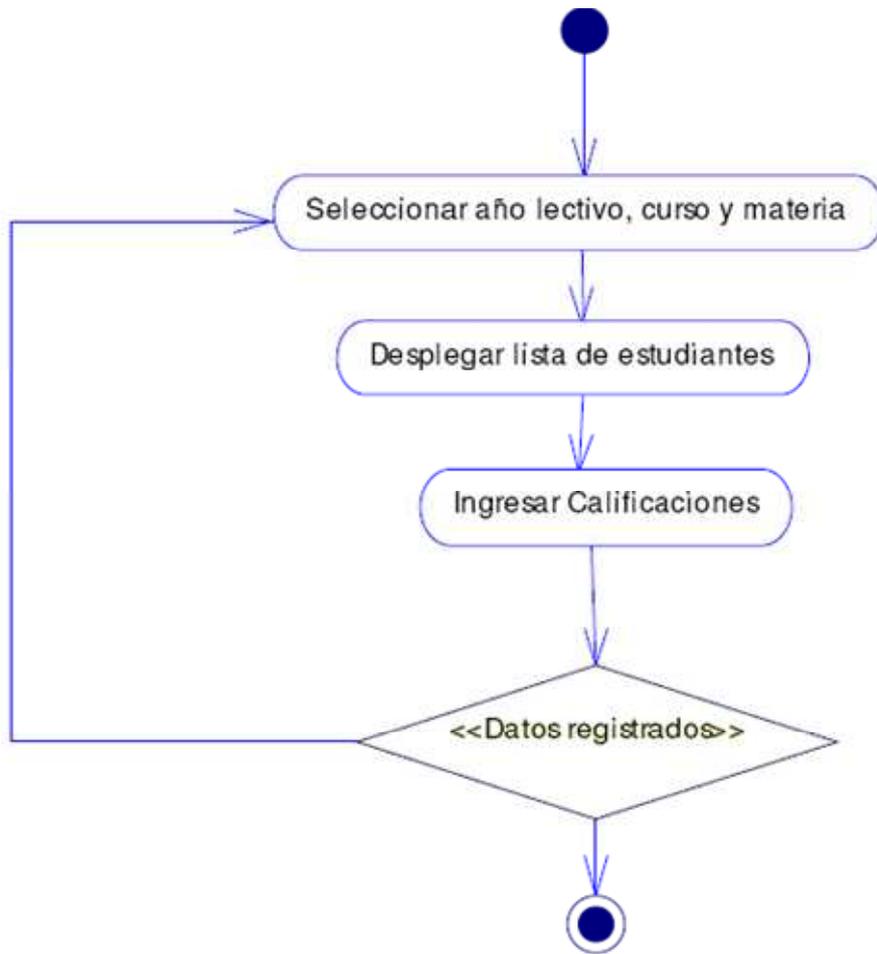
## INGRESAR DATOS ESTUDIANTE/PROFESOR



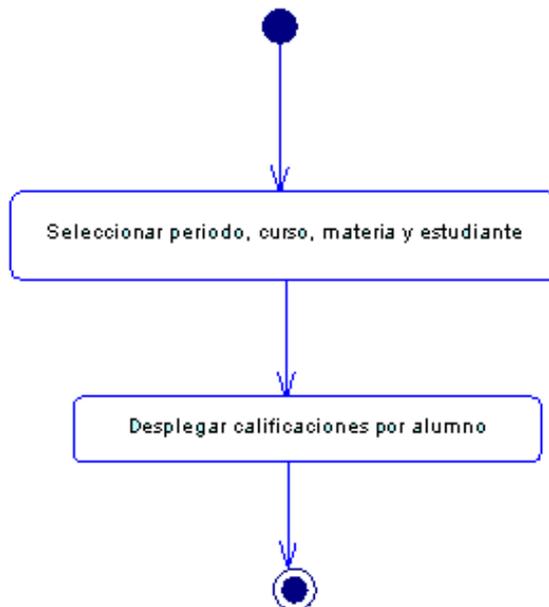
## INGRESAR DATOS MATRICULA/CURSO/ESPECIALIDAD/MATERIA



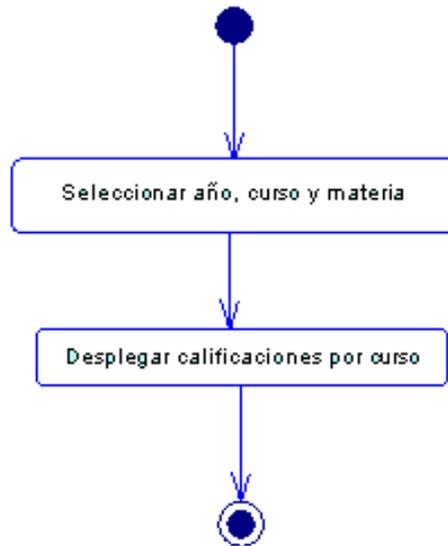
## INGRESAR CALIFICACIONES



### CONSULTAR CALIFICACIONES POR ALUMNO



## CONSULTAR CALIFICACIONES POR CURSO

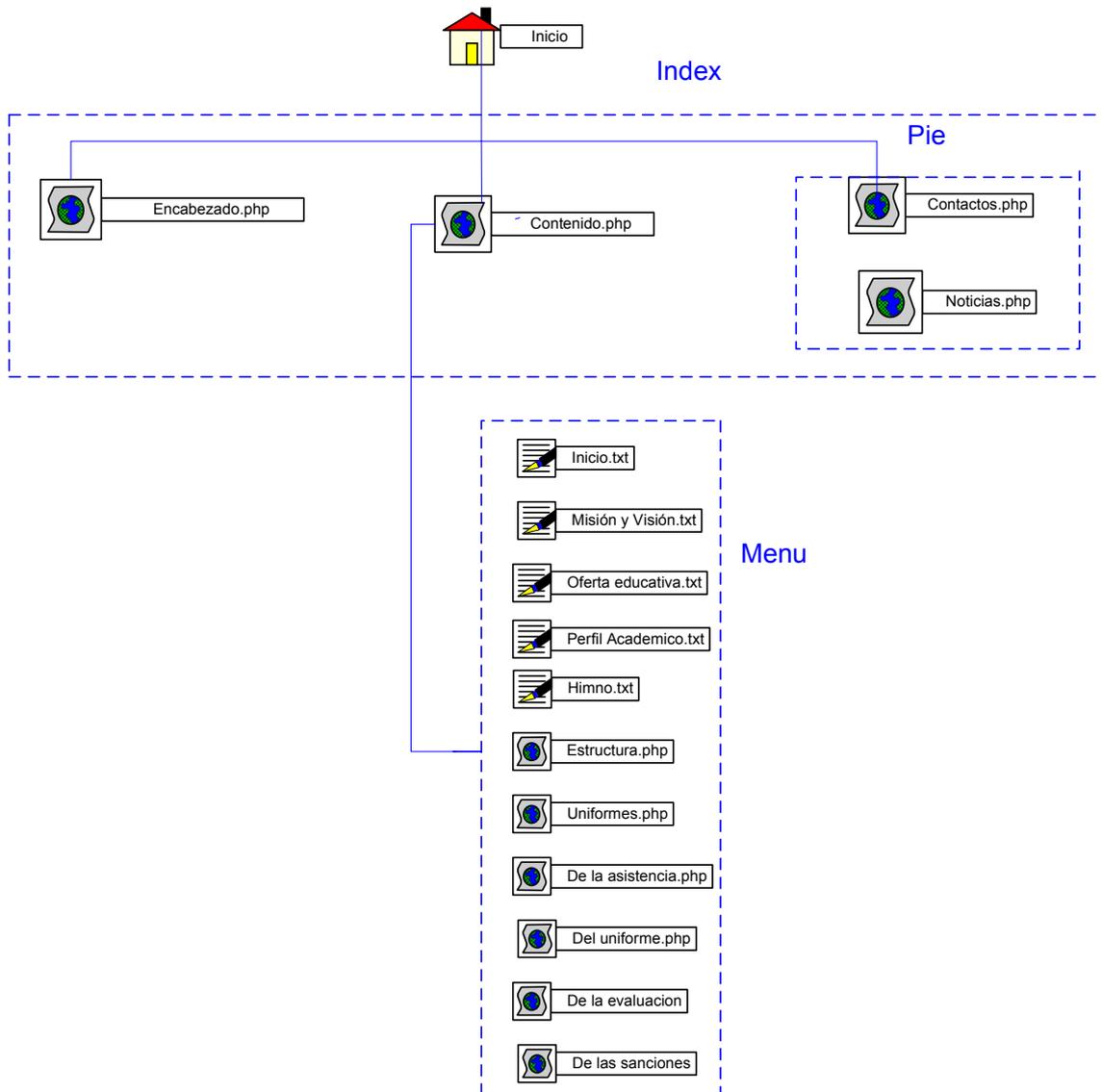


## CONSULTAR LISTA DE PROFESORES

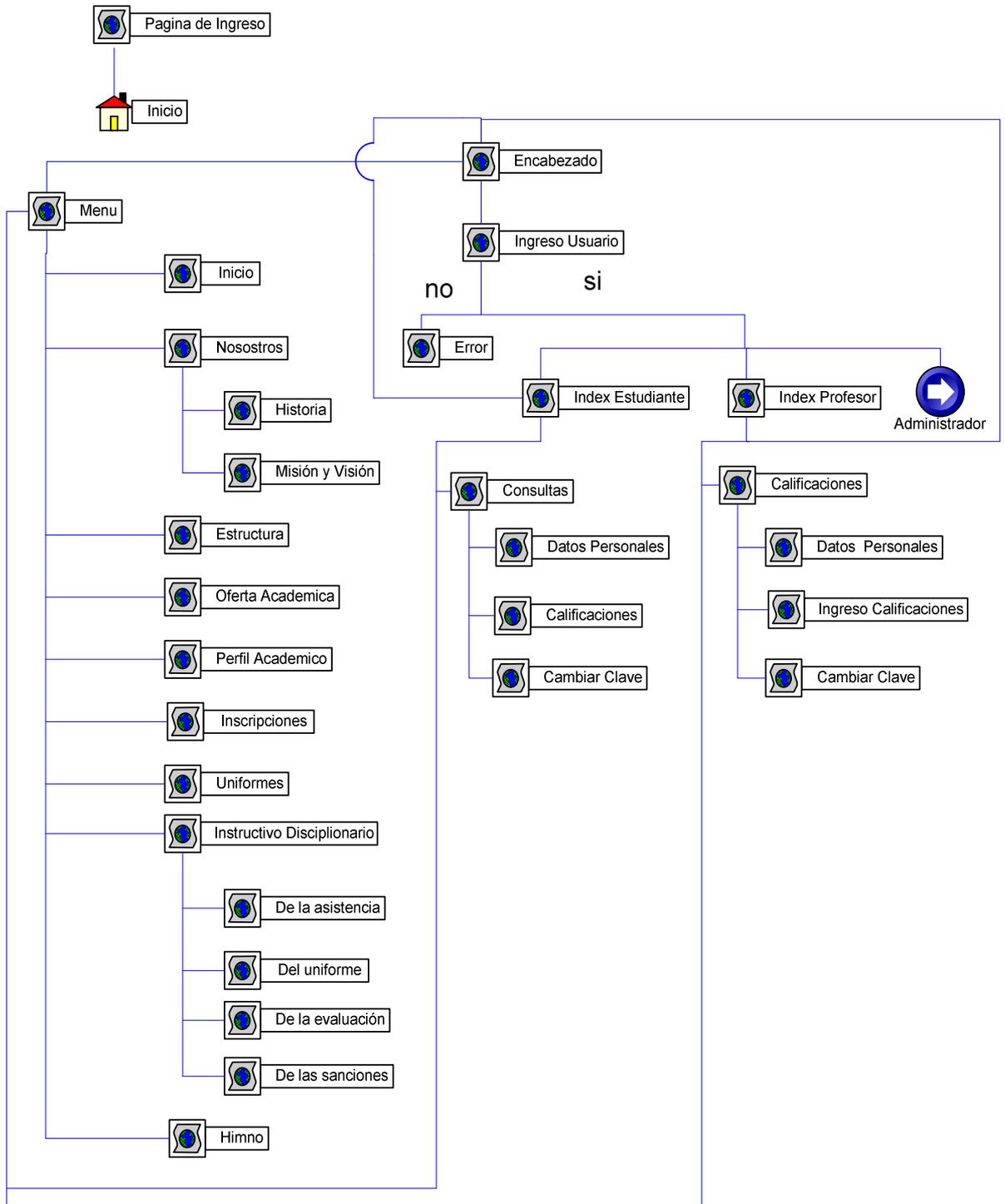


# MODELO ARQUITECTÓNICO

[Página de inicio](#)



# **MODELO NAVEGACIONAL**



# MANUAL DE USUARIO

1. ....	DISEÑO DE LA PÁGINA PRINCIPALO	
1.1	ENCABEZADO .....	1
1.2	CUERPO DE LA PÁGINA.....	1
1.3	PIE DE PÁGINA.....	2
2. ....	INGRESO DE USUARIOS.	3
2.1	USUARIO ESTUDIANTE .....	3
2.2	USUARIO PROFESOR .....	6
2.3	USUARIO ADMINISTRADOR.....	9

# 1. DISEÑO DE LA PÁGINA PRINCIPAL

El portal Web del Colegio Nacional Experimental Amazonas está diseñado con código PHP, Javascript y Dreamweaver MX 2004. A más de presentar toda la información acerca del colegio como su historia, misión, visión, requisitos de inscripción, de uniformes etc., permite que los estudiantes puedan consultar sus notas, que los profesores puedan ingresar notas y permite la administración de la página.

The screenshot shows the main page of the Colegio Nacional Experimental Amazonas website. At the top, there is a header with a navigation bar containing 'USUARIO:', 'CLAVE:', and buttons for 'Estudiante', 'Profesor', and 'Login'. Below the header is a banner image of students in a computer lab and the school's logo, 'COLEGIO EXPERIMENTAL AMAZONAS'. The main content area is framed like a document with a paperclip icon and contains a 'BIENVENIDOS' message. To the left is a 'Menu Principal' with links to 'Inicio', 'Nosotros', 'Estructura', 'Oferta Educativa', 'Perfil Académico', 'Inscripciones', 'Uniformes', 'Instructivo Disciplinario', and 'Himno'. The bottom of the page features contact information: 'Dirección: Iturralde s/n y Lauro Guerrero (Villaflores)', 'Email: colamazonas@andinanet.net', 'Quito - Ecuador', and phone numbers '2612 608', '2612 736', and '2611 611'.

