

**Universidad de las Ciencias Informáticas
“Facultad 5”**



**Título: Módulo de visualización interactiva
de recursos en prácticas de laboratorios virtuales
para aplicaciones Web.**

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

**Autor: Anddy Castellanos Huerta
Tutor: Luís G. Viciado Caraballosó.**

**Ciudad de la Habana
Julio 2007**

DECLARACIÓN DE AUTORÍA

Declaro ser autor de la presente tesis y reconozco a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmo la presente a los ____ días del mes de _____ del año _____.

Firma del Autor

(Anddy Castellanos Huerta)

Firma del Tutor

(Luis G. Viciado Carabaloso)

DATOS DE CONTACTO

Lic. Luis Gabriel Viciado Carabaloso

Graduado de Licenciatura en Educación en la especialidad de Física en el Instituto Superior Pedagógico de la Ciudad de Camaguey. Realizó estudios de Postgrado en la Escuela de Física de la Universidad de La Habana, en la Universidad de Oriente y en Instituto Superior Politécnico José Antonio Echeverría. Profesor Auxiliar desde 1998. Impartió clases de Matemática Superior, Física y Computación en el Instituto Superior Agrícola de Ciego de Ávila. Graduado en Inglés, Francés, Ruso y Portugués. Ha presentado trabajos en eventos nacionales e internacionales relativos al uso de radiaciones ionizantes con fines mutagénicos, Dosimetría de Altas Dosis y uso de nuevas tecnologías con fines educativos. Realizó estudios de postgrado en el Centro de Referencia para la Educación de Avanzada de la CUJAE, sobre el empleo de laboratorios virtuales en las carreras de ingeniería. Imparte docencia en el Departamento de la Especialidad en las disciplinas de Programación y Sistemas Digitales.

Correo electrónico: viciado@uci.cu

Teléfono: 835-8757

"La práctica debe siempre ser edificada sobre la buena teoría."

Leonardo Da Vinci

AGRADECIMIENTOS

A todas esas personas que han hecho posible que hoy esté aquí.

A mis abuelos, por ser mis ejemplo a seguir.

A mis tías y a mi tío, por estar siempre ahí cuando lo he necesitado.

A mi querida hermana Niurvis, por quererme como me quiere.

A mi papá, por su apoyo y cariño.

A todos mis amigos, con los que he compartido durante estos cinco años, a los que por una razón u otra hoy no pueden estar aquí, especialmente a mi amigo Madver.

DEDICATORIA

DEDICATORIA

A mis Abuelos.

RESUMEN

El surgimiento y desarrollo de las Tecnologías de la Información y las Comunicaciones ha revolucionado el proceso enseñanza-aprendizaje, permitiendo que las personas intercambien conocimientos y aprendan online desde cualquier parte del mundo, lo que ha propiciado la aparición de novedosas plataformas de aprendizaje, capaces de producir cursos basados en Internet. Además, han surgido numerosas aplicaciones Web que se usan juntas o de forma independiente a dichas plataformas. Una de estas aplicaciones la constituye el software LAMS, con el cual un profesor puede realizar una secuencia de aprendizaje que posteriormente el alumno debe seguir.

En dicha secuencia, utilizando dicho software, el profesor puede añadir varios recursos que luego serán mostrados al alumno con el objetivo de crear nuevos conocimientos o consolidar los que ya el alumno tiene. Pero esta visualización se realiza de forma poca interactiva, tanto en el momento que el profesor agrega el recurso, como cuando este es visualizado.

En este trabajo se propone una herramienta que permita que el profesor pueda ofrecer diferentes recursos de una forma rápida e interactiva, que es sin duda una de las principales características que debe poseer una aplicación dedicada al proceso de enseñanza- aprendizaje.

Palabras claves:

Proceso de enseñanza-aprendizaje, E-learning, Laboratorios virtuales, Interactividad, Recursos, Visualización interactiva.

INDICE

AGRADECIMIENTOS.....	IV
DEDICATORIA.....	V
RESUMEN.....	VI
INTRODUCCION.....	1
CAPÍTULO 1 FUNDAMENTACIÓN TEÓRICA.....	5
Conceptos fundamentales.....	5
E- learning.....	5
Comunidades de aprendizaje.....	6
Laboratorios virtuales.....	7
Recurso.....	8
Interactividad.....	9
Herramientas y tecnologías para el desarrollo de la aplicación.....	11
Lenguajes.....	11
HTML.....	12
Java Script.....	12
CSS.....	13
ASP.....	14
PHP.....	14
Lenguaje propuesto: PHP.....	15
Sistemas gestores de base de datos.....	16
Oracle.....	17
PostGreSQL.....	17
SQL Server 2000.....	18
MySQL.....	18
Gestor de Base de Datos elegido: MySQL.....	19
Apache.....	19
AJAX.....	20
LAMS (Learning Activity Management System).....	22
Metodología a utilizar.....	23
CAPÍTULO 2. DESCRIPCIÓN DE LA SOLUCIÓN PROPUESTA.....	26
Estado actual del negocio.....	26
Solución propuesta.....	27
Modelo del dominio.....	28
Diagrama del Modelo del dominio.....	29
Requisitos funcionales.....	30

Requisitos no funcionales.....	31
Descripción del Sistema propuesto.....	33
Modelo del sistema.....	34
Descripción de los casos de uso del sistema.....	34
Diagrama de Casos de Uso del sistema.....	35
Casos de uso expandidos.....	36
CAPÍTULO 3. CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA.....	50
Patrones de Diseño.....	50
Diagrama de Clases del Diseño.....	52
Diseño de la Base de Datos.....	57
Modelo de datos.....	58
Descripción de las Tablas.....	60
Principios de Diseño.....	61
Estándares en la interfaz de la aplicación.....	61
Formatos de reportes.....	63
Tratamiento de excepciones.....	65
Concepción general de la ayuda.....	67
Estándar de Codificación.....	68
Modelo de despliegue.....	68
Descripción del modelo de prueba.....	69
Casos de prueba.....	70
CONCLUSIONES GENERALES.....	75
RECOMENDACIONES.....	76
REFERENCIAS BIBLIOGRÁFICAS.....	77
BIBLIOGRAFIA.....	79
ANEXOS.....	80
GLOSARIO DE TERMINOS.....	87

INTRODUCCION

En la actualidad, cada vez hay más Universidades en el mundo para los que la tecnología se ha convertido en una herramienta de incalculable valor a la hora de elaborar y utilizar correctamente una aplicación informática que facilite el proceso de enseñanza-aprendizaje (accesible tanto para profesores como para alumnos) y que ofrezca la potencia y la flexibilidad suficientes como para ajustar el plan de estudios a las necesidades de cada persona. Debido a esto, en el mundo se está promoviendo el desarrollo de aplicaciones educativas efectivas; para que, a través de una computadora, los profesores puedan brindarles a los alumnos todo el material de apoyo que sirve para poder crear y consolidar los conocimientos y que todos los alumnos puedan compartir sus criterio y sus puntos de vista en cualquier momento y lugar. Una aplicación bien hecha puede transformar la forma de enseñar y aprender, ya que motiva tanto a profesores como a alumnos, y mejora los resultados académicos.

Situación Problémica y problema a resolver.

Estas aplicaciones que se crean con fines educativos y que se utilizan actualmente en el mundo se basan en diferentes principios pedagógicos, como por ejemplo el **constructivismo** que plantea que las personas aprenden en el momento en que interpretan aquello que les llega de nuevo, es decir que cuando la persona recibe nueva información, la contrasta con lo que ya sabe y entonces es cuando construye el conocimiento. Otro de los principios que sirven de base pedagógica a estas aplicaciones es el **construccionismo** que plantea que si además de construir una idea en nuestra mente la pensamos y desarrollamos suficiente para poderla transmitir a otras personas, explicarla o exponerla, el conocimiento que hemos adquirido será más consistente.

Estos dos enfoques de la pedagogía se basan en ideas que defienden la idea de que todo individuo cuando percibe algo nuevo lo compara con las cosas que tiene en su mente y así logra formar nuevos conocimientos, por lo el profesor se limita a facilitar a los alumnos los materiales necesarios para que ellos mismos construyan sus conocimientos.

Por el contrario, en nuestra sociedad la pedagogía se basa en un enfoque denominado **Histórico-Cultural**, que plantea que la educación y, en particular, la escuela y el profesor, tienen un papel rector en el sistema de influencias sociales que estimulan la formación y desarrollo de la personalidad del

estudiante, por tanto, la función del profesor no puede ser la de facilitar la libre expresión de las potencialidades “que trae” el estudiante, sino la de diseñar situaciones de aprendizaje que planteen retos al estudiante para que en el proceso de solución de las tareas de aprendizaje, en condiciones de interacción social, puedan formar y desarrollar las potencialidades que le permitan alcanzar la condición de sujetos de su actuación. (MAURA, 1999)

Es decir que el profesor no solo se debe limitar a brindarle a todos los estudiantes el material para que ellos solos construyan su conocimiento, sino que debe ser capaz de conocer a cada uno de sus alumnos en forma particular para así poder asignarle tareas para el cumplimiento de un objetivo específico y personalizarla para cada uno de sus estudiantes de forma particular y que no sea realizada de la misma forma y ni con los mismos pasos por los estudiantes que componen su grupo.

Aunque difieren en el papel que el profesor debe asumir a la hora de realizar su principal labor, que es enseñar, en lo que coinciden estos tres enfoques de la pedagogía, y la mayoría de los existentes, es en la necesidad de que el profesor pueda brindarle a sus alumnos materiales de apoyo para que ellos construyan nuevos conocimientos. Estos materiales, a la hora de verlos como parte una práctica de laboratorio virtual, pueden ser muy diversos y van desde un documento para ser leído e interpretado hasta una película que sirva de ejemplo para lo que el profesor quiera transmitirle al alumno.

Esto de poder facilitar recursos informáticos educativos es muy importante a la hora de que el profesor pueda diseñar prácticas de laboratorio, proponiendo distintas variantes o secuencias, para el cumplimiento de los objetivos. Es decir es muy importante a la hora de que el profesor pueda diseñar una clase sobre Web.

Se llega a la conclusión que **en el desarrollo de prácticas de laboratorios virtuales sobre Web, la visualización interactiva de recursos, tales como páginas Web, videos, imágenes y texto, es necesaria para facilitar a los sujetos que interactúan en el proceso de aprendizaje el cumplimiento de los objetivos propuestos, por lo que es imprescindible garantizar la tecnología para esta finalidad.**

Este trabajo surge como necesidad de dar solución a las situaciones antes expuestas; por lo que nuestro **problema** consiste en **¿Cómo lograr tecnológicamente una visualización interactiva de los recursos necesarios en una práctica de laboratorio virtual remota sobre Web?**

El **objeto de estudio** de este trabajo lo constituyen las tecnologías y herramientas de comunicación para el desarrollo de prácticas de laboratorios virtuales. De ello se deriva que el **campo de acción** que abarca sea el proceso de la visualización interactiva de recursos en prácticas de laboratorios virtuales sobre Web.

Por lo que el **objetivo general** será: Diseñar e implementar una primera versión de un módulo de visualización interactiva para el proceso de una práctica de laboratorio virtual sobre Web.

Para cumplir con el objetivo planteado se proponen las siguientes **tareas**:

- Estudiar todo lo referente a visualización de materiales en prácticas de laboratorios virtuales.
- Estudiar los tipos de recursos que más se visualizan en las distintas plataformas existentes.
- Analizar el estado del arte de las interacciones durante la visualización, en comunidades de aprendizajes.
- Estudiar y explotar las posibilidades que ofrecen las librerías Rico y Prototype para lograr la interactividad requerida en una aplicación dedicada a una practica de laboratorio virtual.
- Definir el conjunto de requisitos de interacciones durante la visualización para la versión del módulo.

El presente documento se estructura en tres capítulos:

- En el **Capítulo 1** se describe detalladamente los conceptos fundamentales que ayudan a entender mejor el entorno en el cual se desarrollo este trabajo, además se explicará en qué consisten los laboratorios virtuales y la visualización de recursos en una práctica de laboratorio virtual, destacando la importancia de la interactividad en ella. También se mencionan las herramientas utilizadas en la elaboración de la aplicación Web comparándolas con otras existentes y justificando su utilización. Se dicen las posibilidades que brindan la tecnología AJAX para lograr dicha interactividad. Por último se describe la metodología RUP que es utilizada en el desarrollo de la aplicación.
- En el **Capítulo 2** se hace la descripción de la solución propuesta, describiendo el estado actual del negocio. Luego se realiza el modelo del dominio y se seleccionan y describen los requisitos

funcionales y no funcionales, haciendo un Diagrama de Casos de Usos del Sistema y la descripción detallada de estos CU.

- En el **Capítulo 3** se expone la construcción de la solución propuesta, a través de los flujos de trabajo de diseño e implementación. Se realizan los diagramas de clases del diseño y los diagramas de secuencia a cada uno de los casos de uso. Luego se eligen las clases persistentes y se realiza el modelo de datos. Además se analizan los principios de diseño utilizados en la realización de la aplicación y por último se le aplican pruebas a los Casos de Uso más significativos.

Capítulo 1

Capítulo 1 Fundamentación Teórica.

Introducción.

El presente capítulo trata de algunos conceptos necesarios para el desarrollo de aplicaciones Web dedicadas a visualizar recursos en un ambiente educativo. Por lo que se hablará de lo que es la educación a distancia y las comunidades virtuales de aprendizaje, los laboratorios virtuales y la visualización interactiva de recursos. Además se describen y se comparan las herramientas y tecnologías que se usan en la actualidad para desarrollar aplicaciones Web, como por ejemplo los lenguajes del lado del cliente y del lado del servidor, los sistemas de gestores de bases de datos y los servidores de bases de datos. También se justifica las distintas herramientas que se eligieron para la elaboración del módulo. Se menciona y describe la metodología que se utiliza.

Conceptos fundamentales.

E- learning.

Es indudable el valor de las nuevas tecnologías de información y comunicación (TIC) en la educación. La inmensa cantidad de información disponible en medios digitales, el alcance y penetración de la televisión educativa, el uso de la computadora para realizar simulaciones y ejercicios interactivos y sobre todo la vastedad e inmediatez de Internet son algunos claros ejemplos de las bondades de la tecnología que hacen posible mejorar el proceso enseñanza-aprendizaje.

La expresión e-learning hace referencia al uso de tecnologías de Internet (e-) junto a una metodología de transmisión de conocimientos y desarrollo de habilidades centradas en el sujeto que realiza un proceso de aprendizaje (learning). El término alude a diferentes tipos de educación a distancia impartida por medio de Internet, desde los cursos de formación continua o formación permanente en el ámbito de la empresa,

hasta las enseñanzas regladas en las universidades y otros centros de educación superior.(PASCUALENA, 2007)

Es decir que el e-learning no es más que una modalidad educativa que permite que los alumnos puedan acceder a cursos, contenidos y actividades a distancia a través de una computadora.

Para la correcta utilización del e-learning se hace imprescindible la creación de aplicaciones educativas para ser utilizadas en el medio digital. Estas aplicaciones deben ser altamente interactivas para aprovechar a plenitud el carácter inmediato de Internet.

El e-learning se desarrolla principalmente en aulas virtuales, caracterizadas por estar en red y a distancia, donde se pueden establecer las comunicaciones en tiempo real con la utilización de las teles o videoconferencias y programas de Chat online, pero también se pueden utilizar la comunicación asincrónica empleando los correos electrónicos, los blogs y los foros.

Este tipo de enseñanza se ha vendido imponiendo debido a una serie de ventajas que tiene sobre la enseñanza tradicional como es el hecho de que se desarrolla en un espacio virtual, cuyo máximo ejemplo lo constituye Internet, además, posibilita del intercambio de información entre personas que se encuentran distantes entre sí.

Una de las principales ventajas de la utilización del e-learning es que se puede lograr una mayor productividad en el proceso de enseñanza-aprendizaje ya que los alumnos obtienen los materiales en su propio escritorio lo que evita la pérdida de tiempo.

Comunidades de aprendizaje

El termino comunidad se ha utilizado en el lenguaje científico y en el lenguaje popular, y se refiere a un grupo de personas con un fin común. Para lograr ese fin, ellos establecen relaciones de comunicación e interactúan entre sí intercambiando sus experiencias.

Debido a los cambios sociales que se han producido en la actualidad se ha dado paso de la sociedad industrial a la sociedad de la información. Por lo que han surgido nuevos tipos de comunidades sin que necesariamente sus miembros tengan que compartir el mismo lugar debido a la utilización de las ventajas que ofrecen las telecomunicaciones, especialmente Internet. Surgiendo así las comunidades virtuales, que

según Howard Rheingold, a quién se le atribuye haber acuñado el término **comunidad virtual**, en su libro, *The Virtual Community*, define las comunidades virtuales como: agregaciones sociales que emergen de la red cuando un número suficiente de personas entablan discusiones públicas durante un tiempo lo suficientemente largo, con suficiente sentimiento humano, para formar redes de relaciones personales en el ciberespacio.(RHEINGOLD, 1993)

La única diferencia que hay entre comunidad y comunidad virtual es que antes el territorio de la comunidad era geográfico y hoy este es virtual, es decir, la nueva comunidad no ocupa un espacio en el mundo físico sino en el ciberespacio. Así, el ser humano traslada a este espacio virtual, sus modos de vida en sociedad, su cultura y configura ese espacio para que responda a sus necesidades.

Una comunidad virtual puede tratarse de un foro donde la gente que participa comparte información debido a que tienen intereses en común y entablan discusiones publicas durante un tiempo lo suficientemente largo como para establecer relaciones interpersonales, también puede referirse a un Chat que plantea temas y objetivos a desarrollar, es decir que los miembros de estas comunidades pueden comunicarse y compartir sus ideas mas allá del lugar físico donde se encuentran.

Pero los cambios que ha introducido la sociedad de la información no sólo afectan a la economía, sino que influyen en las condiciones de vida de todas las personas, afectando también, de manera significativa, la educación. Por lo que han surgido las comunidades virtuales de aprendizaje, cuyos integrantes en sus distintos roles, sean alumnos, tutores, profesores y coordinadores, se conectan a través de Internet para intercambiar contenidos, ver el programa de asignaturas, enviar un correo al profesor, charlar con los compañeros, debatir en un foro o participar en una tutoría.

Laboratorios virtuales.

Una práctica de laboratorio virtual consiste en utilizar las computadoras para realizar experimentos de ciencias en tiempo real y utilizar las redes de computadoras para que los alumnos compartan la adquisición de los datos de dicha práctica. Por lo que la motivación de los laboratorios virtuales surge, básicamente, por la necesidad de optimizar el tiempo que el alumno emplea en la realización de dichas prácticas.

El principal fin de un laboratorio virtual es proporcionar una experiencia similar a la obtenida en un laboratorio de prácticas reales. De hecho, cada laboratorio virtual debería ser diseñado para proporcionar una experiencia similar a la obtenida al trabajar con equipo real y específico. Los laboratorios virtuales son imitaciones digitales de prácticas de laboratorio o de campo, reducidas a la pantalla de la computadora.

Existen tres tipos principales de laboratorios virtuales:

- **Laboratorios Virtuales Software:** Son laboratorios virtuales desarrollados como un programa de software independiente destinado a ejecutarse en la máquina del usuario, y cuyo servicio no requiere de un servidor Web.
- **Laboratorios Virtuales Web:** Este tipo de laboratorios se basa en un software que depende de los recursos de un servidor determinado. Esos recursos pueden ser determinadas bases de datos, software que requiere ejecutarse en su servidor, la exigencia de determinado hardware para ejecutarse.
- **Laboratorios Remotos.** Se trata de laboratorios remotos que permiten operar remotamente cierto equipamiento, bien sea didáctico como maquetas específicas, o industrial, además de poder ofrecer capacidades de laboratorio virtual. (HERÍAS, 2003)

Utilizando los laboratorios virtuales Web los profesores pueden crear clases virtuales en sus computadoras las cuales trabajarían en ese caso como servidores Web. Estas clases pueden contener situaciones que sean una simulación de un problema real con el objetivo que el estudiante obtenga en su computadora los mismos resultados que obtendría utilizando materiales reales. Además, los profesores les pueden proporcionar a los alumnos distintos recursos de apoyo que sirvan para que los alumnos consoliden sus conocimientos sobre un tema específico.

Recurso.

Existe una gran variedad de medios y recursos que se utilizan en las prácticas virtuales educativas con el objetivo de mejorar y consolidar el proceso de enseñanza-aprendizaje.

Los recursos más usados con este fin son los recursos multimedia que no son más que “documentos informáticos que combinan textos, imágenes y sonido, es decir, a recursos que nos ofrecen la posibilidad

de obtener información de diversos medios de comunicación interconectados y que pueden ser controlados por el alumno mediante el ordenador. El entorno multimedia permite al usuario interactuar activamente con la información y reestructurarla de forma personal.”