**Menu Principal**

- Inicio
- Nosotros
- Estructura
- Oferta Educativa
- Perfil Académico
- Inscripciones
- Uniformes
- Instructivo Disciplinario
- Himno

**BIENVENIDOS**

COLEGIO EXPERIMENTAL AMAZONAS

El Colegio Nacional experimental Mixto Amazonas frente a los avances científicos y tecnológicos que la sociedad impone, ha emprendido una transformación integral de los estándares de calidad total, enmarcados en las nuevas normas y metodologías propias del ordenamiento jurídico y administrativo del sistema educativo, así como de los instrumentos legales, técnicos y pedagógicos, emitidos por el Ministerio de Educación y Culturas, partiendo por la realidad, analizando el pasado inmediato y preparándose planificadamente para un futuro que impacte positivamente en el desarrollo socio-económico del sur de la ciudad de Quito.

Para lo cual ofrece el bachillerato Científico Técnico que se sustenta en cambios de las estructuras de los instrumentos curriculares: pñsum, planesy programas de estudios para el bachillerato y la propuesta de un sistema de evaluacionacorde con la nueva propuesta curricular que responda a las aspiraciones de consenso de ; todos los actores internos y sec;tores del entorno social del colegio.

Dirección: Iturralde s/n y Lauro Guerrero (Villaflores)  
 Email: colamazonas@andinanet.net  
 Quito - Ecuador

2612 608  
 2612 736  
 2611 611

Figura 1 Pagina principal

## 1.1 ENCABEZADO

Esta sección se encuentra en todas las páginas de navegación. Se muestra una imagen flash cambiante, un sello de identificación del colegio, además de los campos de ingreso de usuarios, clave y su respectiva opción ya sea para estudiante o profesor.



Figura 2. Encabezado

## 1.2 CUERPO DE LA PÁGINA

En esta sección se encuentra el menú principal, el cual está diseñado con javascript y se repite en todas las páginas de navegación.

Menu Principal	
Inicio	
Nosotros	▶
Estructura	
Oferta Educativa	
Perfil Académico	
Inscripciones	
Uniformes	
Instructivo Disciplinario	▶
Himno	
	De la asistencia
	Del uniforme
	De la evaluación
	De las sanciones

Figura 3. Menú Principal

Este menú contiene información dinámica de la página; muestra un submenú según sea la opción y el requerimiento del usuario.

### 1.2.1 BOTONES DE ACCIÓN

- **Inicio.-** Permite abrir la página de inicio del sitio.
- **Nosotros.-** Despliega un submenú en el cual se puede escoger la información importante para el usuario, referente al colegio.
  - **Historia.**
  - **Misión y Visión.**
  - **Contactos.**
- **Estructura.-** Donde se encuentra información de cómo están estructurado administrativamente las autoridades del colegio.
- **Oferta Educativo.-** Despliega el contenido de información sobre las razones para elegir el colegio como opción educativa.
- **Perfil Académico.-** Muestra información acerca de línea que lleva el colegio para educar a sus estudiantes.
- **Inscripciones.-** Despliega información acerca de los requisitos de inscripción y matriculas para los estudiantes que desean ingresar al colegio.
- **Uniformes.-** Muestra el instructivo de uniformes para hombres y mujeres y sus diferentes tipos.
- **Instructivo Disciplinario.-** Abre un submenú donde se encuentra información referente a la disciplina del colegio.
- **Himno.-** Importante enlace para dar a conocer el himno del colegio a todos sus estudiantes.

### 1.3 PIE DE PÁGINA

En la parte inferior de la página se encuentra la dirección del colegio, la cuenta de email y los teléfonos de contacto.



Figura 4. Pie de página

## 2. INGRESO DE USUARIOS.

Al ingresar el usuario y su clave, el sistema valida mediante una función php si el usuario existe en la base de datos, de existir se inicia la sesión según sea el tipo de usuario: estudiante o profesor en el sistema.

En caso de no existir el usuario se presenta un mensaje de error.

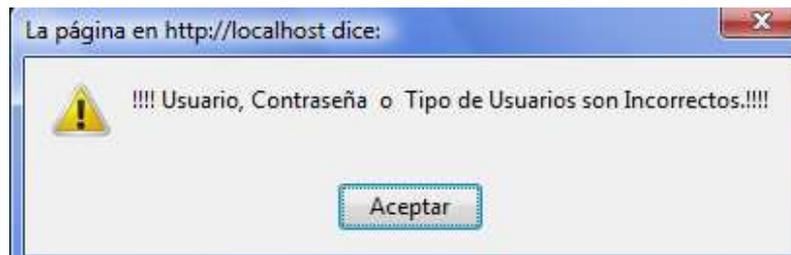


Figura 5. Mensaje de error

### 2.1 USUARIO ESTUDIANTE

Se muestra una interfaz parecida a la de la página de inicio con un menú adicional del estudiante.

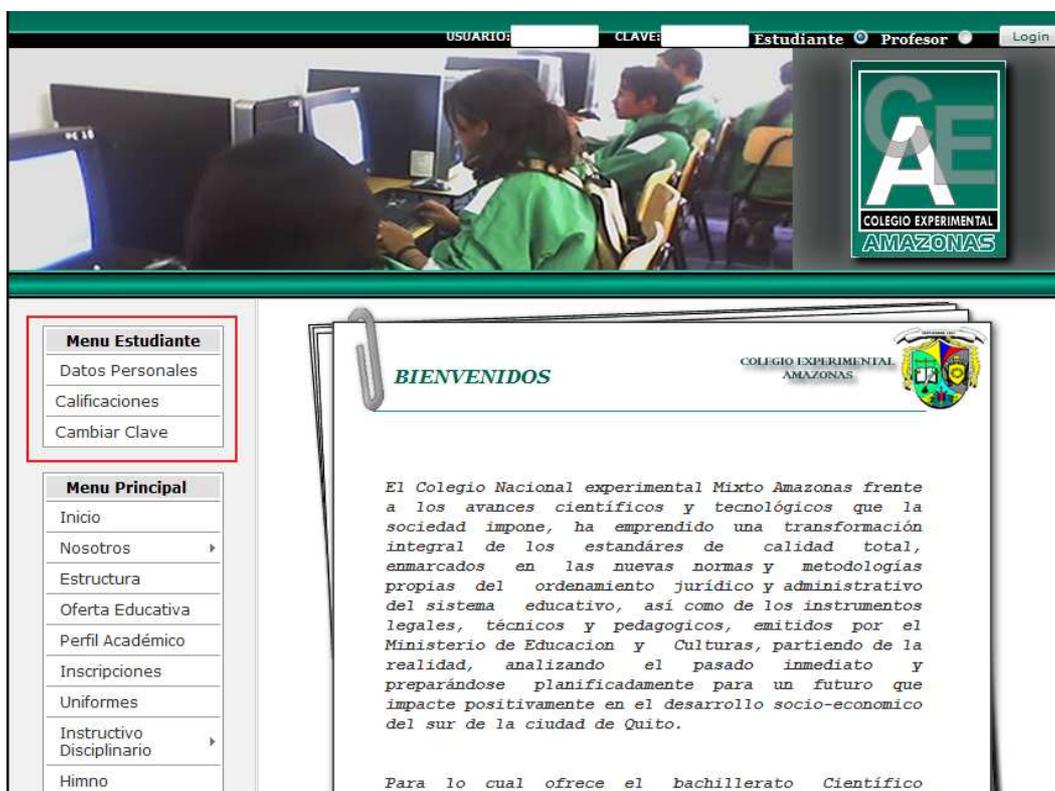


Figura 6. Usuario Estudiante

- **Datos Personales.-** Muestra una página con información personal del estudiante.