El recurso más antiguo y utilizado en la educación es el **documento de texto**, debido a que puede contener todo lo que el alumno debe aprender de una forma clara y directa en los libros, además, es una fuente que hace que se desarrolle la imaginación del lector. Estos libros se fueron transformando hasta llegar a surgir el texto electrónico. Estos textos electrónicos se han convertido en grandes bancos de información que pueden ser creados y transmitidos muy rápidamente utilizando las redes digitales.

De este recurso se deriva otro que viene a ser el recurso más utilizado en Internet, **las páginas Web**. Estas páginas no son más que la combinación de texto e imágenes, principalmente. Esta suma hace lo que se conoce como el hipertexto, el cual puede contener todo tipo de información que puede ser utilizada por los alumnos.

Uno de los recursos más antiguos utilizados en la labor de enseñar es la **imagen**. Este recurso era usado desde la era de piedra para transmitir el conocimiento. En la actualidad las imágenes se usan con fines didácticos desde la educación primaria hasta la superior, aprovechando su cualidad como portadora de información visual y como una forma de comunicación.

El **video** es el recurso informático más completo empleado en la educación ya que no es más que la combinación de imágenes en movimiento y sonido, por lo que logra transmitir la realidad mejor que cualquier otro, permitiendo que el mensaje llegue a quien va dirigido.

La combinación de estos recursos multimedia son los que hacen posibles que se presente suficiente información como para trabajar con ideas y conceptos, que al utilizarlos ayudan a los estudiantes a pensar y a aprender más.

Interactividad

Uno de los aspectos relevantes a la hora de utilizar las TIC como medio de enseñanza es el concepto de interactividad, característica que cada día es más necesaria en los distintos software educativos. “La interactividad es el soporte de un modelo general de enseñanza que contempla a los estudiantes como

participantes activos del proceso aprendizaje, no como receptores pasivos de información o conocimiento.” (M^a BEGOÑA ALFAGEME GONZÁLEZ, 2006)

La interactividad puede estar presente en un todo software, sea educativo o no, de muchas formas, desde la sencilla acción de pulsar una tecla o un botón y obtener una respuesta, donde el programa atiende la petición del usuario y produce una respuesta inmediata hasta las acciones que puede realizar un usuario en un entorno complejo, ocurriendo una serie de condiciones que van activando una próxima posible acción que se podrá realizar. Este tipo de sofisticada interactividad hace que el usuario tenga que esforzarse más para poder lograr su objetivo final y poder hacer encajar más todas las piezas.

En el caso específico de los software educativos se debe encontrar el grado de interactividad que corresponda a los objetivos. No obstante a esto, es necesario siempre lograr tres tipos de interactividades, las cuales se complementan entre sí:

- La interactividad entre el contenido y el alumno lo que hace posible que la información deseada llegue de forma correcta a los alumnos.
- La interactividad entre el alumno y el profesor para lograr la motivación de los alumnos y el apoyo de los profesores. Debe de existir una retroalimentación y la comunicación debe fluir en ambas direcciones.
- Motivar la interactividad entre los propios alumnos, permitiendo el trabajo colectivo y colaborativo para lograr el objetivo final, que es la comprensión del contenido.

La calidad de una aplicación Web, dedicada a la educación a distancia, radica fundamentalmente en la interacción entre los alumnos y la aplicación, motivándolos a seguir usándola y permitiendo que el mensaje que desea transmitir el profesor sea recibido.

El tipo de interactividad que predomina en la educación a distancia suele ser la virtual, pero si se utilizan a plenitud los avances tecnológicos con los que contamos, se puede desarrollar una interactividad reales a través de los medios.

Herramientas y tecnologías para el desarrollo de la aplicación.

Lenguajes.

Los lenguajes de programación son los que hacen posible que los mensajes del programador sean interpretados por los microprocesadores de las computadoras permitiendo la comunicación entre ambos. Los lenguajes permiten escribir las operaciones necesarias para resolver un problema, es decir poner el algoritmo y también se encargan de traducir este algoritmo a lenguaje máquina para que pueda ser ejecutado, brindando un resultado final. Los lenguajes de programación son herramientas que nos permiten crear programas y software. En la actualidad hay muchos tipos de lenguajes de programación, cada uno de ellos con su propia gramática, su terminología especial y una sintaxis particular.

Entre estos programas se destacan los dedicados a la programación Web. Estos lenguajes se utilizan para crear páginas y sitios Web que serán publicadas y estarán a disposición de los usuarios que navegan en una red informática ya sea Internet o una intranet.

Rodeando a estos lenguajes existen conceptos que son necesarios tener en cuenta para poder explotar todas las posibilidades que ellos brindan como son el de URL que se refiere a la dirección donde se encuentra una página o un servicio Web. El protocolo HTTP (Hypertext Transfer Protocol) encargado de llevar la comunicación entre los servidores y los cliente Web.

Con el comienzo de Internet se comenzaron a desarrollar aplicaciones concretas para este medio, por lo que el estudio de los lenguajes de programación Web se volvió muy importante para poder conocer a fondo las potencialidades y particularidades de los distintos lenguajes que permiten la creación de páginas y sitios Web.

Los lenguajes para la programación Web se clasifican en dos partes: los lenguajes del lado del Servidor y los lenguajes del lado del Cliente.

Entre los lenguajes del lado del servidor podemos encontrar entre los más sobresalientes por el auge que estos han tenido, algunos como PERL, ASP, PHP. Estos se caracterizan por desarrollar la lógica de negocio dentro del Servidor, además de ser los encargados del acceso a Bases de Datos y del tratamiento de la Información. Del lado del cliente se encuentran principalmente el JavaScript y el Visual Basic Script, que son los encargados de aportar dinamismo a la aplicación en los navegadores.

HTML.

El HTML (Hyper Text Markup Language o Lenguaje de Marcación de Hipertexto) es el lenguaje de marcas de texto utilizado normalmente para la creación de páginas Web.

EL HTML es un lenguaje muy sencillo que permite describir hipertexto, es decir, texto presentado de forma estructurada y agradable, con enlaces o hipervínculos que permitan la navegación a otros documentos o fuentes de información relacionadas, y con inserciones multimedia (gráficos, sonido, videos). La descripción se basa en especificar en el texto la estructura lógica del contenido (títulos, párrafos de texto normal, enumeraciones, definiciones, citas), así como los diferentes efectos que se quieren dar (especificar los lugares del documento donde se debe poner cursiva, negrita, o un gráfico determinado), creando las conocidas paginas Web, las cuales deben guardarse con los formatos .htm o .html para luego puedan ser visualizadas y presentadas finalmente utilizando algún programa especializado (como Firefox Mozilla, o Internet Explore).

Java Script.

El JavaScript es un lenguaje de programación que se utiliza dentro del HTML. “Se trata de un lenguaje de tipo script compacto, basado en objetos y guiado por eventos diseñado específicamente para el desarrollo de aplicaciones cliente-servidor dentro del ámbito de Internet. Los programas JavaScript van incrustados en los documentos HTML, y se encargan de realizar acciones en el cliente, como pueden ser pedir datos, confirmaciones, mostrar mensajes, crear animaciones, comprobar campos.” (GRACIA., 2006)

El JavaScript se utiliza para crear pequeños programitas encargados de realizar acciones dentro del ámbito de una página Web. Con Javascript podemos crear efectos especiales en las páginas y definir interactividades con el usuario. Lo interpreta el navegador y produce alguna acción determinada en la página Web donde está insertado. Es un lenguaje débilmente tipado, basado en objetos y guiado por eventos, logrando con esto el dinamismo de las páginas que incluyan este tipo de código, útil para el desarrollo de aplicaciones cliente-servidor dentro del ámbito de Internet.

JavaScript es utilizado cuando un programador quiere lograr mejorar sus páginas para lograr un mayor dinamismo e interactividad con el usuario, además se puede usar para realizar validaciones del lado del cliente como por ejemplo no permitir que un campo de un formulario sea enviado en blanco al servidor. Es

un lenguaje bastante sencillo que realiza las cosas con rapidez que puede ser utilizado por personas que no tengan una experiencia previa en otro lenguaje de programación.

El lenguaje JavaScript es de código abierto, por lo que cualquier persona puede utilizarlos sin comprar ninguna licencia.

CSS

Las hojas de estilo en cascada (CSS) constituyen un lenguaje usado para definir la presentación de una página Web hecha en HTML. Las CSS surgen debido a la necesidad de separar lo más posible el contenido de una página de la presentación. Esto posibilita que las páginas solo tengan elementos que hacen referencia a su estructura y contenido, ya que todas las características de la apariencia se definen separadamente, en la cabecera de la página o en un archivo aparte. Utilizando CSS el documento HTML en se vuelve más claro de entender y se consigue reducir considerablemente su tamaño, además, se pueda cambiar la forma en que el documento es visualizada sin tener que cambiar nada en el contenido, ni las etiquetas HTML, solo hay que cambiar las definiciones del estilo.

La forma ideal de la utilización del CSS es utilizándolo en un archivo separado, .css, con lo que se hace posible que un mismo archivo CSS puede darle estilo a ilimitado numero de páginas, facilitando que todas ellas tengan el mismo aspecto, logrando así un mayor control del autor sobre sus sitios Web.

Los cambios que se hacen en las hojas de estilo se reflejan instantáneamente en la apariencia de las páginas HTML que la están referenciando. Ellas afectan aspectos tales como el fondo, el color, la fuente, el margen, el borde y el espacio entre letras.

Las reglas del CSS se escriben utilizando términos habituales de diseño, similar al lenguaje cotidiano lo que hace que su aprendizaje sea rápido y sencillo.

ASP.

Según la propia traducción de la definición hecha por *Microsoft* en relación al lenguaje ASP: “Las Active Server Pages son un ambiente de aplicación abierto y gratuito en el que se puede combinar código HTML, scripts y componentes ActiveX del servidor para crear soluciones dinámicas y poderosas para el Web”. (S., 2007)

ASP es una característica de Microsoft Internet Information Server. Debido a que los scripts que corren en servidor son construidos en una página regular de HTML, este puede ser servido en casi cualquier navegador. Un archivo ASP puede ser creado incluyendo un script escrito en JScript dentro de un archivo HTML.

Las páginas hechas en este lenguaje tienen como extensión .asp y comienzan a ejecutarse cuando un usuario solicita una página Web a través del explorador. El servidor Web llama a ASP, y lee el archivo solicitado, ejecuta las secuencias de comandos que encuentre y envía los resultados al explorador del cliente. Las secuencias de comandos se ejecutan en el servidor y quedan ocultas a los usuarios, estos solo reciben el resultado de la ejecución en formato HTML.

La ventaja de la tecnología ASP consiste en el hecho de que facilita la generación del código HTML, ya que éste puede ser escrito directamente y no desde las instrucciones de escritura de un programa en otro lenguaje. Esto resulta de gran ayuda para los diseñadores de páginas Web, quienes no necesariamente son programadores y a menudo no conocen otros lenguajes de programación además del HTML.

La desventaja del uso de ASP es que sólo funcionan para servidores Web determinados. Originalmente esta tecnología fue creada por Microsoft para el Microsoft Internet Information Server.

PHP.

Con la explosión de las nuevas tecnologías para la Web, la creación de sitios usando sólo un editor de archivos HTML es cada día menos probable. Un sitio moderno, actualizado y atractivo tiene que ser dinámico si quiere prevalecer y conquistar la red. Las soluciones disponibles para conseguirlo son muchas, pero se destaca PHP.

PHP es un lenguaje del lado del servidor, gratuito e independiente de plataforma, que funciona contenido dentro del código HTML de una página, dándole mayor dinamismo a la misma. Su primer uso fue el de mantener un control sobre quien visitaba un currículum en la página Web de su creador.

Su sintaxis es heredada de C/Java y posee gran cantidad de funciones que permiten realizar todas las acciones que soporta el PHP y su objetivo es permitir la generación dinámica de páginas Web y que los usuarios de estas páginas vean la información modificada.

La interpretación y la ejecución del lenguaje ocurren en el servidor por lo que no es necesario que el navegador lo soporte, es independiente del navegador. Con él se puede realizar el procesamiento de información de formularios, cuando el cliente hace una petición al servidor, el servidor ejecuta el intérprete de PHP que es el encargado de procesar el script para generar el contenido de manera dinámica, modificando el contenido a enviar, y regresa el resultado al servidor, el cual se encarga de regresárselo al cliente.

PHP es un lenguaje orientado a objeto, por lo que se puede utilizar clases y herencia. Permite la conexión a diferentes tipos de servidores de bases de datos tales como MySQL, Postgres, Oracle y Microsoft SQL Server.

Lenguaje propuesto: PHP

PHP cada día aumenta su popularidad y presencia en Internet, al ser un software libre se puede acceder fácilmente a su código fuente permitiendo que los desarrolladores, que son a su vez programadores Web, introduzcan nuevas funciones en las versiones mejoradas del lenguaje.

PHP es absolutamente gratis y no posee precios ocultos por lo que cuando se adquiere trae consigo un gran número de bibliotecas o, en caso de no traerlas, pueden ser adquiridas gratis en Internet. Existe gran documentación del lenguaje, ejemplos y manuales.

PHP fue diseñado inicialmente para entornos UNIX, pero es compatible con Windows, convirtiéndolo un lenguaje multiplataforma. Además, es un lenguaje que tiene gran capacidad para crear, leer y manipular datos provenientes de formularios que son ingresados por los propios usuarios,

Es un lenguaje modular y orientado a objetos que es bastante sencillo de aprender ya que su estructura base es tomada del C, lo que permite a programadores su fácil utilización y a aquellos que se inician un aprendizaje rápido.

Tiene una gran capacidad para conectarse con la mayoría de los manejadores de base de datos que se utilizan en la actualidad, pero se destaca su conectividad con MySQL y existe una perfecta integración entre Apache-PHP-MySQL.

Sistemas gestores de base de datos.

Antes de la existencia de las base de datos, para almacenar datos se utilizaban archivos secuenciales. Cuando se quería acceder a un dato, que estaba en una posición determinada, se tenía que recorrer el archivo entero, por lo que existía gran pérdida de tiempo. Para solucionar ese problema aparecieron los archivos indexados, donde se puede acceder directamente a la posición que se desee. Pero estos nuevos archivos no permitían que proteger las operaciones de un usuario con respecto a otro, además, los programas y datos que se debían almacenar eran cada vez más complejos. Por lo que surgen las bases de datos jerárquicas, donde “La información afín está organizada en anillos (listas cíclicas), por ejemplo empresas, ciudades, trabajos... De cada anillo se pasa otro anillo de conceptos subordinados”. (IGNACIO COUPEAU, 2006)

En las base de datos jerárquicas el acceso a los datos son unidireccionales, lo que hace más complicado hacer el camino inverso un busca de nuevos datos, haciendo mayor el tiempo de cálculo.

Para solucionar este problema surgieron las bases de datos relacionales, las cuales tienen como principales propiedades la atomicidad que garantiza que cada petición del usuario se realice completa o no se realice ya que es tratada de forma atómica (que no permite división). Otra de las características es la consistencia que permite que se conserve la integridad y coherencia de los datos ya que se evita la ejecución de las peticiones de los usuarios que no cumpla las restricciones definidas dentro de la base de datos. Además, una vez que se ha completado una transacción los resultados de la misma se hacen permanentes y sobreviven a posibles caídas del sistema.

En los últimos años, el software de bases de datos ha experimentado un auge extraordinario, a raíz de la progresiva informatización de casi la totalidad de las empresas de hoy día. No es extraño pues, que

existan multitud de gestores de bases de datos, programas que permiten manejar la información existente en una base de datos de un modo sencillo. Algunos de los gestores de base de datos más utilizados son Oracle, Microsoft SQL Server, que son soluciones comerciales y PostgreSQL y MySQL que son software libre.

Oracle.

Oracle es un gestor de base de datos que proporciona la gestión de datos de forma fiable y segura, es básicamente una herramienta cliente/servidor para la gestión de base de datos.

Oracle Database ofrece soporte para muchos estándares del sector a través de las principales arquitecturas de sistemas operativos y de hardware disponibles en la actualidad desde Linux hasta Windows, Unix y OS/390. La portabilidad superior de Oracle Database proporciona mayor capacidad para cambiar, con facilidad, la infraestructura preferida de hardware y del sistema operativo.

Se ha diseñado para hacer frente a las exigencias de rendimiento, fiabilidad y escalabilidad que son necesarias para trabajar en la red.

La principal desventaja que tiene, con respecto a otros sistemas gestores de base de datos, es que, a pesar de ser un producto vendido a nivel mundial, la gran potencia que tiene y su elevado precio hace que solo se vea en empresas muy grandes y multinacionales, por norma general.

PostgreSQL.

PostgreSQL es un sistema de gestión de bases de datos objeto-relacional. Se basa en el proyecto POSTGRES, es una derivación libre. Fue el pionero en muchos de los conceptos existentes en el sistema objeto-relacional actual, incluido, más tarde en otros sistemas de gestión comerciales. Aunque no es un sistema de gestión de bases de datos puede incluir características de la orientación a objetos, como puede ser la herencia, tipos de datos, funciones, restricciones, disparadores, reglas e integridad transaccional.

Además de los tipos de datos base, PostgreSQL soporta datos de tipo fecha, monetarios, elementos gráficos, datos sobre redes y cadenas de bits. También permite la creación de tipos propios.

Permite la gestión de diferentes usuarios y de los permisos asignados a cada uno de ellos.

SQL Server 2000.

SQL Server 2000 es un potente motor de bases de datos de alto rendimiento capaz de soportar millones de registros por tabla con un interface intuitivo y con herramientas de desarrollo integradas como Visual Studio 6.0 o .NET, además incorpora un modelo de objetos totalmente programable (SQL-DMO) con el que podemos desarrollar cualquier aplicación que manipule componentes de SQL Server, es decir, hacer aplicación para crear bases de datos, tablas, DTS, backups, etc., todo lo que se puede hacer desde el administrador del SQL Server y podemos hacerlo no solo en Visual C++ sino también en Visual Basic, ASP y por supuesto en .NET. (SOLANO, 2002)

Microsoft SQL Server 2000 es un Servidor de Base de Datos y herramienta de Análisis de la información. Proporciona seguridad, fiabilidad y escalabilidad para poner en marcha cualquier aplicación en un tiempo pequeño, destacando sus sencillas tareas de administración y su capacidad de analizar la información.

SQL Server permite la creación de procedimientos almacenados, los cuales consisten en instrucciones que se almacenan dentro de una base de datos de SQL Server, realizados en lenguaje SQL. Son procedimientos que se guardan semicompilados en el servidor y que pueden ser invocados desde el cliente. Se ejecutan más rápido que instrucciones SQL independientes.

Sus principales desventajas son que tiene una licencia con costos muy altos y que solo funciona en plataforma Windows.

MySQL.

MySQL es el servidor de bases de datos relacionales más popular, desarrollado y proporcionado por MySQL AB. Una de las razones que justifican el rápido crecimiento de popularidad de MySQL, es que se trata de un producto Open Source, y por lo tanto, va de la mano con este movimiento.

MySQL surgió como una necesidad de un grupo de personas sobre un gestor de bases de datos rápido, por lo que sus desarrolladores fueron implementando únicamente lo que precisaban, intentando hacerlo funcionar de forma óptima.

Cualquiera puede descargar el software de MySQL de Internet y usarlo sin pagar por ello lo que lo hace muy útil para productores de aplicación de que requieran almacenamiento y gestión de datos y no cuenten con financiamiento de proyecto. Inclusive, cualquiera que lo necesite puede estudiar el código fuente y

cambiarlo de acuerdo a sus necesidades. MySQL usa la licencia GPL (Licencia Pública General GNU), para definir qué es lo que se puede y no se puede hacer con el software para diferentes situaciones. Sin embargo, si uno está incómodo con la licencia GPL o tiene la necesidad de incorporar código de MySQL en una aplicación comercial es posible comprar una versión de MySQL con una licencia comercial.

Gestor de Base de Datos elegido: MySQL

Para el desarrollo de este trabajo se decidió la utilización de MySQL debido a que es muy rápido al conectarse con el servidor, es seguro y fácil de usar ya que fue desarrollado originalmente para manejar grandes bases de datos mucho más rápido que las soluciones existentes y ha estado siendo usado exitosamente en ambientes de producción sumamente exigentes por varios años. Aunque se encuentra en desarrollo constante, el servidor MySQL ofrece hoy un conjunto rico y útil de funciones. Su conectividad, velocidad, y seguridad hacen de MySQL un servidor bastante apropiado para acceder a bases de datos en Internet.

MySQL permite la recuperación de errores. No pierde información ni corrompe los datos aunque se bloquee. Los registros no tienen límite de tamaño. Tiene un eficiente control de acceso, en el sentido de qué usuarios tienen acceso a qué tablas y con qué permisos.

MySQL tiene un bajo consumo, lo que lo hacen apto para ser ejecutado en una máquina con escasos recursos sin ningún problema.

Apache.

El servidor Web es el programa encargado de gestionar las peticiones de los usuarios que visitan un sitio Web. Este programa se encarga de mostrar las páginas cuando un visitante realiza una petición. Teniendo en cuenta que una gran por ciento de las operaciones que realiza un usuario en el servidor es el de solicitar ver una página Web, la elección de un correcto y eficiente servidor Web es muy importante para el buen funcionamiento general del servido.

La elección del servidor para este trabajo es el Apache. Apache es un servidor Web de código abierto lo que significa que su código fuente está disponible para la comunidad que lo va mejorando constantemente, y cuando aparece un problema de seguridad suele solucionarse en unas pocas horas. Su nombre es un acrónimo de "a patchy server" lo que significa que es un servidor construido con códigos

que ya existían, juntando piezas y parches. Se comenzó a desarrollar en 1995 con el objetivo de mejorar el servidor en la NCSA.

Apache es el servidor Web que más se utiliza en el mundo, debido a que funciona sobre la mayoría de los sistemas operativos y arquitecturas ofreciendo un gran rendimiento en todas. Se puede utilizar lo mismo utilizando tecnología moderna que en sistemas 486 sin ningún problema.

Apache ofrece una gran flexibilidad, lo cual es difícil encontrar en otros servidores Web. Ofrece la posibilidad de utilizarlo como proxy cache o proxy inverso, se puede usar como balanceador de carga o servidor de aplicaciones, aunque para ello debe tener una configuración correcta para cada caso, pero esto se resuelve con la amplia documentación que existe sobre el servidor en Internet.

AJAX

AJAX es el acrónimo de **A**synchronous **J**avaScript **A**nd **X**ML, es una técnica de desarrollo Web para crear aplicaciones interactivas.

AJAX no es una tecnología. Es realmente muchas tecnologías, cada una floreciendo por su propio mérito, uniéndose en poderosas nuevas formas. AJAX incorpora:

- presentación basada en estándares usando XHTML y CSS;
- exhibición e interacción dinámicas usando el Document Object Model ;
- Intercambio y manipulación de datos usando XML and XSLT ;
- Recuperación de datos asincrónica usando XMLHttpRequest ;
- y JavaScript poniendo todo junto. (GARRETT, 2006)

En el modelo clásico de las aplicaciones Web, cuando no usamos AJAX, las acciones del usuario disparan directamente un requerimiento HTTP al servidor, el cual efectúa uno o varios procesos y le devuelve una página HTML al cliente, es decir, mientras el servidor está procesando la respuesta de una petición del usuario, el usuario está esperando por su respuesta y no puede ejecutar ninguna otra acción. Cuando usamos AJAX introducimos un intermediario entre el usuario y el servidor, eliminando la espera del usuario ya que este motor es el responsable de la comunicación directa con el servidor y cuando estos están intercambiando información el usuario pueda seguir efectuando otras peticiones, permitiendo que la

interacción del usuario con la aplicación suceda independiente de la comunicación con el servidor, o sea asincrónicamente.

Cualquier respuesta a una acción del usuario que no requiera un viaje de vuelta al servidor (como una simple validación de datos, edición de datos en memoria, incluso algo de navegación) es manejada por el motor AJAX. Si el motor necesita algo del servidor para responder (sea enviando datos para procesar, cargar código adicional, o recuperando nuevos datos) hace esos pedidos asincrónicamente, usualmente usando XML, sin frenar la interacción del usuario con la aplicación.

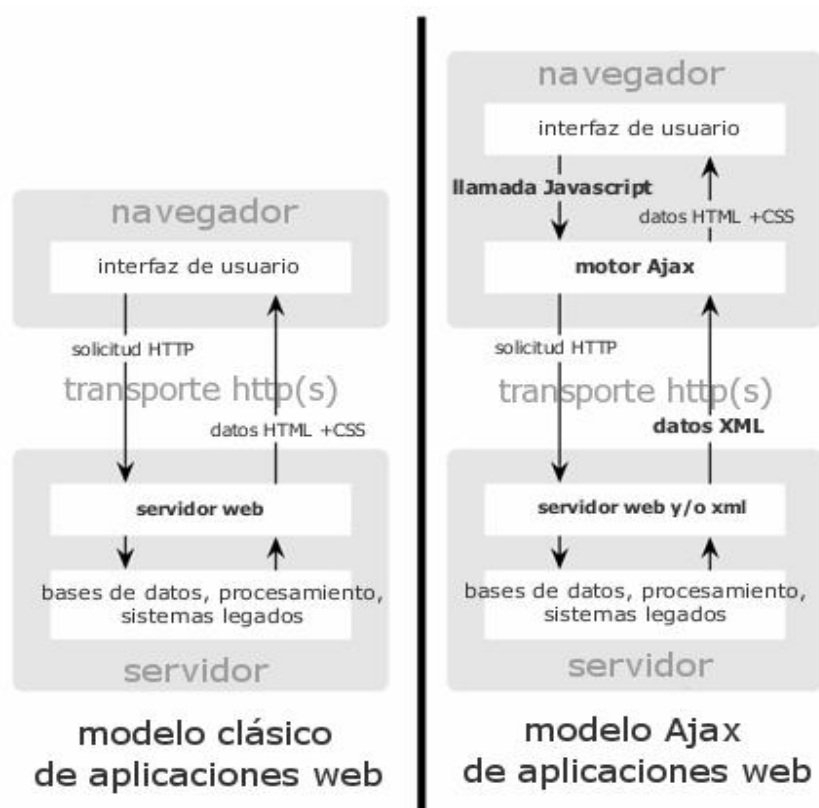


Figura 1.1: Comparación entre el modelo clásico de aplicación Web y el modelo AJAX de aplicaciones Web.

LAMS (Learning Activity Management System)

Este software denominado LAMS o Sistema de Control de Actividades de Aprendizaje es una herramienta para diseñar, gestionar y distribuir en línea actividades de aprendizaje colaborativas, actualmente integrado a plataformas educativas para facilitar el manejo de las actividades con los estudiantes, entre estas a Moodle. Este ofrece a los profesores en un entorno visual intuitivo, un grupo de componentes que permite modelar una o varias secuencias lógicas de actividades. Estas actividades permiten desplegar una gama de tareas individuales, en pequeños grupos de trabajo y en grupos completos en actividades centradas en contenidos y en investigación.

El paradigma que ofrece LAMS es bastante innovador. En vez de centrarse en presentar contenidos a un estudiante solo, el sistema está pensado para que los educadores puedan diseñar actividades de aprendizaje colaborativas dirigidas a todo un grupo. De hecho, no muy diferente de lo que es planificar una lección convencional. Una pantalla de gestión de la actividad muestra el avance de cada estudiante en la secuencia de actividades que constituyen una unidad y proporciona medios para descubrir y analizar su desempeño en el equipo a que pertenece.

El sistema LAMS está enfocado en un aspecto concreto del e-Learning, las secuencias de actividades, y en particular las actividades colaborativas. Existe además en el sistema, un espacio para profesores y diseñadores de cursos que a modo de repositorio puedan compartir las secuencias creadas.

En este sentido LAMS proporciona una variedad de funcionalidades para definir una secuencia de actividades y ponerlas a disposición de los usuarios, ya sean individuales o en grupo. Incluye un entorno visual para el diseño de dichas secuencias, un entorno de monitorización o supervisión, donde los profesores pueden observar la evolución en tiempo real de los alumnos en dicha secuencia, para poder organizar mejor este trabajo también es capaz de crear equipos de estudiantes y de esta forma facilitar la asignación de las actividades.

En la herramienta LAMS se encuentran cuatro espacios bien diferenciados:

1. **Zona de autor**, dedicada a la creación o modificación de secuencias en LAMS. Ofrece una interfaz visual en la que simplemente arrastrando las actividades disponibles
2. **Zona de monitorización** está diseñada para publicar las secuencias que fueron definidas con anterioridad. Primero se ha de seleccionar la secuencia en cuestión, posteriormente se elige el

grupo al que va dirigido ya sean grupos reducidos de alumnos o grupos de alumnos completos. Por último el profesor puede monitorizar la evolución de los alumnos.

3. **Zona de administración**, que está dedicada a la configuración de algunos parámetros del sistema, aunque a diferencia de otras herramientas no hay muchas opciones para configurar. De esta manera se pueden definir varias organizaciones con diferentes números de alumnos.
4. **Zona del alumno**, ya que los alumnos son los que van a ejecutar las secuencias de actividades que le han sido propuestas.

Las características de LAMS permiten al usuario adaptarlo a los métodos tradicionales de enseñanza que este utilice. Se utiliza como guía para la obtención de una herramienta que facilite la interactividad entre los usuarios y que beneficie la enseñanza virtual.

En nuestro proyecto de Laboratorios Virtuales adoptamos este sistema como punto de partida para el desarrollo de prácticas de laboratorios sobre Web, ateniéndonos a que podemos configurar para equipos de trabajo las secuencias lógicas en que transcurren las prácticas de laboratorio y analizar cada uno de los manejos que realizan los estudiantes en el equipo de trabajo que el profesor conformó.

Metodología a utilizar.

La calidad en el desarrollo y mantenimiento del software se ha convertido hoy en día en uno de los principales objetivos estratégicos de las organizaciones, debido a que cada vez los software son más importante en el desarrollo de las distintas esferas económicas ya que repercuten grandemente en estas. Debido a esto el control de la calidad de un software ya no es solo un proceso basado en la inspección y detección de errores, sino que los desarrolladores le dan una importancia y un seguimiento desde el mismo comienzo del proceso, es decir desde que todo comienza a formarse como una idea. Por lo que en los últimos años se han publicado diversos estudios y estándares en los que se exponen los principios que se deben seguir para la mejora tanto de productos como de procesos software.

Con el objetivo de garantizar la calidad en el desarrollo de un software se han establecido diversas metodologías, estas metodologías definen Quién debe hacer Qué, Cuándo y Cómo debe hacerlo, por lo

que de la correcta selección de la metodología dependerá, en su mayoría, la calidad del producto final. No existe una metodología de software universal. Las características de cada proyecto como son: equipo de desarrollo y recursos exigen que el proceso sea configurable. Para controlar, y planificar la propuesta que presenta este trabajo, se decidió utilizar como metodología el Proceso Unificado de Modelado (RUP), por sus características y las facilidades que aporta a todo el proceso.

RUP Se caracteriza por ser iterativo e incremental, estar centrado en la arquitectura y guiado por los casos de uso. Incluye Artefactos y Roles. Artefactos son los productos tangibles del proceso como por ejemplo, el modelo de casos de uso, el código fuente, etc., y el Rol es el papel que desempeña una persona en un determinado momento, una persona puede desempeñar distintos roles a lo largo del proceso. (VERA, 2007)

El RUP divide al proceso de desarrollo de un software en ciclos y cada ciclo se divide en fases. Las cuatro fases que forman, que representa un ciclo de desarrollo en la vida de un producto de software, son:

- **Conceptualización (Concepción o Inicio):** Se describe el negocio y se delimita el proyecto describiendo sus alcances con la identificación de los casos de uso del sistema.
- **Elaboración:** Se define la arquitectura del sistema y se obtiene una aplicación ejecutable que responde a los casos de uso que la comprometen. A pesar de que se desarrolla a profundidad una parte del sistema, las decisiones sobre la arquitectura se hacen sobre la base de la comprensión del sistema completo y los requerimientos (funcionales y no funcionales) identificados de acuerdo al alcance definido.
- **Construcción:** Se obtiene un producto listo para su utilización que está documentado y tiene un manual de usuario. Se obtiene 1 o varios release del producto que han pasado las pruebas. Se ponen estos release a consideración de un subconjunto de usuarios.
- **Transición:** El release ya está listo para su instalación en las condiciones reales. Puede implicar reparación de errores

Una de las características del RUP es que utiliza el UML como lenguaje de representación visual. RUP y UML están estrechamente relacionados entre sí, pues mientras el primero establece las actividades y los criterios para conducir un sistema desde su máximo nivel de abstracción (la idea en la cabeza del cliente), hasta su nivel más concreto (un programa ejecutándose en las instalaciones del cliente), el segundo

ofrece la notación gráfica necesaria para representar los sucesivos modelos que se obtienen en el proceso de refinamiento. (VERA, 2007)

El lenguaje UML (Lenguaje Unificado de Modelado) es un lenguaje para visualizar, especificar, construir y documentar los artefactos de un sistema que involucra una gran cantidad de software y establece la base del análisis y el diseño orientados a objetos, fundado en una amplia base de experiencia de los usuarios

El UML es un lenguaje gráfico para visualizar, especificar, construir y documentar los artefactos de un sistema con gran cantidad de software. UML proporciona una forma estándar de escribir los planos de un sistema, cubriendo tanto las cosas conceptuales, tales como procesos del negocio y funciones del sistema, como las cosas concretas, tales como las clases escritas en un lenguaje de programación específico, esquemas de bases de datos y componentes software reutilizables.

UML es una especificación de notación orientada a objetos. Divide cada proyecto en un número de diagramas que representan las diferentes vistas del proyecto. Estos diagramas juntos son los que representa la arquitectura del proyecto.

Tipos de Diagrama en UML:

- Funcionales: Muestran la funcionalidad del sistema desde el punto de vista del usuario. Incluyen:
 - Diagramas de caso de uso.
- Objetos: Muestran la estructura y la subestructura del sistema usando objetos, atributos, operaciones y asociaciones. Incluyen:
 - Diagramas de clase.
- Dinámico: Muestra el comportamiento interno del sistema, incluye:
 - Diagramas de secuencia
 - Diagramas de actividad
 - Diagramas de estados.

El UML utiliza el Rational Rose como la herramienta CASE. Esta herramienta propone la utilización de cuatro tipos de modelos para realizar un diseño del sistema, utilizando una vista estática y otra dinámica de los modelos del sistema, uno lógico y otro físico. Permite crear y refinar estas vistas creando de esta

forma un modelo completo que representa el dominio del problema y el sistema de software. A continuación los cuatro tipos de modelos:

- **Desarrollo Iterativo:** Utiliza un proceso de desarrollo iterativo controlado, donde el desarrollo se lleva a cabo en una secuencia de iteraciones. Cuando la implementación pasa todas las pruebas que se determinan en el proceso, ésta se revisa y se añaden los elementos modificados al modelo de análisis y diseño. Una vez que la actualización del modelo se ha modificado, se realiza la siguiente iteración.
- **Generador de Código:** Se puede generar código en distintos lenguajes de programación a partir de un diseño en UML.
- **Ingeniería Inversa:** Proporciona mecanismos para realizar la denominada Ingeniería Inversa, a partir del código de un programa, se puede obtener su diseño.
- **Trabajo en Grupo:** Permite varias personas trabajando a la vez en el proceso iterativo controlado, para ello posibilita que cada desarrollador opere en un espacio de trabajo privado que contiene el modelo completo y tenga un control exclusivo sobre la propagación de los cambios en ese espacio de trabajo.

Conclusiones

En este capítulo se realizó un análisis de cómo el surgimiento del e-learning ha contribuido a el surgimiento de comunidades virtuales de aprendizaje, las cuales constituyen un espacio virtual de intercambio de experiencias entre personas que se dedican a la labor de la enseñanza. Luego vimos la evolución de los laboratorios virtuales y los recursos informáticos que más se visualizan en el proceso de enseñanza-aprendizaje y la necesidad de realizar esa visualización de una manera interactiva. Después profundizamos sobre las principales herramientas utilizadas en la programación Web, así como la justificación de las elegidas para desarrollar este trabajo. También se detallan las posibilidades que brinda la tecnología AJAX y se describe las principales características de la metodología de investigación utilizada.

Capítulo 2

Capítulo 2. Descripción de la Solución Propuesta.

Introducción

En el presente capítulo se hace la descripción del estado actual del negocio, describiendo en que se basa la aplicación, además se describe la solución propuesta que trae este trabajo, describiendo para ello los procesos del negocio que tiene que ver con el objeto de estudio, también se definen conceptos que se agrupan en un Modelo de Dominio para poder hacer una captura correcta de los requisitos, tanto funcionales como no funcionales, para poder construir un sistema que satisfaga los objetivos propuestos.

Además, se enumeran los requisitos funcionales y no funcionales que debe tener el sistema que proponemos, lo que nos permite hacer una concepción general del sistema, e identificar mediante un Diagrama de Caso de Uso, las relaciones de los actores que interactúan con el sistema, y las secuencias de acciones con las que interactúan. Para luego describir detalladamente cada uno de los casos de uso, diciendo que acción se realiza, quien es el que la ejecuta y la respuesta del sistema a cada una de ellas.

Estado actual del negocio

El software LAMS ,o Sistema de Control de Actividades de Aprendizaje, en el cual se basa esta aplicación, es una aplicación Web que tiene como objetivo permitir que un profesor pueda construir una secuencia de actividades que debe seguir el alumno, es decir, que se utiliza para construir una clase sobre Web. Este software consta de varios módulos entre los que se encuentran el módulo de pizarra de anuncios, el módulo de Chat, el módulo de envío de archivos, el módulo de preguntas de opciones múltiples, módulo de encuesta, módulo de creación de equipos y el módulo de visualización. Este último tiene como objetivo que el profesor pueda añadir varios recursos que luego serán mostrados al alumno con el objetivo de crear nuevos conocimientos o consolidar los que ya el alumno tiene. El alumno deberá poder ver estos recursos posteriormente e interactuar con ellos.

A la hora de brindar estos recursos el profesor lo hace de una forma muy poco interactiva. Además, LAMS contiene algunas limitaciones en lo relativo a que se está obligado siempre a seguir un grupo de pasos lógicos, pero si no se solucionan concretamente cada tarea, no existe la posibilidad de retornar sobre los pasos y realizar otra secuencia en dependencia de la manipulación que se realizó en cada una de las actividades. Igualmente no se tienen incorporados elementos de inteligencia que permitan determinar o escoger otras secuencias en dependencia del nivel de colaboración y desempeño mostrado por el equipo en cada actividad.

Solución propuesta

La solución que se propone para lograr todos los objetivos propuestos en este trabajo es la elaboración de un módulo de visualización de recursos que permita que el profesor pueda ofrecer dichos recursos de una forma rápida e interactiva, que es sin duda una de las principales características que debe poseer una práctica de laboratorio virtual remota sobre Web. La interfaz de este módulo debe ser agradable y sencilla, ya que quien va a utilizar esta aplicación no tiene por que poseer un conocimiento avanzado en el trabajo en una computadora, es decir, lo mismo podrá utilizarla un profesor y alumnos de primaria que a penas saben manejar una computadora que alumnos universitarios que cursan carreras relacionadas con la computación.

Este módulo debe permitir que el profesor pueda brindar recursos tales como páginas Web de Internet o hechas por el y previamente publicadas especificando en ambos casos la URL de estas paginas, documentos de texto, imágenes y videos que deberán estar guardados en un servidor. Estos recursos serán acompañados de una breve descripción. El profesor podrá redactar unas instrucciones, utilizando un sencillo editor de texto, con el fin de ayudar que el estudiante sepa cual es el objetivo de la visualización, permitiendo así que lleguen a él todos los conocimientos que quieran ser transmitidos por el profesor.

Además, el profesor debe poder eliminar un recurso que haya puesto y que ya no quiera que sea visualizado o que quiera reemplazarlo por otro.

El alumno debe poder interactuar con los recursos. Esta interacción dependerá del tipo de recurso que sea. Si es una página Web debe poder navegar a partir de la misma, si es un documento se permitirá que se abra o se guarde en la computadora del estudiante, si es un video se podrá ver desde la propia pagina

y si es una imagen verá una vista previa y también se permitirá que se vea la imagen ampliada, con su tamaño real.

La interactividad requerida será lograda gracias a la utilización de la tecnología AJAX y la combinación de las librerías Rico y Prototype. Esto unido a la evolución de PHP permiten considerar la posibilidad de construir una aplicación Web modular que resuelva las limitaciones encontradas y permitan otras funcionalidades más adaptables a prácticas de laboratorios virtuales sobre Web, que es precisamente el interés del proyecto en nuestra Universidad.