**Menu Estudiante**

- Datos Personales
- Calificaciones
- Cambiar Clave

**Menu Principal**

- Inicio
- Nosotros ▶
- Estructura
- Oferta Educativa
- Perfil Académico
- Inscripciones
- Uniformes
- Instructivo Disciplinario ▶
- Himno

**DATOS PERSONALES**

COLEGIO EXPERIMENTAL AMAZONAS

**NOMBRE:** ANDERSON RAMIRO ALMEIDA ARCOS

**C.I. :** 1718635452

**NACIONALIDAD :** Ecuatoriana

**FECHA DE NACIMIENTO:** 1996-02-01

**GÉNERO:** masculino

**DIRECCIÓN:** Cotocollao

**TELÉFONO:** 2654345

**CORREO ELECTRÓNICO:** aalmeida@yahoo.com

**REPRESENTANTE:** SRA. CARLA ARCOS

Dirección: Iturralde s/n y Lauro Guerrero (Villaflora)  
 Email: colamazonas@andinanet.net  
 Quito - Ecuador

2612 608  
 2612 736  
 2611 611

**Figura 7.** Datos Personales del Estudiante

- **Calificaciones.** Muestra una página con el reporte de calificaciones que incluye información de la sección, curso, paralelo, especialidad y todas las materias que el estudiante está tomando y sus respectivas notas.

**Reporte de Calificaciones** COLEGIO EXPERIMENTAL AMAZONAS

**Periodo: 2007 - 2008**

**Nombres:** ALMEIDA ARCOS      **Seccion:** MATUTINO  
**Apellidos:** ANDERSON RAMIRO      **Curso:** PRIMERO  
**Fecha:** 20-09-08      **Paralelo:** D  
**Especialidad:** S/E

MATERIA	I.	II	III	SUPLET.
<i>Ciencias Naturales</i>	o	o	o	
<i>Matematicas</i>	o	o	o	
<i>Idioma Extranjero</i>	o	o	o	
<i>Lenguaje y Comunicacion</i>	o	o	o	
<i>Computacion</i>	o	o	o	
<i>Estudios Sociales</i>	o	o	o	
<i>Cultura Fisica</i>	o	o	o	
<i>Musica</i>	o	o	o	
<i>Tecnicas de Estudio</i>	o	o	o	

**Figura 8.** Reporte de Calificaciones

- **Cambio de Clave.-** Al hacer click en el enlace se muestra una pantalla en donde se debe ingresar la contraseña antigua, la nueva y la confirmación para que pueda ser cambiada.

**CAMBIAR CONTRASEÑA** COLEGIO EXPERIMENTAL AMAZONAS

**Contraseña Antigua:**   
**Contraseña Nueva:**   
**Confirmar contraseña:**

**Figura 9.** Cambiar Contraseña Estudiante

## 2.2 USUARIO PROFESOR

Se muestra una interfaz parecida a la de la página de inicio con un menú adicional del profesor.

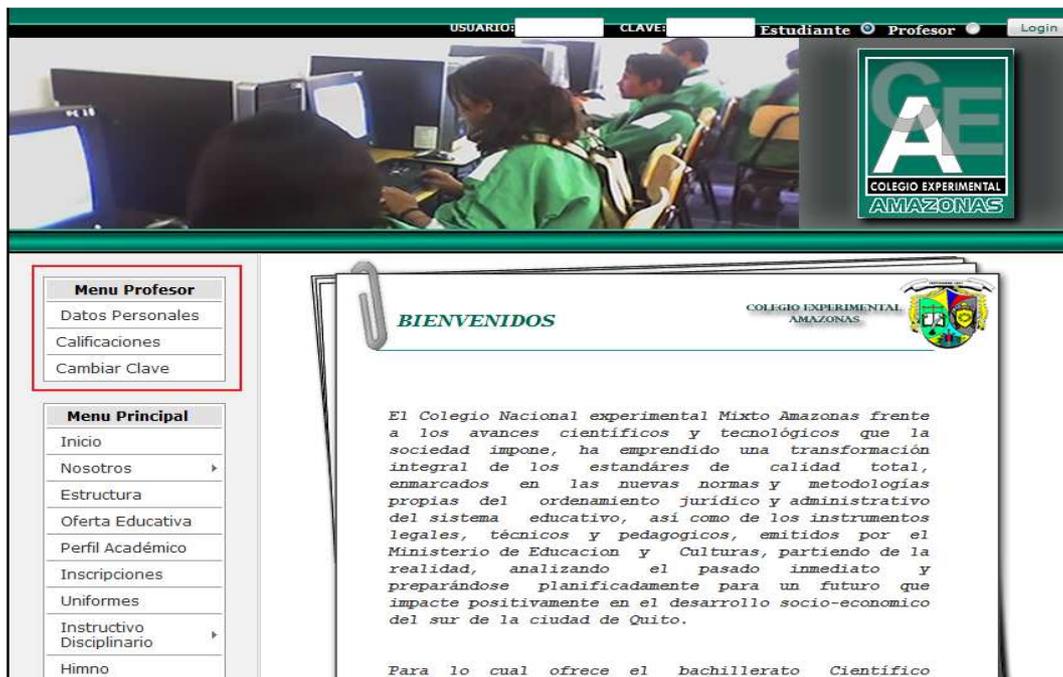


Figura 10. Usuario Profesor

- **Datos Personales.** Muestra una página con los datos personales del profesor.

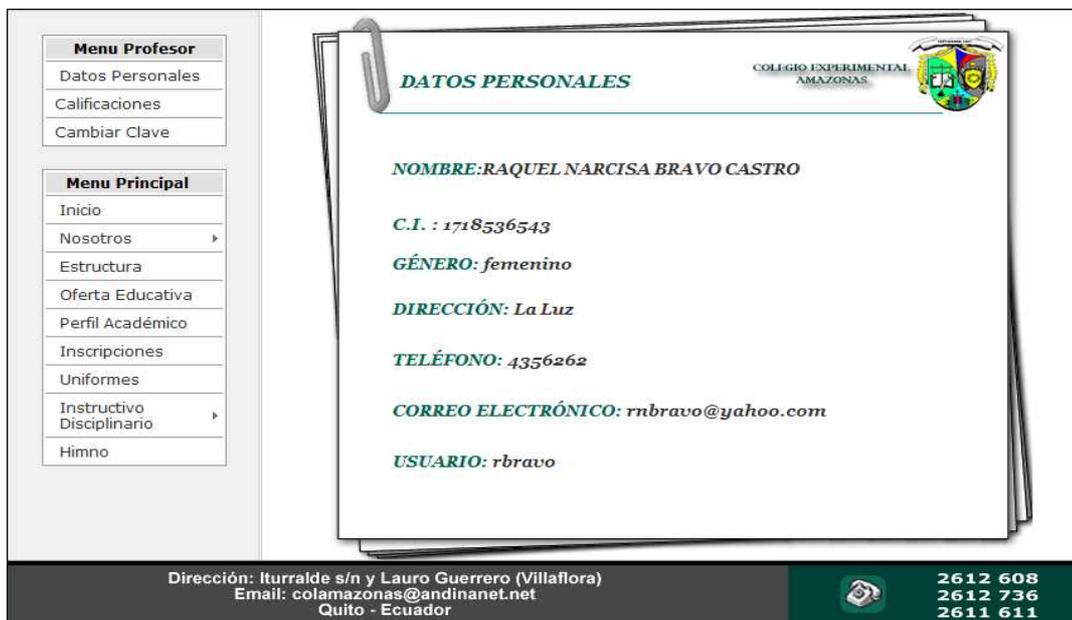


Figura 11. Datos Personales del Profesor

- **Calificaciones.-** Muestra una página con información del periodo, las materias, el curso y el paralelo que actualmente el profesor está dictando.

**Menu Profesor**

- Datos Personales
- Calificaciones
- Cambiar Clave

**Menu Principal**

- Inicio
- Nosotros
- Estructura
- Oferta Educativa
- Perfil Académico
- Inscripciones
- Uniformes
- Instructivo Disciplinario
- Himno

**Materias Dictadas**

Período: 2007 - 2008

Materias que actualmente usted dicta:

MATERIA	CURSO	PARALELO
-> <a href="#">Musica</a>	Primero	A
-> <a href="#">Musica</a>	Primero	B
-> <a href="#">Musica</a>	Primero	C
-> <a href="#">Musica</a>	Primero	D
-> <a href="#">Musica</a>	Primero	E

Dirección: Iturralde s/n y Lauro Guerrero (Villaflores)  
Email: colamazonas@andinanet.net  
Quito - Ecuador

2612 608  
2612 736  
2611 611

**Figura 12.** Materias Dictadas

Al hacer click en una de la materia de la lista se despliega una página con el listado de los estudiantes que toman dicha materia y los campos de texto para que el profesor ingrese las notas de cada trimestre y la nota de supletorio.

**Menu Profesor**

- Datos Personales
- Calificaciones
- Cambiar Clave

**Menu Principal**

- Inicio
- Nosotros
- Estructura
- Oferta Educativa
- Perfil Académico
- Inscripciones
- Uniformes
- Instructivo Disciplinario
- Himno

**Nómina de Estudiantes**

Período: 2007 - 2008

Materia : [Musica](#)      Curso: [Primero](#)      Paralelo: [A](#)  
Materia : [Musica](#)      Curso: [Primero](#)      Paralelo: [B](#)  
Materia : [Musica](#)      Curso: [Primero](#)      Paralelo: [C](#)  
Materia : [Musica](#)      Curso: [Primero](#)      Paralelo: [D](#)  
Materia : [Musica](#)      Curso: [Primero](#)      Paralelo: [E](#)

NOMBRES	NOTAS			
	1er.	2da.	3ra.	Sup.
ACOSTA FUENTES JOHANA CAROLINA		XXX	XXX	XXX

GUARDAR

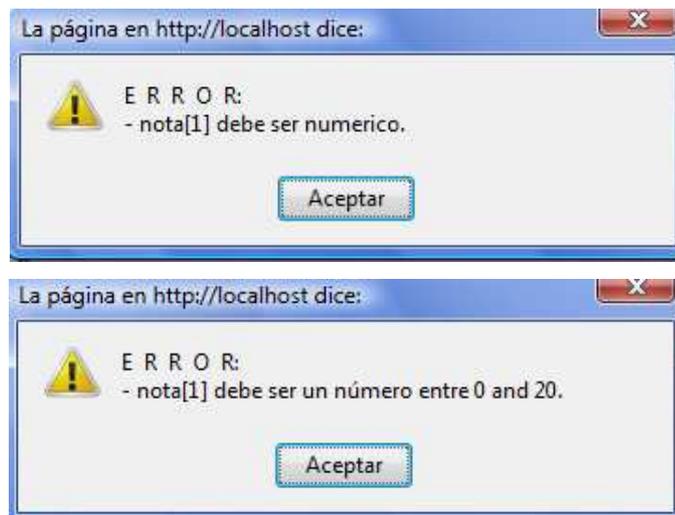
Dirección: Iturralde s/n y Lauro Guerrero (Villaflores)  
Email: colamazonas@andinanet.net  
Quito - Ecuador

2612 608  
2612 736  
2611 611

**Figura 13.** Ingreso de notas

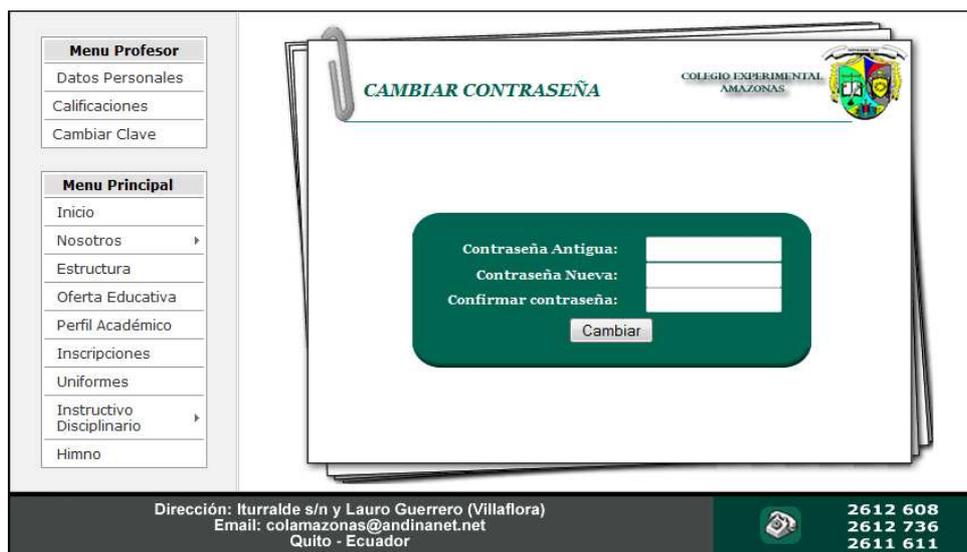
Al guardar las notas se llama a una función php, la cual valida las notas ingresadas y si existe un error emite un mensaje según sea la falta efectuada.

Si no se ingresa la nota se ingresan letras o no están en el rango de 0 a 20 se muestran los siguientes cuadros de error.



**Figura 14** Mensaje de error en el ingreso de nota

- **Cambiar Clave.-** Al hacer clic en el enlace se muestra una pantalla en donde se debe ingresar la contraseña antigua, la nueva y una confirmación de contraseña para que pueda ser cambiada.



**Figura 15.** Cambiar contraseña Profesor

## 2.3 USUARIO ADMINISTRADOR

Para ingresar a la administración de la página, esta creada una nueva interfaz con características distintas a las de la página de inicio; como la que se muestra a continuación.



**Figura 16.** Ingreso al Sistema de Administración.

### 2.3.1 MENÚ ADMINISTRACIÓN DE LOS DATOS

Una vez ingresado el usuario que en este caso se validó la cédula y la contraseña, se muestra la interfaz de la figura 17.

En este menú se encuentran los diferentes vínculos a los cuales el administrador va

a tener acceso, para modificar los datos que los usuarios ingresan además de la información del colegio en la base de datos.

- **Intuición.-** Este vínculo contiene todo lo referente a la información del colegio. Las pestañas que aquí aparecen dan lugar a cambiar dicha información por si es necesario lo que se denomina una página dinámica.