Modelo del dominio

Se decide realizar modelo del dominio debido a que el negocio que estamos estudiando, tiene muy bajo nivel de estructuración, con soluciones muy diversas y dispersas nos basaremos en un modelo del dominio, ya que nos permite de manera visual mostrar al usuario los principales conceptos que se manejan en el dominio del sistema en desarrollo. Esto hará posible que los usuarios, los clientes y los desarrolladores e interesados utilicen un vocabulario común para poder entender el entorno en que se desarrolla el sistema. Para capturar correctamente los requisitos y poder construir un sistema correcto se necesita tener un firme conocimiento del funcionamiento del objeto de estudio. Este modelo va a contribuir posteriormente a identificar algunas clases que se utilizarán en el sistema.

Para identificar todos los conceptos que se utilizarán en el diagrama, se hará mediante un glosario de términos:

Se le denominará **Profesor** a aquel usuario que tenga el privilegio de elaborar las instrucciones y proporcionar los distintos recursos informáticos, que estarán previamente en un servidor, que luego serán visualizados por el resto de los usuarios independientemente de sus privilegios.

Se le denominará **Estudiante** a aquel usuario para el cual esta dirigida la visualización de los recursos.

Se le denominará **Usuario** a la generalización del profesor y del estudiante. El usuario solo podrá ver las instrucciones y los recursos, nunca podrá añadir un recurso o redactar las instrucciones.

Se le denominará **Instrucciones** a aquel texto construido por el profesor con el objetivo de guiar al estudiante durante el tiempo en que están siendo visualizados los recursos. El estudiante nunca podrá modificar ese texto.

Se le denominará **Páginas Web** a todas las URL de Páginas Web que el profesor quiera que sean visualizadas posteriormente al estudiante, permitiéndole navegar a partir de esta página Web. Es decir será un vínculo hacia una página de Internet o a una página hecha por el profesor que esta publicada en un servidor.

Se le denominará **Recurso** a aquellos archivos, ya sean documentos, imágenes o videos, que el profesor quiera que sean visualizados posteriormente al estudiante. El profesor debe especificar donde se encuentra el archivo almacenado y que tipo de archivo es.

Diagrama del Modelo del dominio

El diagrama que se muestra en la figura es el del modelo del dominio, mediante el cual se visualizan y relacionan las principales clases conceptuales del dominio.

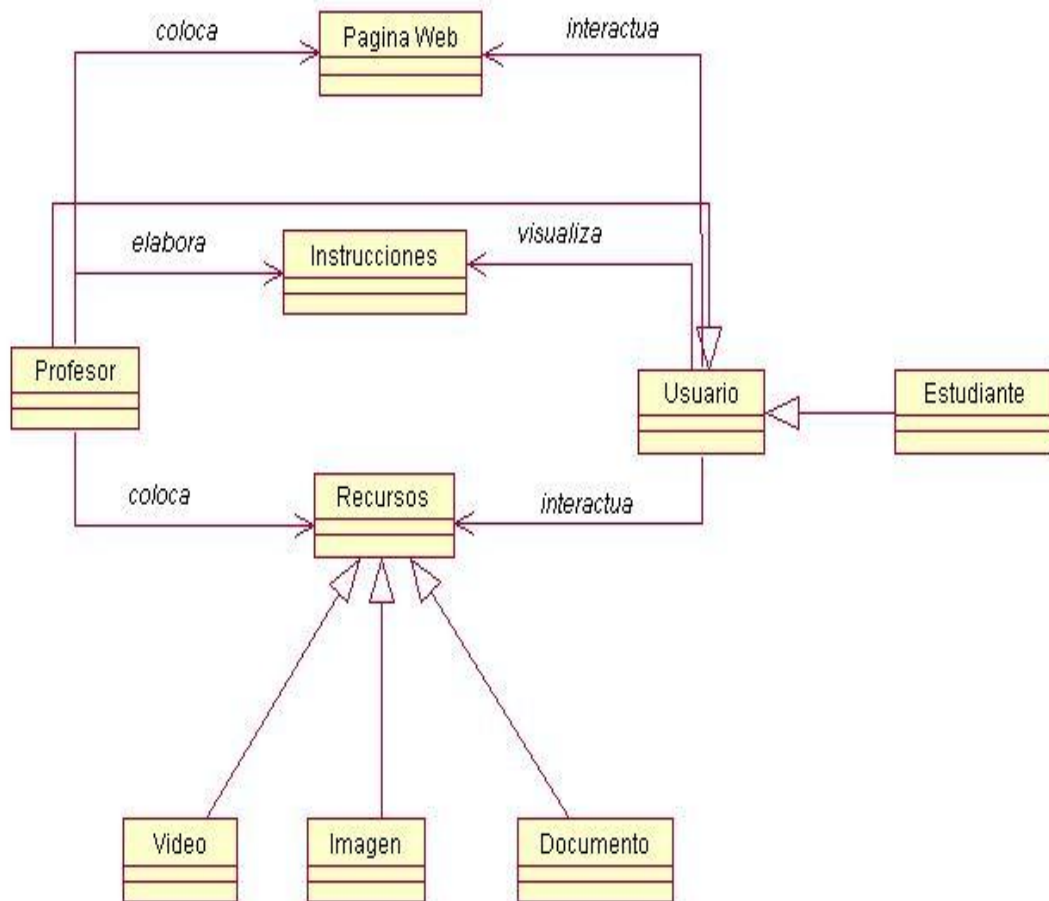


Figura 2.1: Diagrama del modelo del dominio.

Requisitos funcionales

Después de tener bien claros los conceptos que rodean al objeto de estudio y la forma en que estos se relacionan, se procede a enumerar los requisitos funcionales que harán posible que el sistema cumpla con los objetivos de este trabajo. Estos requisitos funcionales no son más que funciones que el sistema debe ser capaz de realizar. Dentro de ellos se incluyen las acciones que podrán ser ejecutadas por los usuarios y las acciones ocultas que realizara el sistema. De acuerdo con los objetivos plantados el sistema debe ser capaz de:

1. **Elaborar instrucciones.**
2. **Gestionar Página Web.**
 - 2.1 Añadir Página Web.
 - 2.2 Eliminar Página Web.
3. **Gestionar Recurso.**
 - 3.1 Añadir Recurso.
 - 3.2 Eliminar Recurso.
4. **Visualizar recursos.**
 - 4.1 Visualizar Instrucciones.
 - 4.2 Interactuar Página Web.
 - 4.3 Interactuar Documento.
 - 4.4 Visualizar Imagen.
 - 4.5 Visualizar Video.

Requisitos no funcionales

Los requisitos no funcionales son propiedades o cualidades que el producto debe cumplir. Estos requisitos son los que hacen que el producto sea atractivo, fácil de usar, interactivo usable, rápido, confiable o seguro. Estas características, en caso de no cumplirse, pueden limitar al sistema de una forma u otra.

Requisitos de Apariencia o interfaz externa

La interfaz del profesor y la interfaz del estudiante deben poseer las siguientes características:

- La aplicación debe poseer una interfaz agradable ya que se trata de una aplicación Web.
- Debe poseer una combinación de colores apropiada.
- La aplicación debe tener como principal característica la interactividad.
- Se necesita una interfaz profesional, pero a la vez sencilla.
- Debe tener la capacidad de brindar confianza y seguridad a los usuarios que le van a utilizar.

Requisitos de Portabilidad:

- Necesidad de que la aplicación sea multiplataformas.

- Debe garantizarse que se pueda cambiar a otro tipo de Base de Datos.
- Debe garantizarse que se pueda ver de la misma forma en los navegadores Internet Explorer 4, o superior, y en Mozilla Firefox.

Requisitos de Seguridad

- Garantizar de que el profesor sea únicamente el que pueda brindar algún recurso.
- Garantizar que cualquier usuario pueda ver las visualizaciones.

Requisitos de Rendimiento:

- Se debe lograr que la transmisión por parte del sistema se realice en el menor tiempo posible.
- La aplicación debe ser eficiente, rápida y precisa.
- El sistema debe garantizar estar disponible el mayor tiempo posible ya que debe permitir que sea utilizado en cualquier momento.
- Debe garantizarse que se puede entrar al sistema desde cualquier lugar de la red local.
- Debe garantizarse la integridad y precisión en la transmisión de la información.
- El sistema debe permitir que darle respuesta a varios usuarios que estén solicitando sus servicios a la misma vez.

Requisitos de ayuda:

- El sistema contará con una ayuda que guiará a los usuarios y permitirá que este se familiarice rápidamente con la aplicación.

Requisitos de Software:

- En la máquina cliente debe existir un navegador compatible o superior con Internet Explorer 4 o a Mozilla Firefox.

Requisitos de Hardware

- Microprocesador de 600 MHz.

- 128 Mb de memoria RAM
- 4 GB de disco duro.

Descripción del Sistema propuesto.

Para cumplir los objetivos propuestos en este trabajo y teniendo en cuenta todos los requisitos funcionales y no funcionales antes mencionados se construirá un módulo de visualización interactiva que permita a un profesor añadir un recurso y a un alumno ver e interactuar con este recurso. Este módulo se unirá a otros para conformar una aplicación Web que haga posible la elaboración de una secuencia de aprendizaje.

Para utilizar todas las funcionalidades del sistema se considerarán los roles de profesor y estudiantes, aunque se asume que estos cuando entran al módulo de visualización ya están registrados como tales. Los profesores son los únicos que podrán añadir un recurso o redactar las instrucciones.

Aunque la visualización irá dirigida al alumno, el profesor podrá ver una vista previa de los recursos que haya añadido para así tener una mejor idea de cómo llegará el mensaje que desea transmitirle al alumno.

Para lograr que el profesor se familiarice con la aplicación se pondrán una serie de vínculos en palabras claves, como por ejemplo en la interfaz que permite añadir un documento, una imagen o un video habrán vínculos hacia unas pequeñas ventanas que le comunicarán que tamaño y que tipo de extensión soporta cada uno de los recursos. Estas ventanas luego deberán ser cerradas, ya que es imposible navegar a partir de ellas ya que solo tienen el objetivo de orientar al usuario.

En la interfaz del profesor se utilizan los acordeones y en la de los estudiantes los tab de la librería RICO, lo que permite que exista gran cantidad de información y utilidades en un espacio reducido, ya que como este módulo se unirá a otros para formar una aplicación Web no se debe de hacer muchas páginas Web que hagan difícil la navegación en usuarios que no estén familiarizados con el trabajo en una computadora.

En la visualización de las imágenes se utilizan las posibilidades que brindan las librerías Prototype, la Litebox-1.0 y la Moo.fx. La combinación de estas tres librerías hacen posible que las imágenes puedan ser presentadas como vistas en miniatura y luego, con un simple clic sobre la imagen, ver la imagen en su tamaño original y moverse por toda la lista de imágenes sin tener que cerrar ninguna de ellas logrando la interactividad necesaria en toda clase de laboratorio virtual sobre Web.

Modelo del sistema

Utilizando las facilidades que nos brinda el UML, vamos a capturar los requisitos funcionales del sistema y a representarlos mediante un diagrama de casos de uso. Para ello tenemos que definir de acuerdo a lo planteado en los epígrafes anteriores, cuales serían los actores que van a interactuar con el sistema, y los casos de uso que me van a representar las funcionalidades.

Los actores no son parte del sistema sino que representan a terceros que interactúan con el sistema, es decir, es un rol de un usuario, que puede intercambiar información o puede ser un recipiente pasivo de información y representa a un ser humano, a un software o a una máquina que interactúa con el sistema. Estos se corresponden con trabajadores del negocio. El actor del sistema definido en este caso se define a continuación:

Actores	Justificación
Profesor	Representa a una persona que se encarga de elaborar las instrucciones, proporcionar los distintos recursos informáticos así de como eliminar un recurso existente, también podrá ver una vista previa de los recursos añadidos
Estudiante	Representa a una persona que podrá ver e interactuar con los recursos.
Profesor	Representa la generalización de los roles de estudiante y profesor para decir que ambos pueden ver los recursos.

Descripción de los casos de uso del sistema

A continuación se presentan los casos de uso que van a servir para satisfacer los requerimientos funcionales de sistema:

CU1.	Elaborar instrucciones.
Actor.	Profesor
Descripción.	El profesor elabora una serie de instrucciones, utilizando para ello un editor de texto sencillo. Las instrucciones servirán para que el alumno conozca el objetivo de la visualización y llegue mejor a él el mensaje que se desea transmitir con la misma.

Referencia.	R1.
-------------	-----

CU2	Gestionar Pagina Web.
Actor	Profesor.
Descripción	El profesor tiene la posibilidad de añadir, eliminar la URL de una página Web que se encuentre en Internet o publicada en un servidor local, así como redactar una pequeña descripción de la misma.
Referencia	R2.

CU3	Gestionar Recurso.
Actor	Profesor.
Descripción	El profesor tiene la posibilidad de añadir o eliminar un recurso, especificando si este recurso es un documento de texto, una imagen o un video. Especificando donde se encuentra dicho recurso y poniendo una breve descripción del mismo.
Referencia	R3.

CU4	Visualizar recurso
Actor	Profesor, Estudiante y Usuario.
Descripción	El profesor y el estudiante tienen la posibilidad de leer las instrucciones y de ver e interactuar con un recurso. En el caso de los documentos podrá guardarlos en su computadora.
Referencia	R4.

Diagrama de Casos de Uso del sistema

En el diagrama de casos de uso del sistema se representan las relaciones existentes entre los actores y los casos de uso:

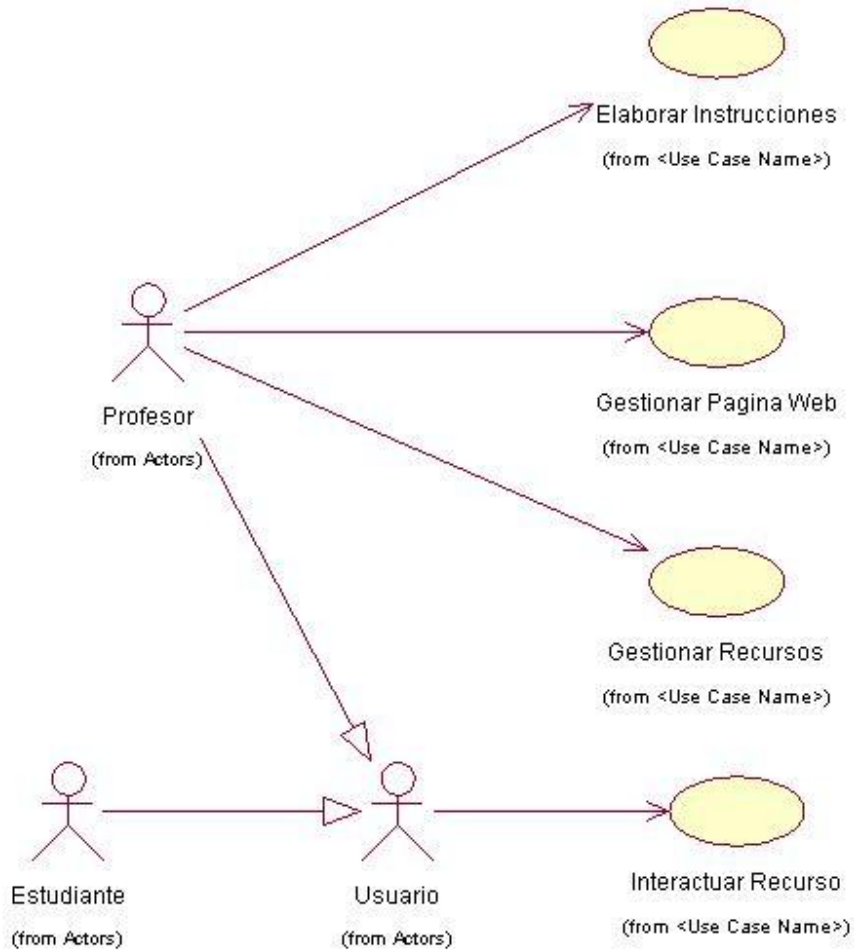
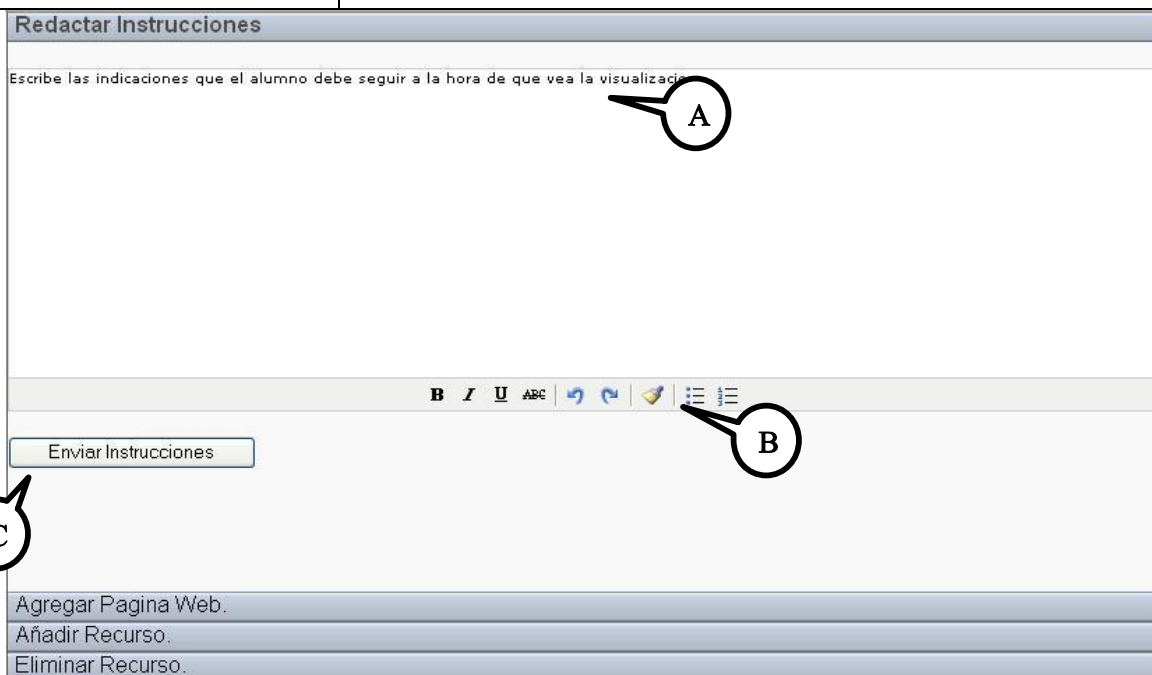


Figura 2.2: Diagrama de Caso de Usos del Sistema.

Casos de uso expandidos

Utilizando los casos de uso expandidos se hace una descripción paso a paso de la secuencia de acciones que los actores realizan para realizar un proceso a través del sistema. Este es el último paso en el análisis para luego comenzar con la construcción de la solución propuesta.

Caso de Uso Elaborar instrucciones	
Actores	Profesor.
Propósito	Permitir que un que el profesor pueda redactar un texto de instrucciones.
Resumen	El caso de uso se inicia cuando el profesor decide redactar un texto con las instrucciones que el estudiante deberá seguir en el transcurso de la visualización, para lograr el objetivo propuesto con la misma.
Referencias	R1
Precondiciones	El profesor debe haber entrado al sistema con ese rol.
Poscondiciones	Al final del proceso queda creado el texto de instrucciones.



Pantalla 1

Flujo normal de eventos	
Acción del Actor	Respuesta del sistema
1. El profesor entra a la aplicación y selecciona la opción de redactar instrucciones.	2. El sistema muestra la interfaz para redactar las instrucciones (ver Pantalla 1)
3. El profesor redacta las instrucciones (A)	4. El sistema almacena correctamente los datos de las

utilizando los componentes del editor de texto (B) y presiona el botón Enviar Instrucciones (C) para almacenarlas en la Base de Datos.	instrucciones.
Flujo alternativo	
Acción del Actor	Respuesta del sistema

Caso de Uso Gestionar Pagina Web.	
Actores	Profesor
Propósito	Permitir que un que el profesor pueda añadir la URL de una página Web que esté publicada en un servidor o en Internet o eliminar una que halla sido añadida.
Resumen	El caso de uso se inicia cuando el profesor decide que desea añadir una URL de una página Web que será visualizada posteriormente al estudiante, redactando una breve descripción de la misma, o cuando desea eliminar una pagina que halla sido previamente añadida y que ya no desea que sea visualizada.
Referencias	R2
Precondiciones	El profesor debe haber entrado al sistema con ese rol. Las páginas Web deben estar publicadas.
Poscondiciones	Al final del proceso serán añadidas una o varias páginas Web.
Flujo normal de eventos	
Acción del Actor	Respuesta del sistema
1. El profesor selecciona las operaciones a realizar.	2. a) Si selecciona añadir URL de página Web de Internet, ir a sección "Añadir página Web". b) Si selecciona eliminar página Web ir a sección "Eliminar página Web".
Sección "Añadir página Web"	

Redactar Instrucciones

Agregar Pagina Web.

Escriba la Url de la pagina Web que desea visualizar.

URL del sitio que el alumno debe visitar:

Descripcion:

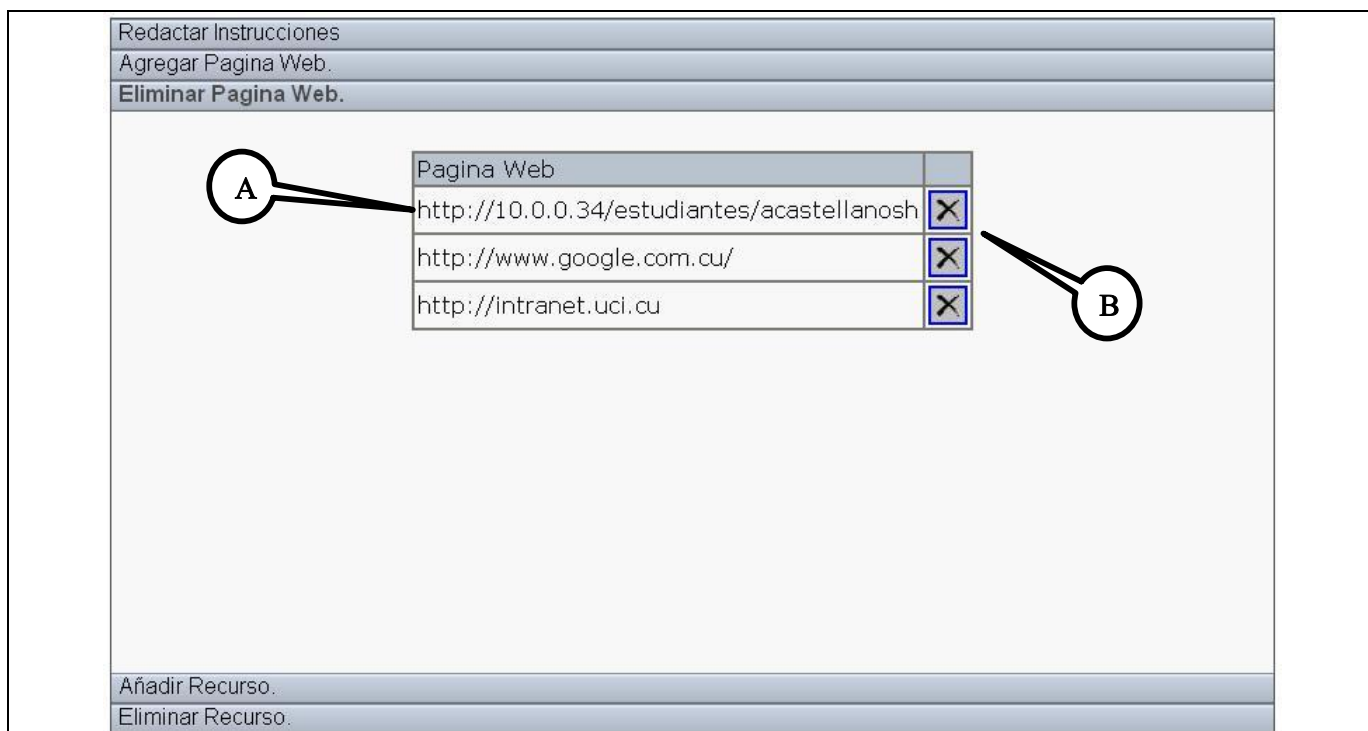
Eliminar Pagina Web.

Añadir Recurso.

Eliminar Recurso.

Pantalla2

Acción del Actor	Respuesta del sistema
<p>1. El profesor escribe la dirección URL de una página Web (A) (Ver Pantalla 2) que existe en Internet o que está publicada en un servidor, redacta una pequeña descripción (B) y oprime el botón “Enviar Pagina Web” (C).</p>	<p>2. El sistema añade a la Base de Datos la URL de la página Web que será visualizada posteriormente.</p> <p>3. El sistema redirecciona la página permitiendo que el profesor pueda hacer cualquier otra acción que desee en la interfaz dedicada a él, mostrando un mensaje confirmando que se añadió la URL de la página Web.</p>
<p>Sección “Eliminar página Web”</p>	



Pantalla 3

Acción del Actor	Respuesta del sistema
<p>1. El profesor decide eliminar una página Web y va a la sección dedicada a eliminarlas (Ver Pantalla 3).</p> <p>3. El profesor marca elige la página que quiere eliminar y pulsa el botón Eliminar (B).</p>	<p>2. El sistema le muestra las páginas que están almacenadas en la Base de Datos, (A).</p> <p>4. El sistema elimina la URL de la página Web.</p> <p>5. El sistema redirecciona la página permitiendo que el profesor pueda hacer cualquier otra acción que desee en la interfaz dedicada a él mostrando un mensaje confirmando que se eliminó URL de la página Web.</p>
Flujo alternativo	
Acción del Actor	Respuesta del sistema

Caso de Uso Gestionar Recurso	
Actores	Profesor
Propósito	Permitir que el profesor pueda añadir un recurso que se encuentra en un servidor o eliminar uno q haya añadido.

Resumen	El caso de uso se inicia cuando el profesor decide que desea añadir un recurso que será visualizado posteriormente al estudiante o eliminar un recurso que haya sido añadido.	
Referencias	R3	
Precondiciones	El profesor debe haber entrado al sistema con ese rol.	
Poscondiciones	Al final del proceso será añadido un recurso.	
Flujo normal de eventos		
Acción del Actor	Respuesta del sistema	
1. El profesor selecciona las operaciones a realizar.	2. a) Si selecciona añadir un recurso ir a sección "Añadir Recurso". b) Si selecciona eliminar recurso ir a sección "Eliminar Recurso".	
Sección "Añadir Recurso"		
<p style="text-align: center;">Pantalla 4</p>		
Acción del Actor	Respuesta del sistema	
1. El profesor pulsa el botón Examinar (A) (Ver Pantalla 4).	2. El sistema le muestra las distintas carpetas de la PC del profesor.	

<p>3. El profesor busca el Recurso y pulsa el botón “Abrir”.</p> <p>4. El profesor redacta una pequeña descripción del recurso (B).</p> <p>5. El profesor marca (C) que tipo de recurso es el añadido y pulsa el botón Enviar Recurso (D).</p>	<p>6. El sistema verifica que el recurso es del tipo que marcó el profesor.</p> <p>7. El sistema añade a la Base de Datos el recurso añadido.</p> <p>8. El sistema redirecciona la página permitiendo que el profesor pueda hacer cualquier otra acción que desee en la interfaz dedicada a él, mostrando un mensaje confirmando que se eliminó el recurso.</p>
--	---

Sección “Eliminar Recurso”



Pantalla 5

Acción del Actor	Respuesta del sistema
1. El profesor decide eliminar un recurso y va a la sección dedicada a eliminar los recursos (Ver Pantalla	2. El sistema le muestra los recursos que están almacenadas en la Base de Datos (A), con su

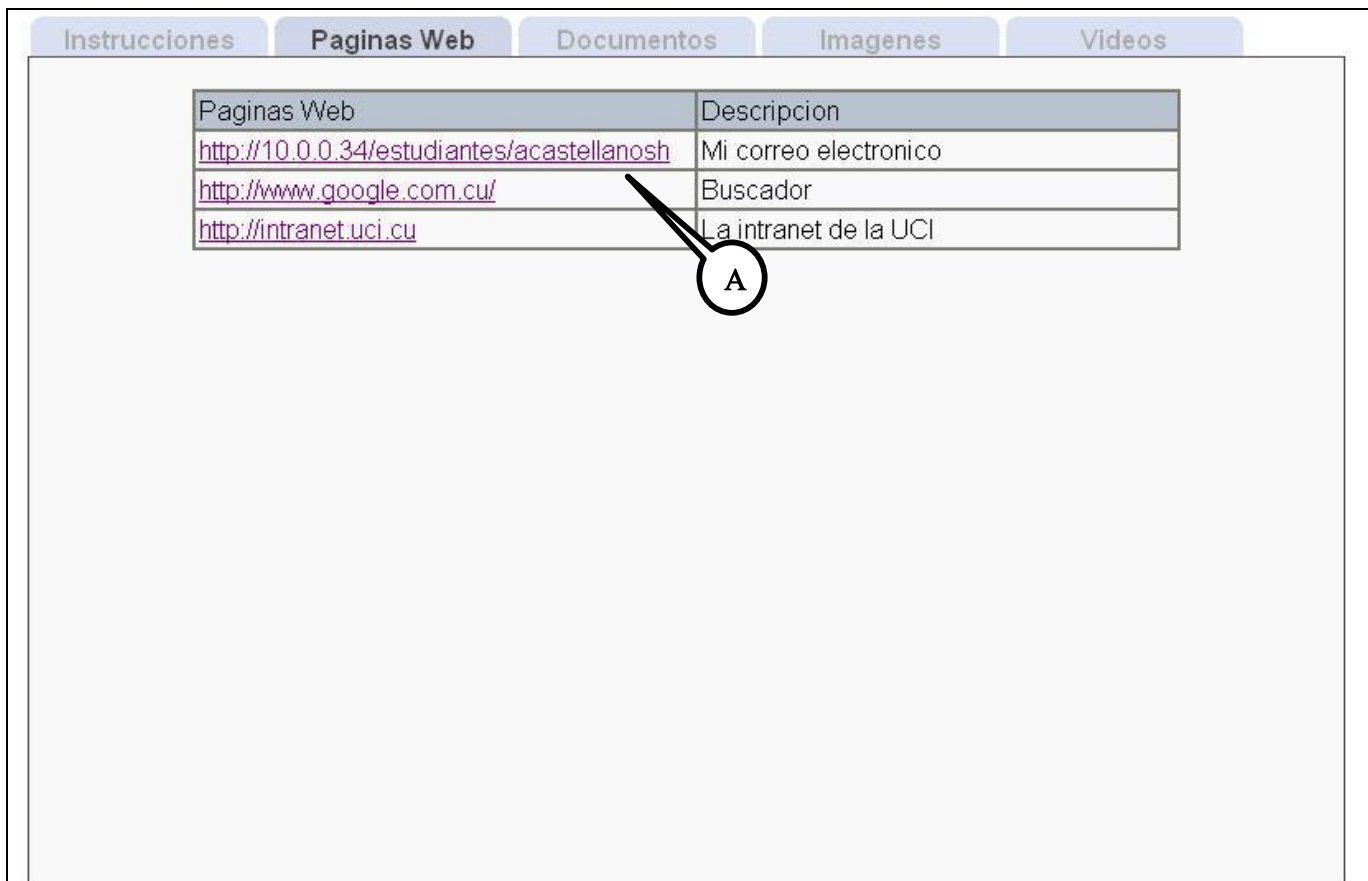
<p>5).</p> <p>3. El profesor elige el recurso que quiere eliminar y pulsa el botón Eliminar (B).</p>	<p>respectivo tipo de recurso.</p> <p>4. El sistema elimina el recurso de la Base de Datos.</p> <p>5. El sistema redirecciona la página permitiendo que el profesor pueda hacer cualquier otra acción que desee en la interfaz dedicada a él, mostrando un mensaje confirmando que se eliminaron los recursos.</p>
<p>Flujo alternativo Sección “Añadir Recurso”</p>	
<p>Acción del Actor</p>	<p>Respuesta del sistema</p>
	<p>7. El sistema le comunica al profesor que el recurso que quería añadir no es del tipo especificado.</p>

<p>Caso de Uso Visualizar Recursos.</p>	
<p>Actores</p>	<p>Estudiante, Profesor.</p>
<p>Propósito</p>	<p>Permitir que el estudiante o el profesor puedan ver las instrucciones y los recursos que el profesor haya añadido.</p>
<p>Resumen</p>	<p>El caso de uso se inicia cuando el profesor o el estudiante deciden que desea leer las instrucciones, ver los videos o las imágenes o guardar o abrir los documentos que el profesor haya añadidopreviamente.</p>
<p>Referencias</p>	<p>R4</p>
<p>Precondiciones</p>	<p>El profesor debe haber añadido algún recurso o redactado las instrucciones.</p>
<p>Poscondiciones</p>	<p>Al final de proceso serán visualizadas las instrucciones y los recursos.</p>
<p>Flujo normal de eventos</p>	



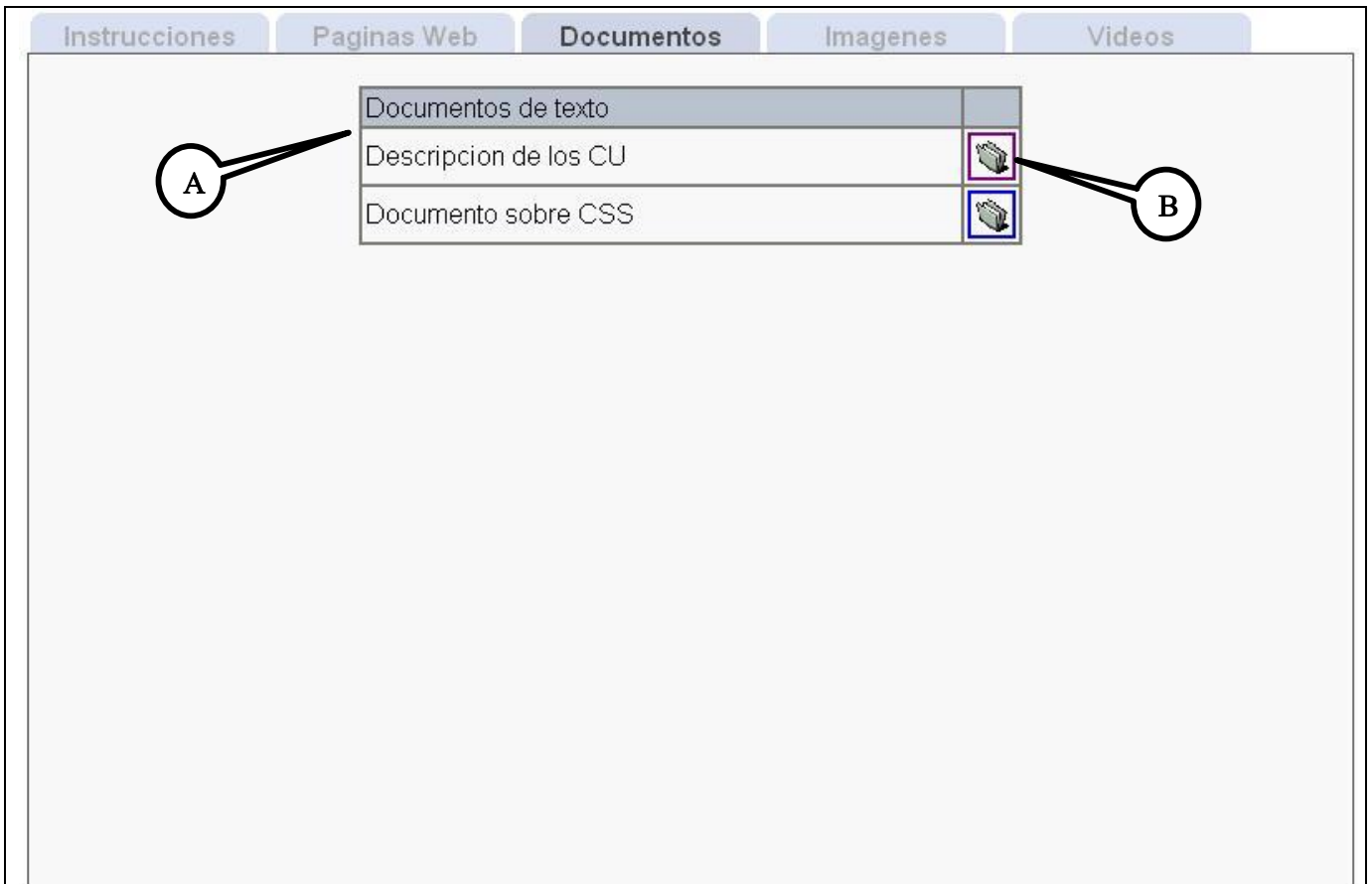
Pantalla 6

Acción del Actor	Respuesta del sistema
1. El usuario lee las instrucciones (A) y seleccionan las operaciones a realizar (Ver Pantalla 6).	2. a) Si selecciona ver páginas Web ir a sección “Página Web”. b) Si selecciona ver documentos ir a sección “Documentos”. c) Si selecciona ver imágenes ir a sección “Imágenes”. d) Si selecciona ver videos ir a sección “Videos”.
Sección “Paginas Web”	

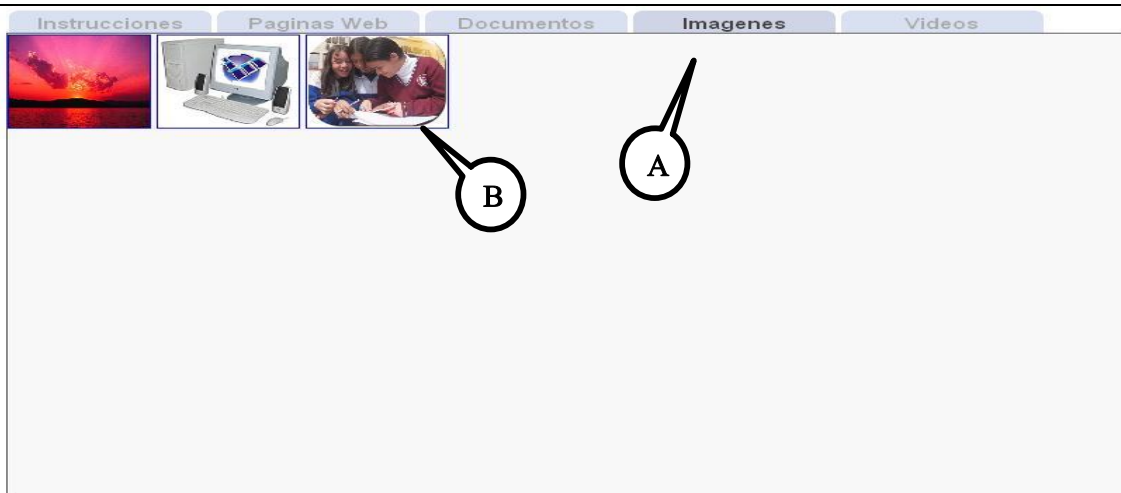


Pantalla 7

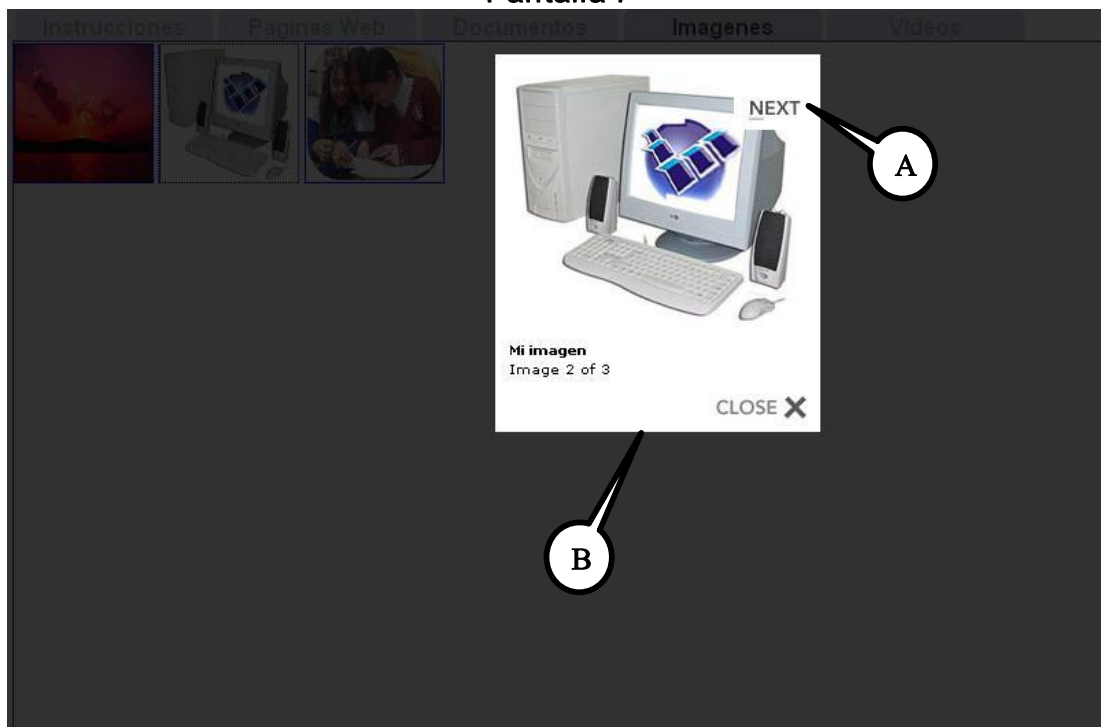
Acción del Actor	Respuesta del sistema
1. El usuario pulsa el tab "Página Web".	2. El sistema le muestra las URL de las páginas Web que han sido añadidas y la breve descripción de las mismas.
3. El usuario presionan el nombre de la URL (A).	4. El sistema abrirá la página Web que corresponda a esa URL en el navegador que utiliza el estudiante.
Sección "Documentos".	