The screenshot displays the 'Sistema de Administración' interface for Colegio Experimental Amazonas. At the top, there is a green header with the school's logo and name. Below the header is a navigation menu with tabs for 'Institución', 'Estudiantes', 'Profesores', 'Materias', 'Especialidades', and 'Notas'. The main content area is titled 'Institución - Datos' and contains a sub-menu with tabs for 'Datos', 'Historia', 'Misión', 'Visión', 'Perfil', and 'Himno'. The 'Datos' tab is active, showing a form with the following fields:

Nombre:	Colegio Nacional experimental
Dirección:	Iturralde s/n y Lauro Guerrero
Telefono:	2612 608 - 2612 736 - 2611 611
Mail:	colamazonas@andinanet.net
Rector:	Dr. Jorge Urgiles Msc.
Vicerector1:	Mat. Vicente Parra Msc.
Vicerector2:	Lic. Ernesto Becerra
Inspector G.:	Lic Carlos Peña

At the bottom of the form is a 'Modificar' button.

Figura 17. Sistema de Administración

- **Estudiantes.-** Al hacer click en **estudiante**, se despliega una página en la cual se listan todos los estudiantes que se han registrado, un link en la parte superior izquierda para el ingreso de nuevos estudiantes y tres iconos, uno para modificar información del estudiante , otro para borrar al estudiante  y el otro para inscribir al mismo . Además de el icono para imprimir lista de alumnos por curso y materia  imprimir

**Nuevo estudiante.-** Este enlace muestra un formulario donde se registra la información del estudiante a ser guardada en la base de datos.

COLEGIO EXPERIMENTAL AMAZONAS

Colegio Experimental Amazonas  
Sistema de Administración

Institución Estudiantes Profesores Materias Especialidades Notas

**Nuevo Estudiante**

ESTUDIANTES

NOMBRE :

APELLIDO:

CEDULA :

NACIONALIDAD :

FECHA NAC.:

GENERO: FEMENINO ▾

REPRESENTANTE.:

DIRECCION:

TELEFONO:

MAIL:

USUARIO:

PASSWORD:

Guardar

Figura 18. Registro Nuevo Estudiante.

**Modificar Información del Estudiante.-** Al hacer click en el icono modificar  se despliega un formulario con los datos del estudiante previamente ingresados, los mismos que van a ser modificados.

COLEGIO EXPERIMENTAL AMAZONAS

Colegio Experimental Amazonas  
Sistema de Administración

Institución Estudiantes Profesores Materias Especialidades Notas

ESTUDIANTE

NOMBRE : Jenifer Elisa

APELLIDO: Almache Molineros

CEDULA : 1718637654

FECHA NAC. : 0000-00-00

NACIONALIDAD : Ecatoriana

GENERO: FEMENINO ▾

REPRESENTANTE : Sr.Cesar Almache

DIRECCION: La Gasca

TELEFONO: 26354876

MAIL:

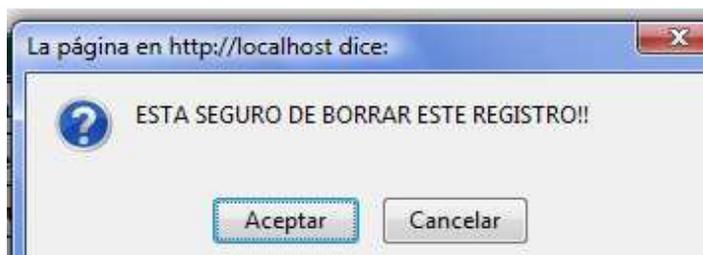
USUARIO: jalmache

PASSWORD: ••••••••

Guardar

Figura 19. Modificar Datos del Estudiante.

**Borrar Estudiante.** Al hacer click en el icono borrar  se despliega un mensaje, el cual se debe aceptar si están seguros de la acción a realizar.



**Figura 20.** Confirmar Borrar Datos del Estudiante.

**Inscribir Estudiante.** Al hacer click en  se despliega una página en la cual se elije la sección, curso, especialidad y paralelo del estudiante al cual se va a inscribir.

**Figura 21.** Inscripción del Estudiante.

**Imprimir.** Con este icono  se despliega una ventana, desde donde se puede imprimir reporte de alumnos por curso y por materia según se necesite.



**Figura 22.** Impresión de Reportes.

- **Profesores.-** Al hacer click en **profesor** se despliega una página en la cual se listan todos los profesores que se encuentran dictando clases en el colegio.

En la parte superior izquierda están los enlaces para crear nuevo profesor y asignar materias a los profesores que constan en la lista. A la derecha de cada profesor se encuentran iconos, uno para modificar información del profesor  y otro para borrar al profesor . En la parte inferior un vínculo que abre una ventana para imprimir lista de profesores.

**Nuevo Profesor.** Este enlace muestra un formulario donde se registra la información del profesor a ser guardada en la base de datos.

**Colegio Experimental Amazonas**  
Sistema de Administración

**Institución**   **Estudiantes**   **Profesores**   **Materias**   **Especialidades**   **Notas**

**Nuevo Profesor**

**PROFESOR**

NOMBRE :

APELLIDO:

CEDULA :

GENERO: FEMENINO ▾

DIRECCION:

TELEFONO:

MAIL:

USUARIO:

PASSWORD:

**Figura 23.** Registro Nuevo Profesor.

**Asignar Materia.** Este vínculo abre una ventana que muestra varios combos desplegable; donde se escoge entre otras opciones al profesor y la materia que va a dictar en dicho periodo.

**Colegio Experimental Amazonas**  
Sistema de Administración

**Institución**   **Estudiantes**   **Profesores**   **Materias**   **Especialidades**   **Notas**

**Asignar Materia**

Profesor:

Sección:

Curso:

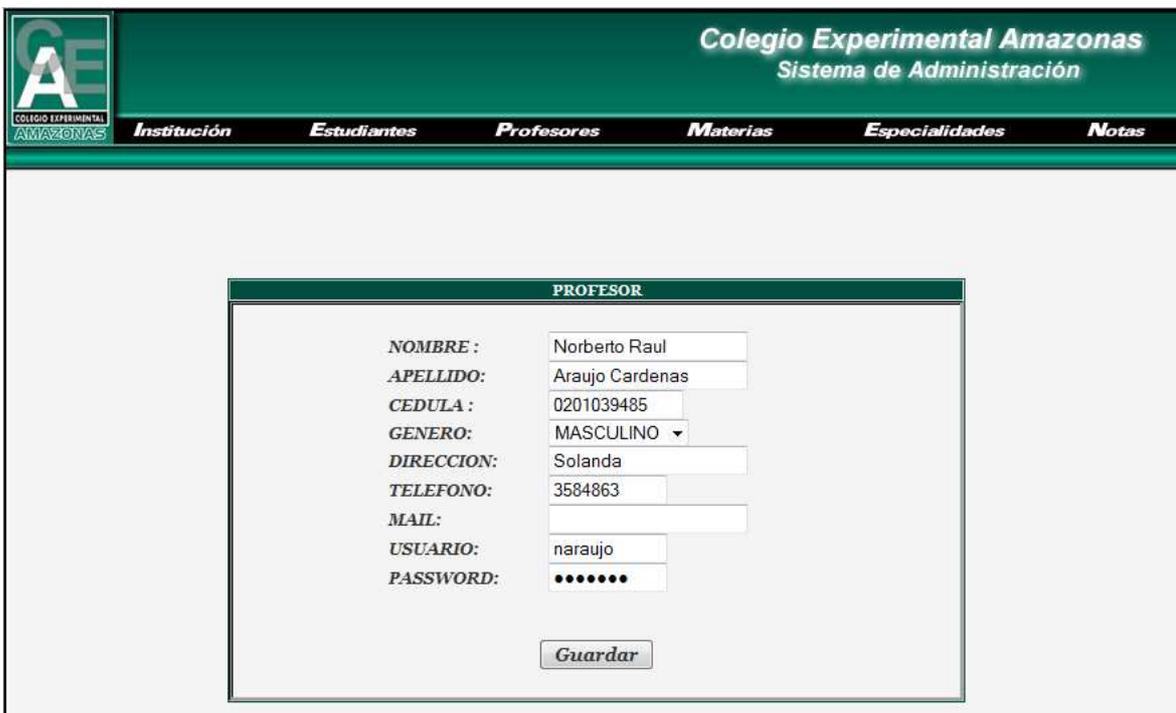
Especialidad:

Materia:

Paralelo:

**Figura 24.** Asignar Materias al Profesor.

**Modificar Información del Profesor.** Al hacer click en el icono modificar , se despliega un formulario con los datos del profesor previamente ingresados; los mismos que van a ser modificados.



Logo: **CE** Colegio Experimental AMAZONAS

Colegio Experimental Amazonas  
Sistema de Administración

Institución    Estudiantes    **Profesores**    Materias    Especialidades    Notas

**PROFESOR**

NOMBRE : Norberto Raul

APELLIDO: Araujo Cardenas

CEDULA : 0201039485

GENERO: MASCULINO ▾

DIRECCION: Solanda

TELEFONO: 3584863

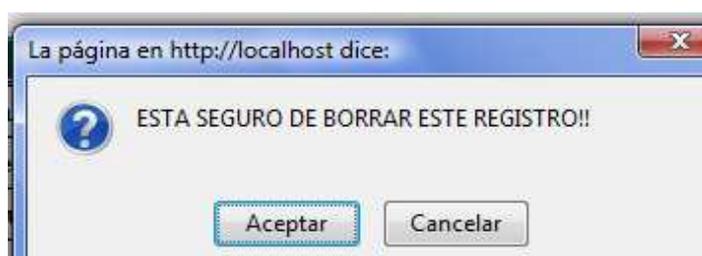
MAIL:

USUARIO: naraujo

PASSWORD: ●●●●●●

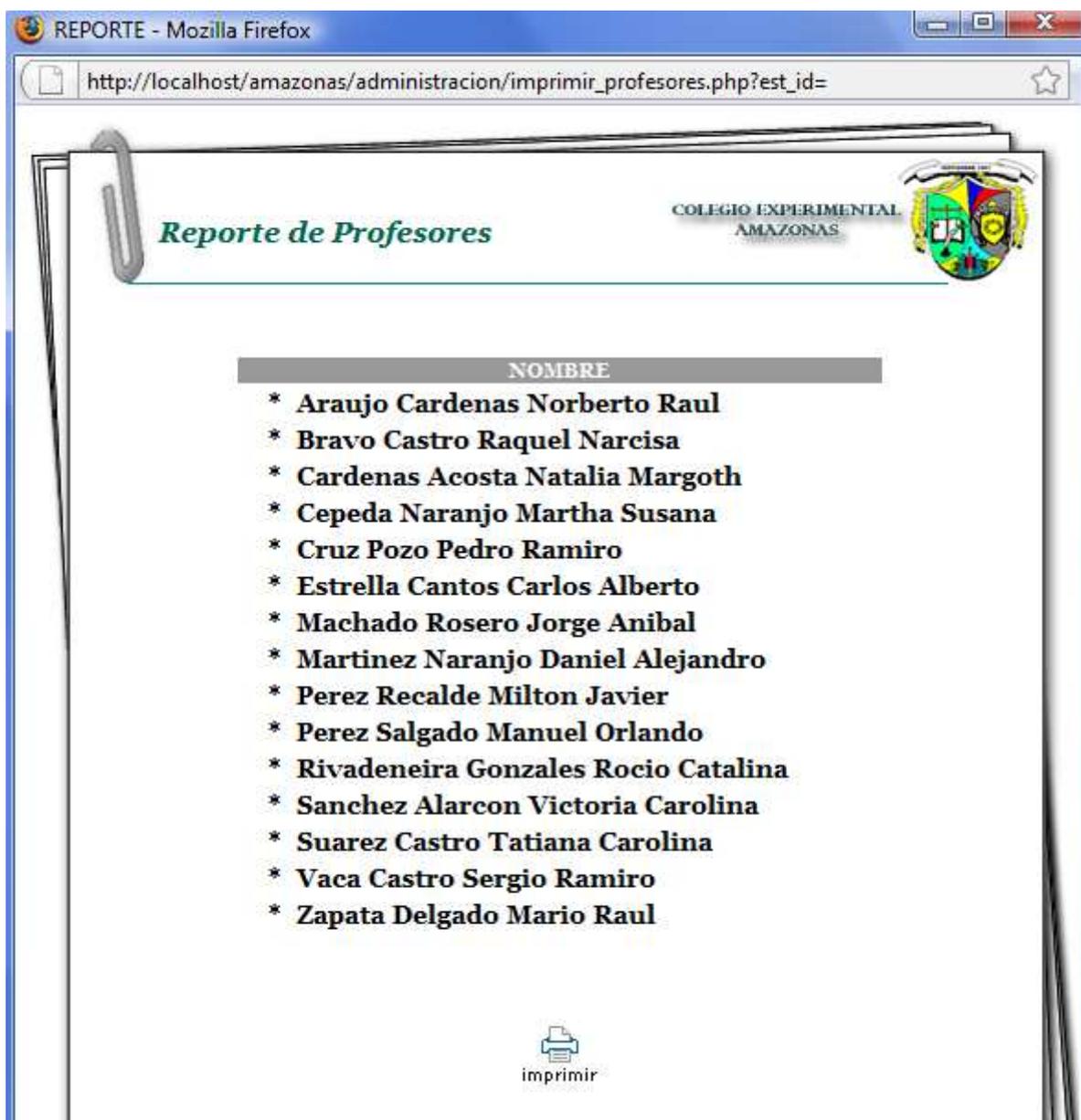
**Figura 25.** Modificar Datos del Profesor.

**Borrar Estudiante.** Al hacer click en el icono borrar , se despliega un mensaje, el cual se debe aceptar si están seguros de la acción a realizar.



**Figura 26.** Confirmar Borrar Datos del Profesor.

**Imprimir.** Con este icono  , se despliega una ventana desde donde se puede imprimir un reporte de todos los profesores que están dictando clases en el establecimiento



**Figura 27.** Reporte de Profesores.

- **Materias.** Al hacer click en **materia** se despliega una página en la cual se listan todas las materias que se dictan en el colegio. En la parte superior izquierda aparece un enlace para ingresar nuevas materias. En la parte derecha de cada materia se tiene dos iconos, el uno para borrar la materia  y otro para modificarla .

**Nueva Materia.** Al hacer click en el vínculo nueva materia, se despliega una página en el cual se debe elegir el nombre de la materia, el curso, la sección y la especialidad a cual va pertenecer la nueva materia.