Acción del Actor	Respuesta del sistema
<p>1. El usuario pulsa el tab “Documento”</p> <p>3. El usuario pulsa sobre la imagen de la carpeta abierta para abrir el documento.</p> <p>5. El usuario decide abrir el documento.</p>	<p>2. El sistema le muestra los documentos que han sido añadidos (A).</p> <p>4. El sistema le muestra la opción de abrir o guardar el documento.</p> <p>6. El sistema abre el documento para que el usuario pueda visualizarlo, permitiendo que este pueda ser sobrescrito.</p>
<p>Sección “Imágenes”.</p>	



Pantalla 7

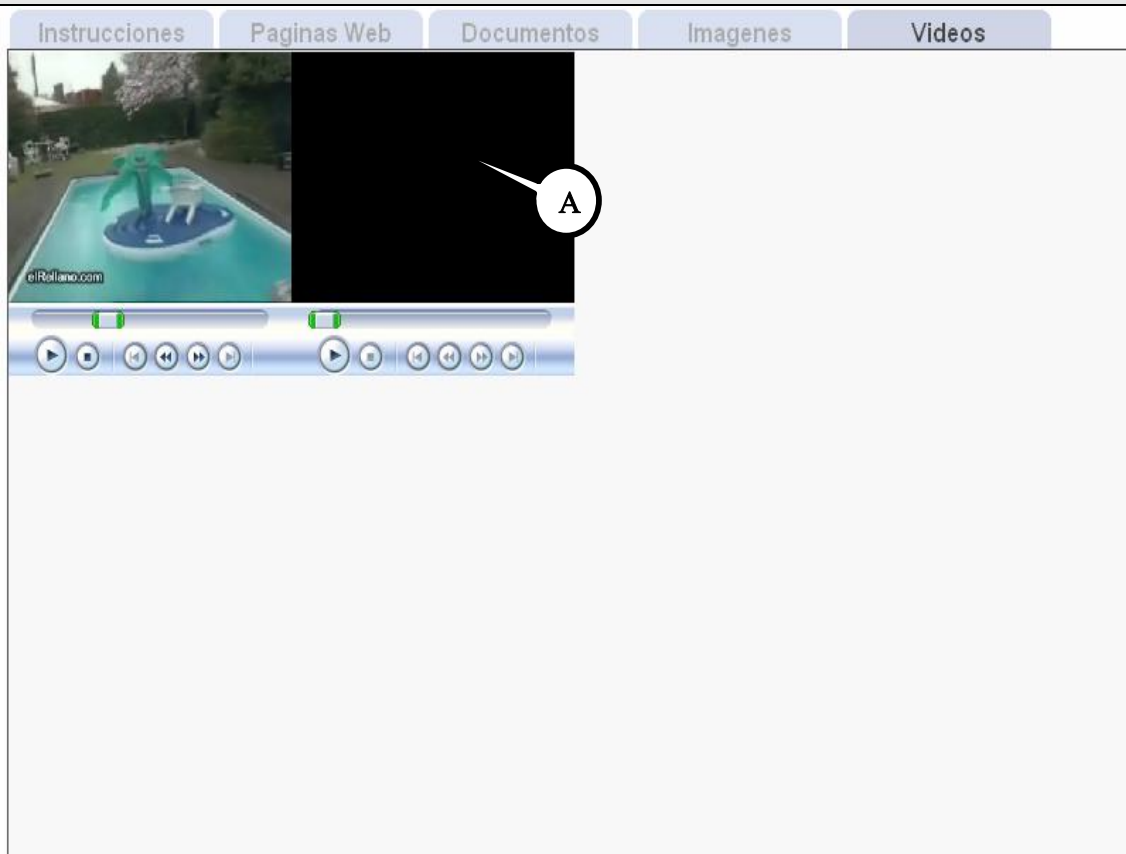


Pantalla 8

Acción del Actor	Respuesta del sistema
1. El usuario pulsa el link “Imágenes” (A)	2. El sistema le muestra una interfaz donde hay una vista en miniatura de las imágenes que tienen en la base de Datos (ver Pantalla 7).

<p>3. El usuario selecciona la imagen que quiere ver.</p>	<p>4. El sistema le muestra la imagen ampliada, con su tamaño original y los botones “PREV”, “NEXT” (A) y “CLOSE” (B) sobre la propia imagen (ver Pantalla 8).</p>
<p>5. El usuario pulsando sobre uno de los botones dependiendo de la acción que desee realizar.</p>	<p>6. El sistema ejecuta la acción.</p>

Sección “Añadir Video”



Pantalla 9

Acción del Actor	Respuesta del sistema
<p>1. El usuario pulsa el tab “Videos”</p>	<p>2. El sistema le muestra una vista en miniatura de los videos que tienen en la Base de Datos (Ver Pantalla 9) permitiendo que el estudiante o el profesor pueda utilizar los botones del reproductor.</p>

3. El estudiante o el profesor dan doble click sobre el video.	4. El sistema le muestra la reproducción del video en modo pantalla completa.
--	---

Conclusiones

En este capítulo se comenzó a desarrollar la propuesta de solución, obteniéndose a partir del análisis de los procesos del negocio, un listado con las funciones que debe tener el sistema, permitiendo una mejor comprensión del contexto a automatizar y de las características y restricciones que deben existir en el sistema para cumplir con los requerimientos de los clientes. Además, se realizó una descripción de la solución propuesta y se definieron los requisitos funcionales y no funcionales que debe cumplir.

Luego se describieron paso a paso todas las acciones de los actores del sistema con los casos de uso con los que interactúan.

Con el desarrollo de este flujo de trabajo y los artefactos obtenidos a partir de este, se puede pasar al flujo de diseño para comenzar la construcción de la solución de software propuesta, el cual será presentado en el próximo capítulo.

Capítulo 3

Capítulo 3. Construcción de la solución propuesta

Introducción

El presente capítulo expone la construcción de la solución propuesta, a través de los flujos de trabajo de diseño e implementación que se usan para la construcción de una aplicación Web. Tratando a los componentes de la aplicación como clases, utilizando las extensiones del UML, se presentan a utilizando el diagrama de clases Web. Luego se presenta el diagrama de clases persistentes, así como el modelo de datos obtenido a partir de éste para construir finalmente la base de datos que soportará el trabajo del sistema. Posteriormente, se explican los principios de diseño de la aplicación. Se desarrolla y se representa el diagrama de despliegue. Por último se les realizan al software una serie de prueba de la caja negra.

Patrones de Diseño

Para el desarrollo de la aplicación se empleó el patrón de diseño Modelo-Vista-Controlador (MVC) descompone una aplicación interactiva en tres grandes bloques:

El *modelo* contiene los datos y la funcionalidad de la aplicación. Es independiente de la representación de los datos.

Los objetos de negocio son los objetos del Modelo que implementan la lógica de negocio. Sus funciones fundamentales son:

- Realizar la validación de los datos introducidos por el usuario, tanto sintáctica (numérico, fecha) como lógica (importe menor que saldo.)

- Ejecutar la petición realizada por el usuario. Para ello podrá valerse de transacciones contra Sistemas Host, consultas a bases de datos, consultas a proveedores de contenidos, etc.
- Generar los objetos que utilizarán las Vistas para mostrar los resultados obtenidos.
- Garantizar la navegación y el flujo de pantallas correcto.

Las **vistas** muestran la información al usuario de una cierta forma. Existen todas las que se necesite definir, es la encargada de interactuar con el usuario y se corresponde con lo que tradicionalmente se conoce como Interfaz de Usuario.

Para este caso las vistas serán del tipo PHP. Las PHP son las encargadas de generar las páginas HTML que constituyen la visualización de los resultados generados por la petición realizada por el usuario. Para recibir la información que deben mostrar acceden a objetos generados por el Modelo. La recepción de la información se puede hacer de varias maneras, aunque las más habituales son:

- Accediendo a objetos almacenados dentro del Modelo como propiedades.
- El Modelo almacena los resultados como atributos de la petición.
- El Modelo escribe los resultados en la "sesión".

Cada *vista* tiene un **controlador** asociado. Los controladores reciben entradas en forma de eventos que responden a mandos realizados por el usuario a través del ratón o del teclado. El control traduce estos eventos a peticiones a la *vista* o al *modelo*.

Cada petición se identifica mediante un parámetro. En base a esta identificación, el Controlador decide qué objeto u objetos de negocio (Modelo) debe ejecutar para resolver la petición.

Tras la ejecución de los objetos de negocio, y en función del resultado devuelto por estos, el Controlador determina qué PHP se usará para visualizar el resultado, generando una redirección que concluirá con la generación del código HTML de la página.

Ventajas

- Múltiples vistas del mismo modelo.
- Vistas sincronizadas.
- Flexibilidad para cambiar las vistas y los controladores.
- La aplicación puede soportar distintos tipos de interfaz de usuario.
- Aumenta en gran medida el nivel de reusabilidad de código. Facilita una evolución continuada de los sistemas, sin puntos de ruptura, ya que un cambio en un sistema afectará a uno o más componentes pero nunca afectará significativamente al "core" de la aplicación.
- Facilidad de desarrollo y acortamiento del "Tiempo de Comercialización" gracias a la paralelización de tareas.

Inconvenientes

- Complejidad creciente.
- Cambios innecesarios. Puede ser que no todas las vistas estén interesadas en los cambios.
- Conexión entre la vista y el controlador. Hay que usar los dos a la vez.
- Si cambia el interfaz del modelo, hay que cambiar todas las vistas y todos los controladores.
- Acceso ineficiente a los datos en la vista. Puede necesitar varias llamadas al modelo para actualizar todos sus datos.
- Tanto la vista como el controlador son específicos de una plataforma.
- Algunas herramientas de diseño de interfaces de usuario incorporan parte del procesamiento de eventos entrada. El controlador deja de ser necesario.

Diagrama de Clases del Diseño.

El modelo de diseño es un modelo de objetos que describe la realización física de los casos de uso centrándose en como los requisitos funcionales y no funcionales, junto con otras restricciones relacionadas con el entorno de implementación, tienen impacto en el sistema a considerar. Sirve de

abstracción de la implementación y es utilizada como entrada fundamental de las actividades de implementación.

El diagrama de clases para las Aplicaciones Web, difiere un poco del resto de las aplicaciones que estamos acostumbrados a construir, puesto que en ellas son más importantes la modelación de la lógica y estado del negocio que los detalles de presentación. En el caso específico de las Aplicaciones Web, en el diagrama de clases se definen las pautas para el diseño de la interfaz de la aplicación partiendo de la importancia que representa para la obtención del producto final como para el usuario que interactúa con la misma, además se modelan las páginas y los enlaces entre estas, todo el código que irá creando las páginas, así como el contenido dinámico de las mismas, una vez que estén en el navegador del cliente.

Teniendo en cuenta esto se realizan diagramas, como por ejemplo el de clases, agrupados en paquetes, empleando los mecanismos de extensión definidos por el UML para aplicaciones Web. Estos paquetes coinciden con los casos de uso para que pueda entenderse mejor la lógica de la aplicación.

A continuación se presentan los diagramas de clase Web de los casos de uso.

Diagrama de clases del caso de uso “Elaborar Instrucciones”

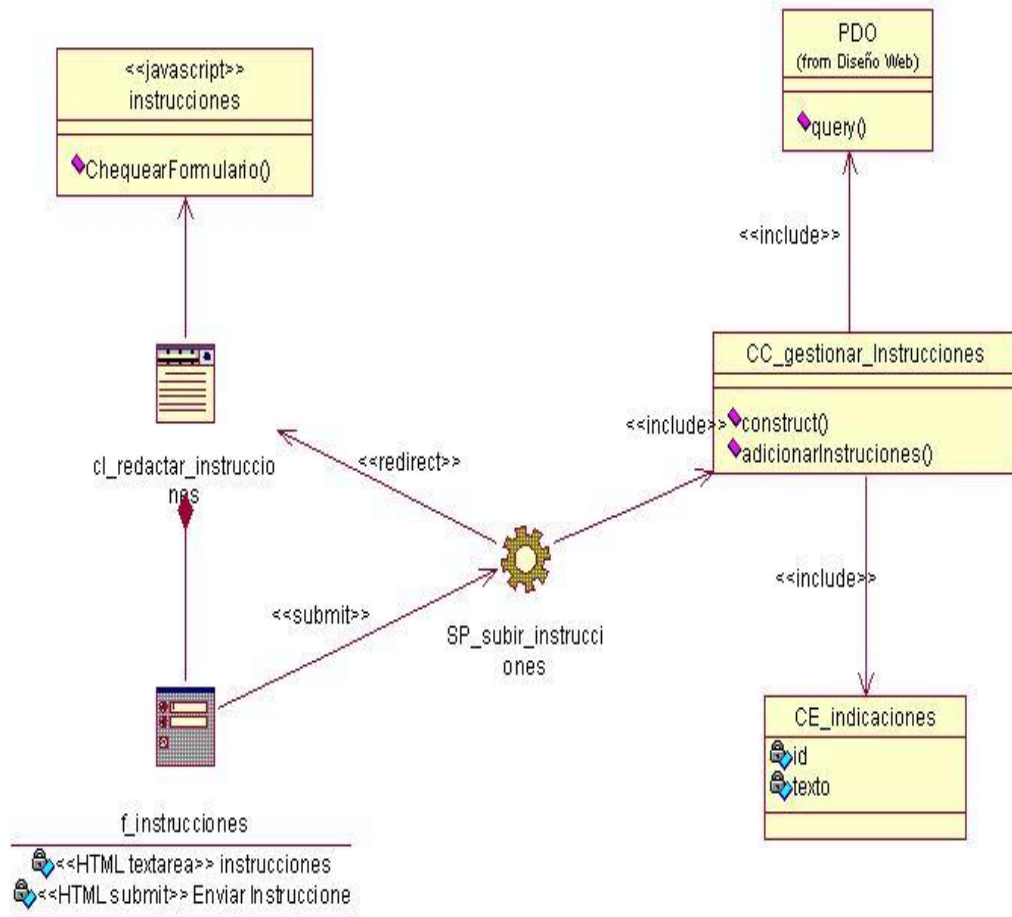


Figura 3.1: Diagrama de clases del diseño Redactar Instrucciones.

Diagrama de secuencia del caso de uso “Elaborar Instrucciones”. [Ver Anexo 1].

Diagrama de clases del caso de uso “Gestionar Pagina Web”

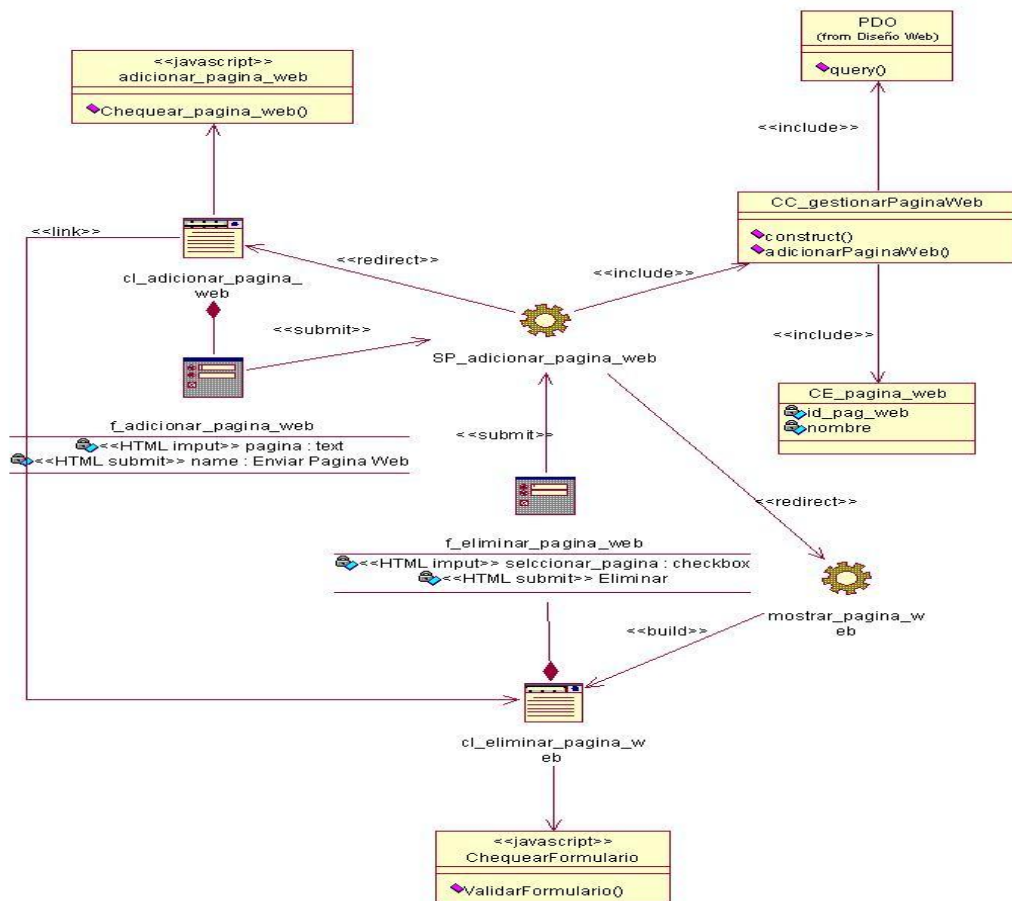


Figura 3.2: Diagrama de clases del diseño Gestionar Página Web.

Diagrama de secuencia del caso de uso “Gestionar Pagina Web” . Sección Adicionar Pagina Web [Ver Anexo 2].

Diagrama de secuencia del caso de uso “Gestionar Pagina Web” . Sección Eliminar Pagina Web [Ver Anexo 3].

Diagrama de clases del caso de uso “Gestionar Recurso”

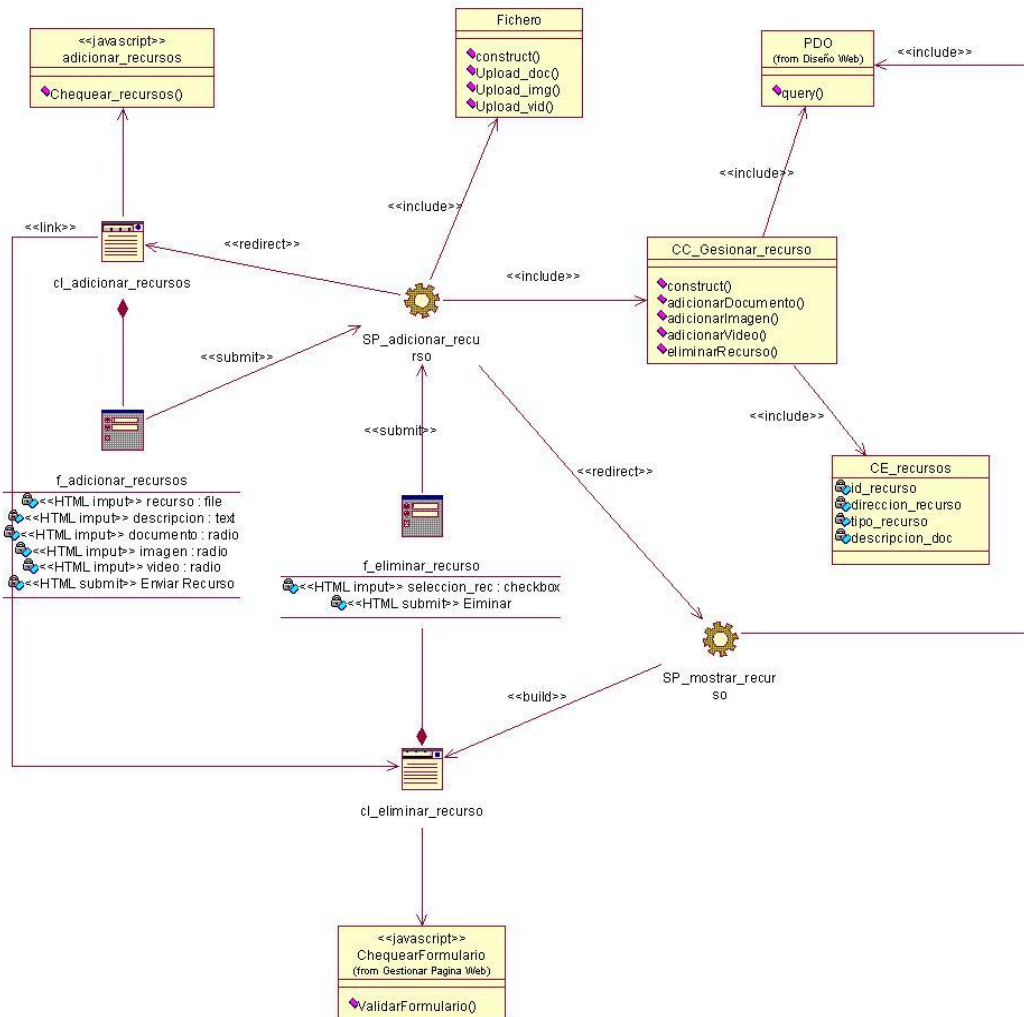


Figura 3.3: Diagrama de clases del diseño Gestionar Recursos.