CE  
COLEGIO EXPERIMENTAL  
AMAZONAS

Colegio Experimental Amazonas  
Sistema de Administración

Institución Estudiantes Profesores **Materias** Especialidades Notas

**Nueva Materia**

MATERIA

Nombre:

Sección: \*\*\*\*\* ▼

Curso: \*\*\*\*\* ▼

Especialidad: \*\*\*\*\* ▼

Enviar

**Figura 27.** Creación de Nueva Materia.

**Modificar Materia.** Al hacer click en el icono modificar  se despliega una pantalla con los datos de la materia, los cuales se pueden modificar y guardarlos.

CE  
COLEGIO EXPERIMENTAL  
AMAZONAS

Colegio Experimental Amazonas  
Sistema de Administración

Institución Estudiantes Profesores **Materias** Especialidades Notas

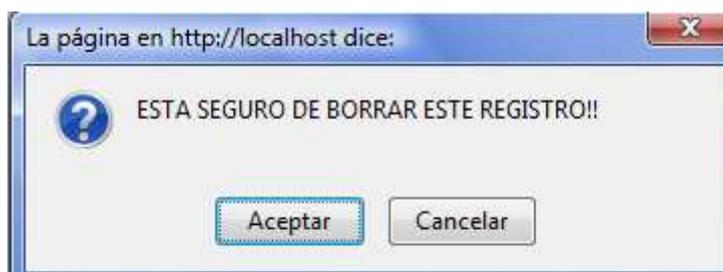
MATERIA

NOMBRE MATERIA: Idioma Extranjero

Guardar

**Figura 28.** Modificar nombre de la Materia.

**Borrar Materia.** Al hacer click en el icono borrar , se despliega un mensaje, el cual se debe aceptar si se está seguro de la acción a realizar.



**Figura 29.** Borrar la Materia.

- **Especialidad.** Al hacer click en **especialidad**, se despliega una página en la cual se listan todas las especialidades que ofrece el colegio a sus estudiantes.

En la parte superior izquierda de cada especialidad se tiene un vínculo que abre una página para ingresar nuevas especialidades. Además de dos iconos, el uno para borrar la especialidad  y el otro para modificar .



**Figura 30.** Nueva Especialidad

**Nueva Especialidad.** Al hacer click en el vínculo nueva especialidad se despliega un formulario el cual se debe llenar con los datos correspondientes y guardarlos.

Logo: **CE** COLEGIO EXPERIMENTAL AMAZONAS

Header: **Colegio Experimental Amazonas**  
Sistema de Administración

Navigation: **Institución** **Estudiantes** **Profesores** **Materias** **Especialidades** **Notas**

Section: **Nueva Especialidad**

Form Title: **ESPECIALIDAD**

Fields: **NOMBRE:** [Text Box] **DETALLE:** [Text Area]

Button: **Guardar**

**Figura 31.** Nueva Especialidad

**Modificar Especialidad.** Al hacer click en el icono modificar  se despliega una pantalla con los datos actuales de la especialidad, los cuales podemos modificar y guardarlos.

Logo: **CE** COLEGIO EXPERIMENTAL AMAZONAS

Header: **Colegio Experimental Amazonas**  
Sistema de Administración

Navigation: **Institución** **Estudiantes** **Profesores** **Materias** **Especialidades** **Notas**

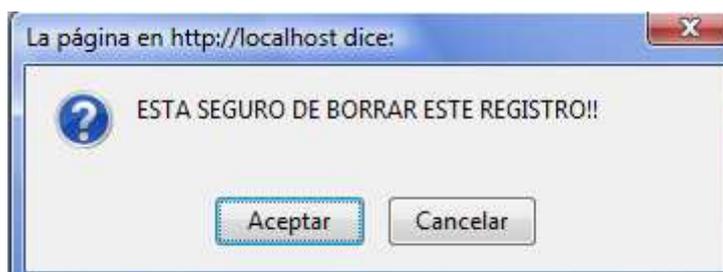
Form Title: **ESPECIALIDAD**

Fields: **NOMBRE:** **Ciencias Sociales** **DETALLE:** **www**

Button: **Guardar**

**Figura 32.** Modificar una Especialidad

**Borrar Especialidad.** Al hacer click en el icono borrar  se despliega un mensaje, el cual se debe aceptar si están seguros de la acción a realizar.



**Figura 33.** Borrar la Especialidad.

- **Notas.-** este enlace muestra una lista con todos los estudiantes existentes en el Colegio.

ESTUDIANTES
<a href="#">Almache Molineros Jenifer Elisa</a>
<a href="#">Almeida Arcos Anderson Ramiro</a>
<a href="#">Añarumba Sigua Jonathan Bryan</a>
<a href="#">Betancourt Delgado Diana Alexandra</a>
<a href="#">Caiza Toapanta Estefania Abigail</a>

**Figura 34.** Lista de Estudiantes.

Al hacer clic en uno de los estudiantes se abre una nueva página donde se muestra cuadros de texto pertenecientes a cada materia en cada periodo (nota 1, nota 2, nota 3, y supletorio); las mismas que pueden ser modificadas una a una según sea necesario.

Cabe indicar que estas notas serán cambiadas previas a una autorización de las autoridades pertinentes.

**Colegio Experimental Amazonas**  
 Sistema de Administración

Institución
Estudiantes
Profesores
Materias
Especialidades
Notas

Almeida Arcos Anderson Ramiro

**MATERIAS**

NOMBRE	NOTA 1	NOTA 2	NOTA 3	SUPLERORIO
CIENCIAS NATURALES	0 <input type="button" value="Cambiar"/>			
MATEMATICAS	0 <input type="button" value="Cambiar"/>			
IDIOMA EXTRANJERO	0 <input type="button" value="Cambiar"/>			
LENGUAJE Y COMUNICACION	0 <input type="button" value="Cambiar"/>			
COMPUTACION	0 <input type="button" value="Cambiar"/>			
ESTUDIOS SOCIALES	0 <input type="button" value="Cambiar"/>			
CULTURA FISICA	0 <input type="button" value="Cambiar"/>			
MUSICA	0 <input type="button" value="Cambiar"/>			
TECNICAS DE ESTUDIO	0 <input type="button" value="Cambiar"/>			

  
 imprimir

**Figura 35.** Notas en cada Materia.

**Imprimir Reporte de Notas.** En la parte inferior de la página donde se muestran las calificaciones, existe un icono de imprimir . Este icono muestra un reporte de las calificaciones del estudiante el mismo que se puede imprimir de ser requerido.

REPORTES - Mozilla Firefox

http://localhost/amazonas/administracion/imprimir\_notas.php?est\_id=4

**Reporte de Calificaciones**

COLEGIO EXPERIMENTAL  
AMAZONAS

Periodo: 2007 - 2008

Nombres: ALMEIDA ARCOS      Seccion: MATUTINO  
 Apellidos: ANDERSON RAMIRO      Curso: PRIMERO  
 Fecha: 21-09-08      Paralelo: D  
 Especialidad: S/E

MATERIA	I.	II	III	SUPLET.
<i>Ciencias Naturales</i>	0	0	0	
<i>Matematicas</i>	0	0	0	
<i>Idioma Extranjero</i>	0	0	0	
<i>Lenguaje y Comunicacion</i>	0	0	0	
<i>Computacion</i>	0	0	0	
<i>Estudios Sociales</i>	0	0	0	
<i>Cultura Fisica</i>	0	0	0	

Figura 35. Reporte de Notas del Estudiante.