Diagrama de secuencia del caso de uso “Gestionar Recurso”. Sección Adicionar Recurso [Ver Anexo 4].

Diagrama de secuencia del caso de uso “Gestionar Recurso”. Sección Eliminar Recurso [Ver Anexo 5].

Diagrama de clases del caso de uso “Interactuar Recurso”

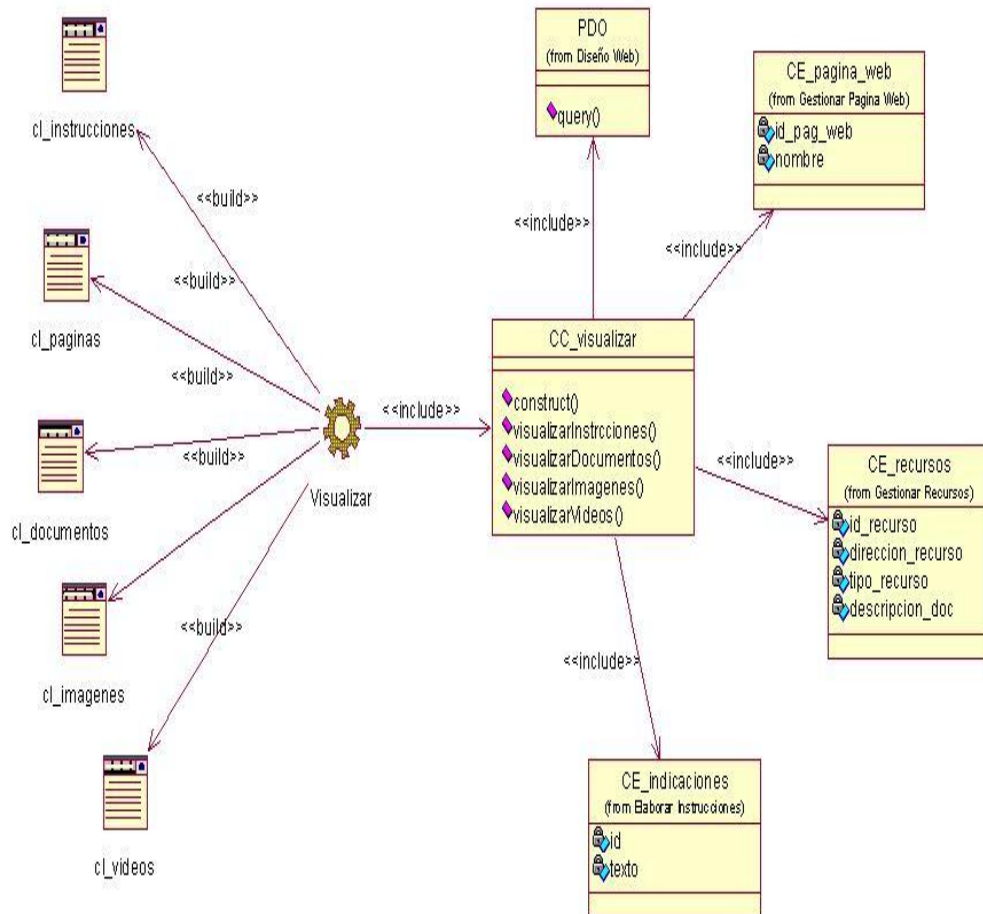


Figura 3.4: Diagrama de clases del diseño Interactuar Recurso.

Diseño de la Base de Datos.

Las clases persistentes son aquellas que contienen y almacenan los datos durante los distintos procesos de la aplicación, estas clases se pueden modelarse en el diagrama de clases persistentes. En este

diagrama se muestran relaciones entre los distintos datos que almacenan esas clases. Esto es muy importante a la hora de modelar la base de datos del sistema.

En el diagrama de clases persistentes las clases no se relacionan entre si ya que en este caso especifico cada una de las clases son independiente a las otras, ya que a la hora de manejar los datos de una clase, ya sea introducir nuevos datos o utilizar los que ya tienen las clases no se necesita ni se modifica ninguna otra clase

Diagrama de clases persistentes

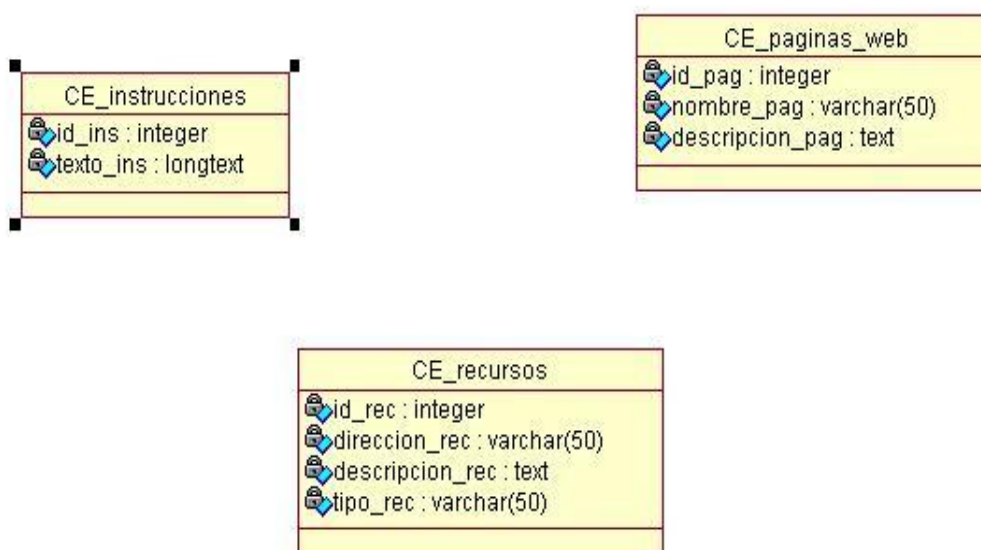


Figura 3.5: Diagrama de clases Persistente.

Modelo de datos

El modelo de datos que se obtuvo a partir del diagrama de clases persistentes y se muestra a continuación.

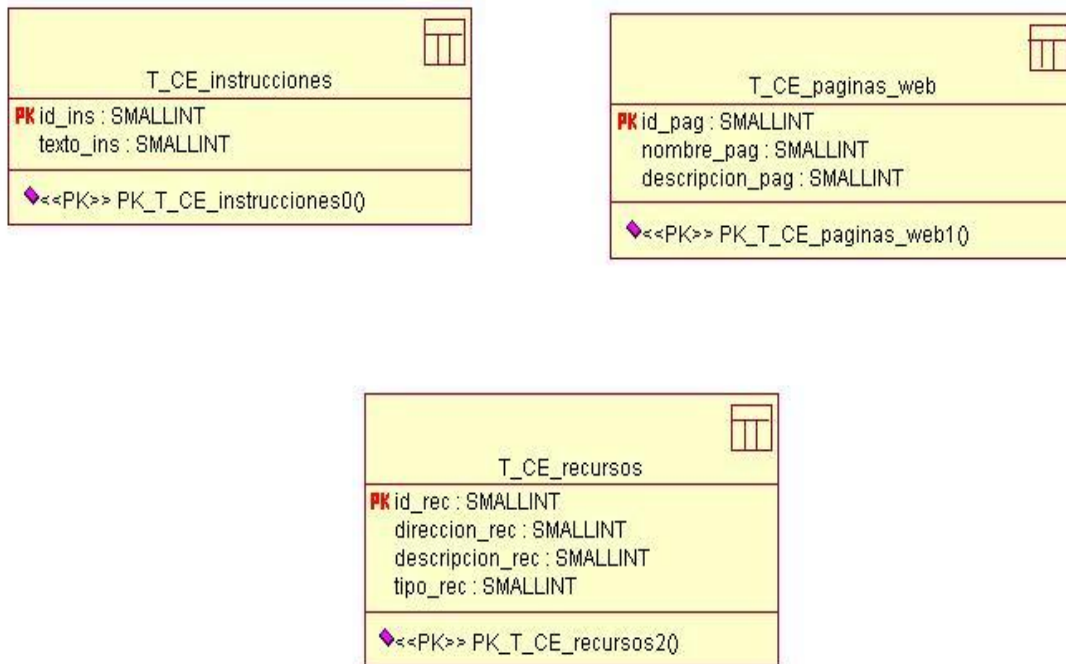


Figura 3.6: Diagrama Modelo de datos.

Descripción de las Tablas.

NOMBRE: T_INSTRUCCIONES		
DESCRIPCION: Tabla que almacena el texto redactado por el profesor con las instrucciones que el estudiante debe seguir durante la visualización de los recursos. Este texto se mostrará en la interfaz dedicada al estudiante.		
ATRIBUTO	TIPO	DESCRIPCION
id_ins	integer	Identificador de tabla
Texto_ins	lontext	Texto con las instrucciones.

NOMBRE: T_PAGINAS_WEB		
DESCRIPCION: Tabla que almacena el nombre de una URL que conduce a una pagina Web de Internet o publicada en un servidor local. Este nombre será mostrado al estudiante y le permitirá que pueda acceder a dichas páginas a través de él.		
ATRIBUTO	TIPO	DESCRIPCION
Id_pag	integer	Identificador de tabla
nombre_pag	varchar(50)	Nombre de una URL que conduce a una página Web.
descripción_pag	text	Breve descripción de la página Web.

NOMBRE: T_RECURSOS		
DESCRIPCION: Tabla que almacena la dirección física en la que se encuentra el recurso y una breve descripción del mismo y el tipo de recurso que es. La dirección del recurso hace posible que este sea mostrado al estudiante y el tipo determina donde y de que forma será mostrado dicho recurso		
ATRIBUTO	TIPO	DESCRIPCION

id_rec	integer	Identificador de tabla
direccion_rec	varchar(50)	Direccion física del lugar, en el servidor, donde se encuentra el recurso.
descripcion_rec	text	Breve descripción del recurso
tipo_rec	varchar(50)	Indica el tipo de recurso que es en dependencia si es un documento (doc), una imagen (jpg) o un video (avi).

Principios de Diseño.

El diseño de la interfaz de una aplicación Web, sea cual sea su objetivo, tiene que basarse esencialmente en el usuario para el cual va dirigido, en este caso profesores y estudiantes. Estos muchas veces no tienen una amplia preparación en cuanto a la utilización de aplicaciones informáticas, por lo que se tiene que lograr que la interfaz gráfica del usuario sea lo más amigable posible y lograr que este se sienta identificado con la misma ya que esta interfaz es el medio por el cual el usuario interactúa con el sistema.

Estándares en la interfaz de la aplicación.

La página principal de la aplicación la constituye la interfaz dirigida al profesor. En ella el profesor puede añadir todos los recursos que serán visualizados posteriormente. Esta página está formada por un acordeón construido utilizando la librería RICO y la Prototype, el cual posibilita que se pueda presentar en un espacio aparentemente reducido gran cantidad de información, ya que está contenida en un menú desplegable. En esta interfaz se utiliza el color gris en varias de sus tonalidades, para diferenciar cuando un menú está elegido y cuando no. En los distintos botones se usan nombres claros que hacen que cualquier usuario pueda entender cual utilizar para cada una de las funcionalidades.

Se utilizó las hojas de estilo para poder hacer cambios en el estilo de los distintos componentes de los acordeones sin afectar la funcionalidad de los mismos. Utilizándose como tipo de letra la Arial y la Verdana en su tamaño normal.

Para la mayor comprensión de donde el profesor puede redactar el texto con las instrucciones, en el área dedicada a esta funcionalidad aparecerá un texto, que puede ser borrado, que le indica al profesor que

ese es el sitio donde escribir las indicaciones que el alumno seguirá durante la visualización, logrando así que el profesor entienda mejor la interfaz.

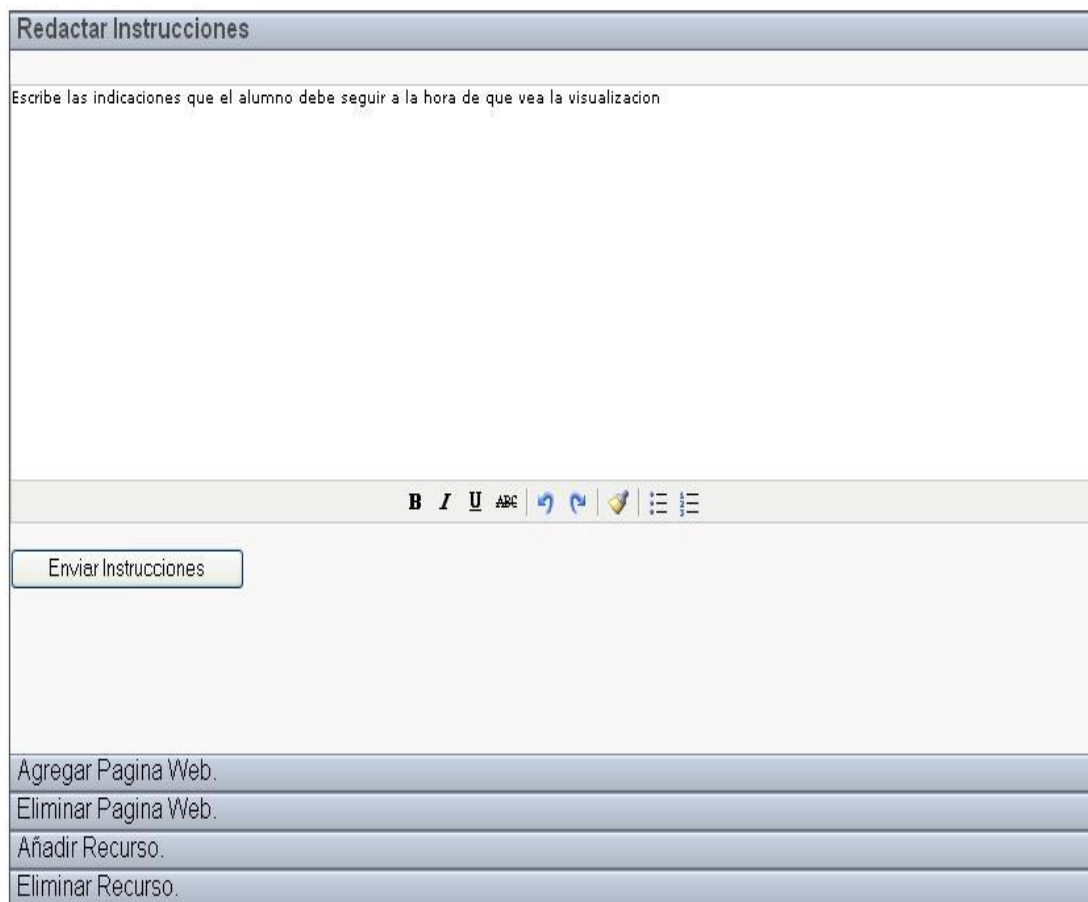


Figura 3.7: Interfaz Principal del Profesor.

En la interfaz dedicada al alumno aparecerá la visualización de todos los recursos. Aquí el alumno puede ver las indicaciones y cada uno de los recursos que el profesor le orientó. En esta página se utilizan los tabs que proporcionan la librería RICO, resaltando siempre el que está seleccionado. Lo que hace que la navegación por esta página se haga de forma rápida y sencilla. Aquí también se utilizan las distintas tonalidades del gris como color base, lo que hace que la interfaz sea amigable.



Figura 3.8: Interfaz Principal del Estudiante.

Formatos de reportes.

Los reportes estarán presentes en la página dedicada al profesor, en las secciones de eliminar páginas Web y eliminar recursos ya que en esas interfaces se le presentarán las listas con los recursos o las páginas Web que están en la base de datos para que el elija la que quiere eliminar.



Figura 3.9: Reporte en la Interfaz del profesor "Eliminar recurso".

Reporte en la Interfaz del profesor "Eliminar Página Web" [Ver Anexo 6].

También se existirán reportes en las distintas interfaces dedicadas a los estudiantes, ya que a estos se le presentaran las indicaciones y los distintos recursos que fueron brindados por los profesores. Las imágenes y los videos se les presentarán en el fondo de la interfaz dedicada a cada uno de los recursos como una pequeña vista previa de los mismos, los documentos serán presentados como archivos con sus nombres y se les podrán guardar o abrir para ver su contenido.

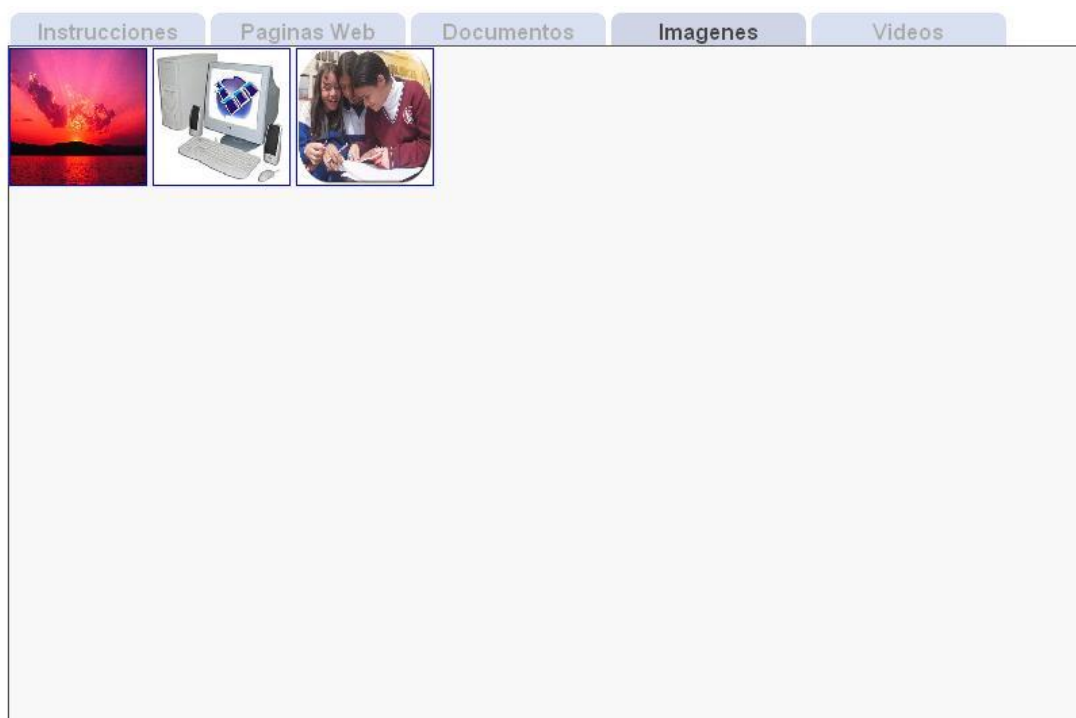


Figura 3.10: Reporte en la Interfaz del estudiante “Mostrar imágenes”.

Reporte en la Interfaz del estudiante “Mostrar Páginas Web” [Ver Anexo 7]

Reporte en la Interfaz del estudiante “Mostrar Documentos” [Ver Anexo 8]

Reporte en la Interfaz del estudiante “Mostrar Videos” [Ver Anexo 9]

Tratamiento de excepciones.

Valorando la importancia que requiere el tratamiento de los errores, teniendo en cuenta siempre la creación de interfaces amigables, en el sistema propuesto se hace un estricto control de los mismos y su posibilidad de ocurrencia, empleando validaciones encargadas de chequear la existencia de cualquier fallo producido.

Para garantizar la integridad de los datos y evitar errores que se introducen al sistema, se trata de prevenir la incorrecta entrada de datos; en caso de existir un dato incorrecto se notifica al usuario con un mensaje de error. De este modo cada formulario que se muestra al cliente se valida antes de ser enviados al servidor. Se le mostrara un mensaje de error al usuario cuando este no ingrese ningún archivo y pulse el

botón “Enviar” (Ver Figura 3.11). Además, se validará por parte del servidor que el profesor envíe el tipo de archivo correcto en cada caso, es decir que el profesor envíe una imagen si señaló que enviaría ese recurso, impidiendo así que no envíe ningún otro tipo de archivo, mostrándole un mensaje al profesor si el archivo no es del tipo esperado. Además se le muestra un mensaje al profesor si trata de enviar un archivo más grande de lo permitido. También, se utilizan mensajes de confirmación, para acciones que son irreversibles como es el caso de las eliminaciones.

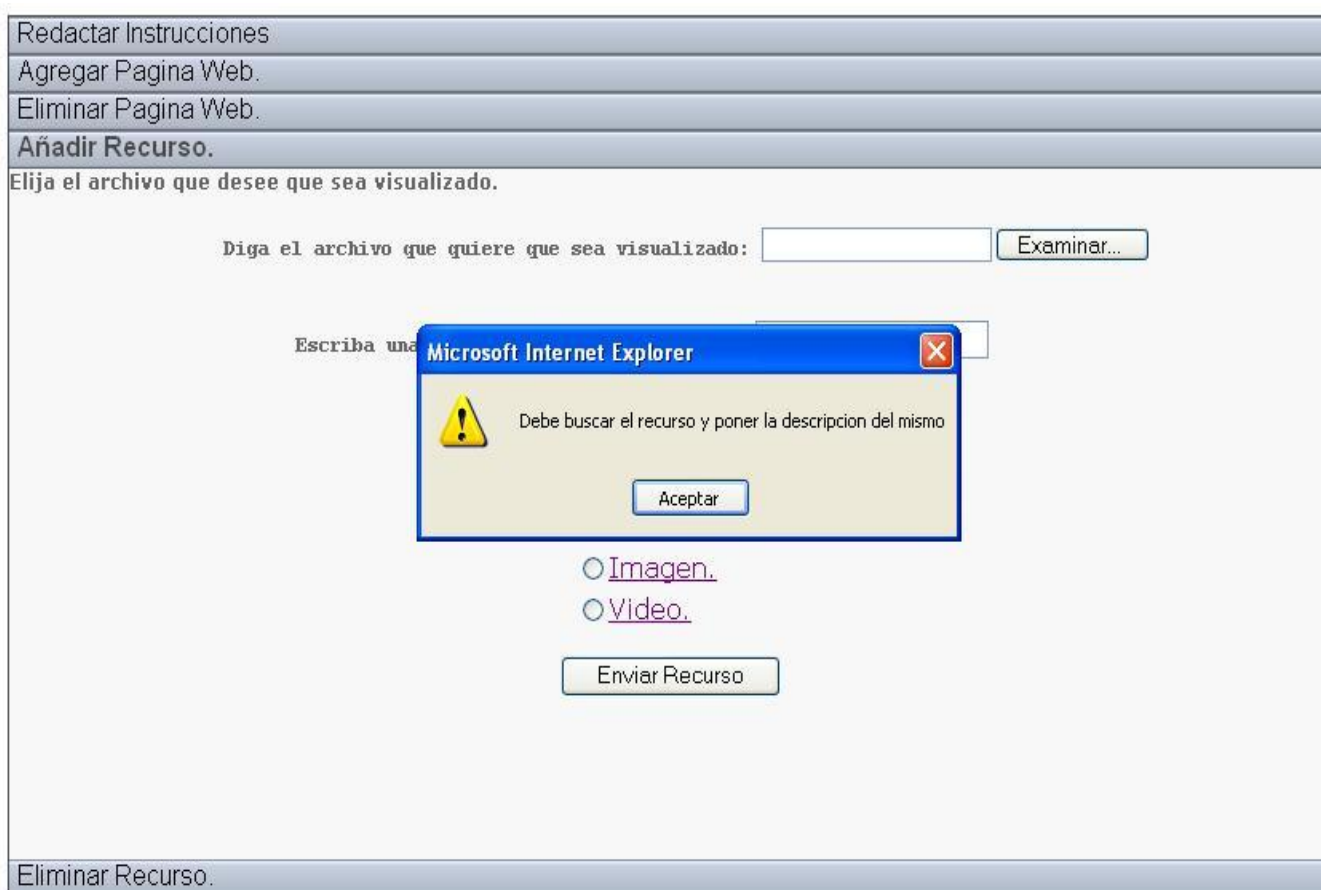


Figura 3.11 Tratamiento de excepciones Interfaz de profesor “Añadir archivo”.

Concepción general de la ayuda.

Para la ayuda se utilizan vínculos en palabras o frases claves que abren unas pequeñas ventanas que contienen información que ayuda al profesor a que se familiarice con la aplicación. Para continuar la navegación, después que es mostrada una ventana de ayuda, hay que cerrar dicha ventana impidiendo que usuarios que no estén familiarizados con el trabajo en una computadora se pierdan en la navegación de la aplicación Web.

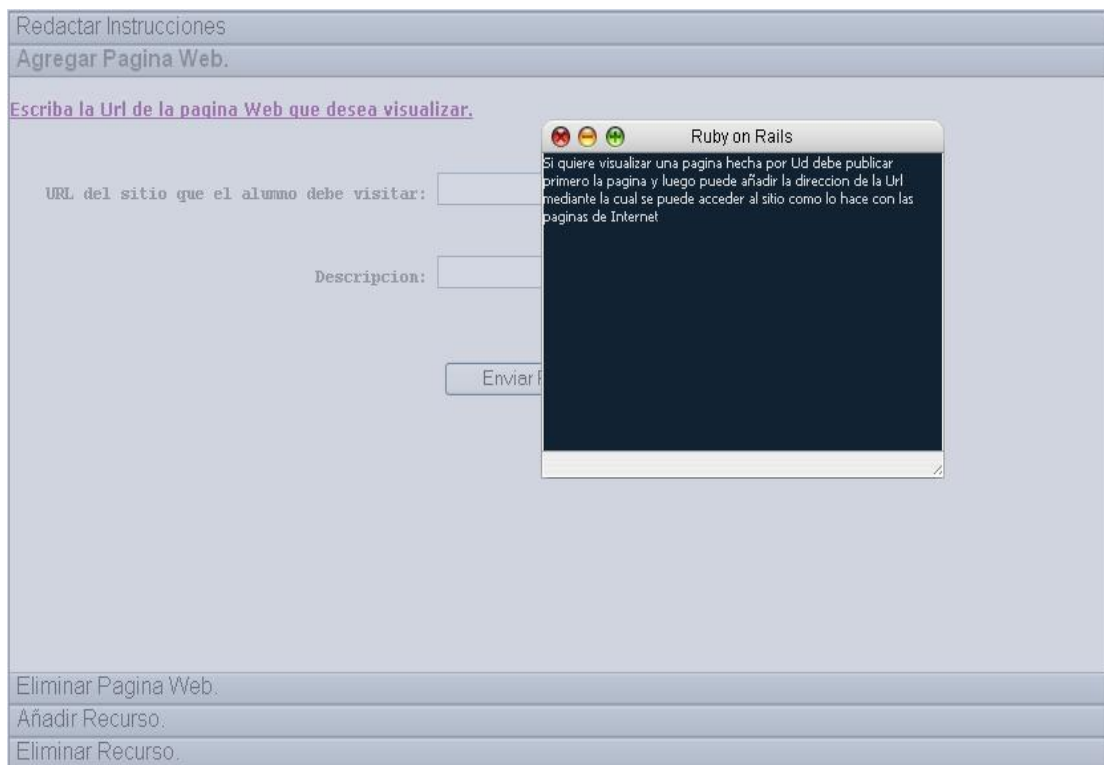


Figura 3.12: Ayuda.

Estándar de Codificación.

El estándar de codificación tiene como objetivo asegurar que el código de un programa de computación sea legible para poder ser entendido, utilizado y mejorado por todos los programadores que le interesen y lo deseen. Además asegura que el proceso de mantenimiento del software sea más sencillo.

Para el desarrollo de este trabajo se siguió un estándar de codificación que abarca los siguientes puntos:

- Los nombre de las variables locales comienzan con minúscula y al hacer asignaciones se deja un espacio a ambos lados del signo igual (=).
- Para los comentario se usan “// “ para un sola línea y “/*” para cuando son varias líneas.
- Para las estructuras de control (if , while) se deja un espacio entre la palabra clave y el paréntesis.
- El nombre de las clases y de las funciones son descriptivos, evitándose siempre el uso de abreviaturas.

Modelo de despliegue

Los Diagramas de Despliegue muestran la disposición física de los distintos nodos que componen un sistema y el reparto de los componentes sobre dichos nodos. La vista de despliegue representa la disposición de las instancias de componentes de ejecución en instancias de nodos conectados por enlaces de comunicación. Un nodo es un recurso de ejecución tal como una computadora, un dispositivo o memoria.

En el diagrama de despliegue se indica la situación física de los componentes lógicos desarrollados. Es decir se sitúa el software en el hardware que lo contiene. Cada Hardware se representa como un nodo.

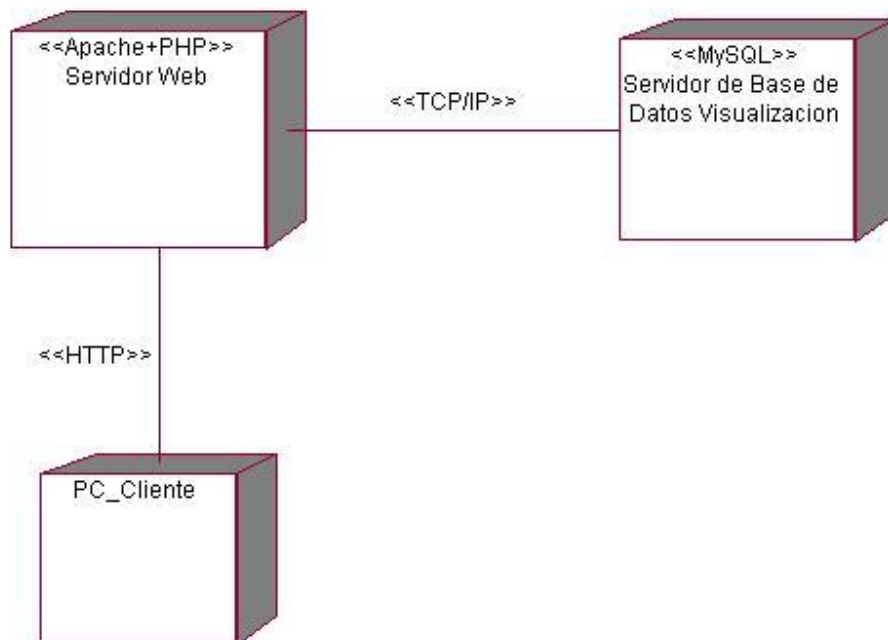


Figura 3.13: Diagrama de Despliegue.

Descripción del modelo de prueba.

Las pruebas que se les realizan al software constituyen un elemento crítico para garantizar la calidad de un software. Las pruebas no se realizan en una etapa específica del proyecto, sino que las pruebas deben ocurrir durante todo el ciclo de vida del mismo.

Concretamente para probar el código fuente de un software se utilizan las denominadas pruebas unitarias que tienen como objetivo encontrar errores en el código del programa. Las pruebas unitarias se clasifican en las pruebas de la caja negra y la prueba de la caja blanca.

La prueba de la caja negra se refiere a las pruebas que se llevan a cabo sobre la interfaz del software. O sea, los casos de prueba pretenden demostrar que las funciones del software son operativas, que la

entrada se acepta de forma adecuada y que se produce un resultado correcto, así como que la integridad de la información externa se mantiene. En esta prueba no importa lo que hace internamente el código, solo importan las entradas y las salidas.

En la prueba de la caja blanca se comprueba los caminos lógicos del software proponiendo casos de prueba que se ejerciten conjuntos específicos de condiciones. Se puede examinar el estado del programa en varios puntos para determinar si el estado real coinciden con el esperado o mencionado.

En el caso del **módulo de visualización de recursos** se realizan pruebas de la caja negra para los casos de usos principales, para probar que a las distintas entradas hechas por el usuario, a la hora de interactuar con el sistema, ocurren las salidas esperadas.

Casos de prueba

Caso de uso: Elaborar Instrucciones
Caso de prueba: Elaborar instrucciones, verificando que las instrucciones sea redactas.
Entrada: <ul style="list-style-type: none">• El profesor redacta las instrucciones.
Resultado: <ul style="list-style-type: none">• El sistema permite la entrada del texto con las instrucciones y envía al profesor a la página principal.
Condiciones: <ul style="list-style-type: none">• El profesor está ha entrado al sistema con ese rol.

Caso de uso: Elaborar Instrucciones
Caso de prueba: Elaborar instrucciones, verificando que las instrucciones sean redactas.
Entrada: <ul style="list-style-type: none">• El profesor deja el lugar donde tiene que redactar las instrucciones vacío.
Resultado: <ul style="list-style-type: none">• El sistema le comunica al profesor que debe redactar las instrucciones.

Condiciones:

- El profesor está ha entrado al sistema con ese rol.

Caso de uso: Gestionar Pagina Web (Adicionar)

Caso de prueba:

Adicionar una nueva página Web.

Entrada:

- El profesor entra el nombre y la descripción de una página Web.

Resultado:

- El sistema añade la página Web y le comunica al profesor que se ha añadido la página Web satisfactoriamente, enviándolo a la página principal.

Condiciones:

- El profesor está ha entrado al sistema con ese rol.

Caso de uso: Gestionar Pagina Web (Adicionar)

Caso de prueba:

Adicionar una nueva página Web.

Entrada:

- El profesor deja el campo del nombre o la descripción de la página vacíos.

Resultado:

- El sistema le comunica al profesor que debe llenar todos los campos

Condiciones:

- El profesor está ha entrado al sistema con ese rol.

Caso de uso: Gestionar Pagina Web (Eliminar)

Caso de prueba:

Eliminar una página Web.

Entrada:

- El profesor elije las páginas Web que desea eliminar.

Resultado: <ul style="list-style-type: none">• El sistema elimina las páginas Web y le comunica al profesor que se han eliminado las páginas Web satisfactoriamente, enviándolo a la página principal.
Condiciones: <ul style="list-style-type: none">• El profesor está ha entrado al sistema con ese rol.

Caso de uso: Gestionar Pagina Web (Eliminar)
Caso de prueba: Eliminar una página Web.
Entrada: <ul style="list-style-type: none">• El profesor no elije ninguna de las páginas Web.
Resultado: <ul style="list-style-type: none">• El sistema le comunica al profesor que tiene que elegir una de las páginas Web para poder eliminarla.
Condiciones: <ul style="list-style-type: none">• El profesor está ha entrado al sistema con ese rol.

Caso de uso: Gestionar Recurso (Adicionar)
Caso de prueba: Adicionar un recurso.
Entrada: <ul style="list-style-type: none">• El profesor entra el nombre y la descripción y el tipo de recurso.
Resultado: <ul style="list-style-type: none">• El sistema añade el recurso y le comunica al profesor que se ha añadido el recurso satisfactoriamente, enviándolo a la página principal.
Condiciones: <ul style="list-style-type: none">• El profesor está ha entrado al sistema con ese rol.

Caso de uso: Gestionar Recurso (Adicionar)

Caso de prueba: Adicionar un recurso.
Entrada: <ul style="list-style-type: none">• El profesor no entra el nombre o la descripción del recurso.
Resultado: <ul style="list-style-type: none">• El sistema le comunica al profesor que debe entrar el nombre y la descripción del recurso.
Condiciones: <ul style="list-style-type: none">• El profesor está ha entrado al sistema con ese rol.

Caso de uso: Gestionar Recurso (Adicionar)
Caso de prueba: Adicionar un recurso.
Entrada: <ul style="list-style-type: none">• El profesor no señala o señala mal el tipo de recurso que va a adicionar.
Resultado: <ul style="list-style-type: none">• El sistema le comunica al profesor que debe entrar el tipo del recurso correctamente.
Condiciones: <ul style="list-style-type: none">• El profesor está ha entrado al sistema con ese rol.

Caso de uso: Gestionar Recurso (Eliminar)
Caso de prueba: Eliminar un recurso.
Entrada: <ul style="list-style-type: none">• El profesor elije los recursos que desea eliminar.
Resultado: <ul style="list-style-type: none">• El sistema elimina los recursos y le comunica al profesor que se han eliminado los recursos satisfactoriamente, enviándolo a la página principal.
Condiciones:

- El profesor está ha entrado al sistema con ese rol.

Caso de uso: Gestionar Recurso (Eliminar)
Caso de prueba: Eliminar un recurso.
Entrada: <ul style="list-style-type: none">• El profesor no elije ninguno de los recursos.
Resultado: <ul style="list-style-type: none">• El sistema le comunica al profesor que tiene que elegir uno de los recursos para poder eliminarlo.
Condiciones: <ul style="list-style-type: none">• El profesor está ha entrado al sistema con ese rol.

Conclusiones

En el capítulo se construyó la propuesta de la solución, realizándose los diagramas de clases Web y los diagramas de secuencia, el diagrama de clases persistente y el modelo de datos. Se explica la selección de los patrones de diseño y de arquitectura, se analizaron los principios de la interfaz y los estándares de codificación. Y se realizaron distintas pruebas de la caja negra a los principales casos de uso.

CONCLUSIONES GENERALES

Con la realización de este trabajo se arribaron a las siguientes conclusiones:

- Se realizó un estudio de los principales recursos que sirven de apoyo a un profesor en el proceso de enseñanza-aprendizaje, llegando a la conclusión que los recursos que más se visualizan en prácticas de laboratorio virtuales son los documentos de texto, las páginas Web, las imágenes y los videos.
- Se realizó un estudio sobre como se realizan las visualizaciones de dichos recursos actualmente en las comunidades de aprendizaje mediante el cual se pudo definir que para lograr una buena interactividad era necesario la utilización de la tecnología AJAX y las librerías Rico y Prototype.
- Se llevó a cabo el diseño y la implementación de la primera versión de un módulo de visualización interactiva de recursos, que integrado a otros módulos, permite que un profesor pueda diseñar una secuencia de actividades que un alumno seguirá posteriormente dándole cumplimiento al objetivo propuesto en este trabajo.

RECOMENDACIONES

Luego de concluir con el presente trabajo, se recomienda:

- Continuar el estudio con el objetivo de encontrar nuevas funcionalidades para refinar e implementar una herramienta más completa y general.
- Continuar con el estudio de las posibilidades que brindan la tecnología AJAX para lograr una mayor interactividad tanto a la hora de añadir un recurso como a la hora de visualizar dicho recurso.
- Incorporar otros tipos de extensiones para los recursos seleccionados, así como la posibilidad de agregar otros tipos de recursos que sirvan para complementar los ya seleccionados.
- Analizar la integración de este módulo de visualización con otros módulos del sistema general.

REFERENCIAS BIBLIOGRÁFICAS

1. GARRETT, J. J. *Ajax: Un nuevo acercamiento a las aplicaciones web* de 26/2/2007]. Disponible en: <http://www.maestrosdelweb.com/editorial/ajax/>.
2. GRACIA., L. C. Y. J. *Conceptos básicos* de 15/2/2007]. Disponible en: <http://www.webestilo.com/javascript/js00.phtml>.
3. HERÍAS, F. A. C. *Propuesta de Portal de la Red de Laboratorios Virtuales y Remotos de CEA* de 16/1/2007]. Disponible en: http://209.85.165.104/search?q=cache:g6eYkgZojisEJ:dfists.ua.es/docenweb/Docs/PropuestaDePortal.pdf+definiendo+laboratorio+virtual+sobre+web&hl=es&ct=clnk&cd=4&gl=cu&lr=lang_es
4. IGNACIO COUPEAU, C. P. Y. J. U. *Introducción a SQL y ORACLE* de 10/2/2007]. Disponible en: http://www.unav.es/cti/manuales/Intro_SQL/intro_SQL.html.
5. M^a BEGOÑA ALFAGEME GONZÁLEZ, I. M. S. F. Y. J. V. M. *LA INTERACTIVIDAD: COMO CARACTERÍSTICA DE LA ENSEÑANZA MEDIANTE REDES*. de 12/1/2007]. Disponible en: <http://tecnologiaedu.us.es/edutec/paginas/58.html>
6. MAURA, V. G. *El profesor universitario: ¿un facilitador o un orientador en la educación de valores?* de 23/3/2007]. Disponible en: <http://www.oei.es/valores2/viviana.htm>.
7. PASCUALENA, M. L. S. *Organización y gestión de equipos para el desarrollo de contenidos educativos multimedia* de 20/1/2007]. Disponible en: http://reddigital.cnice.mec.es/6/Articulos/articulo_capitulo.php?articulo=4&capitulo=1.
8. RHEINGOLD, H. *The virtual community*. 1993. S., C. V. D. H. *¿Qué es el ASP?* de 17/2/2007]. Disponible en: <http://www.maestrosdelweb.com/editorial/aspintro/>

REFERENCIAS BIBLIOGRAFICAS

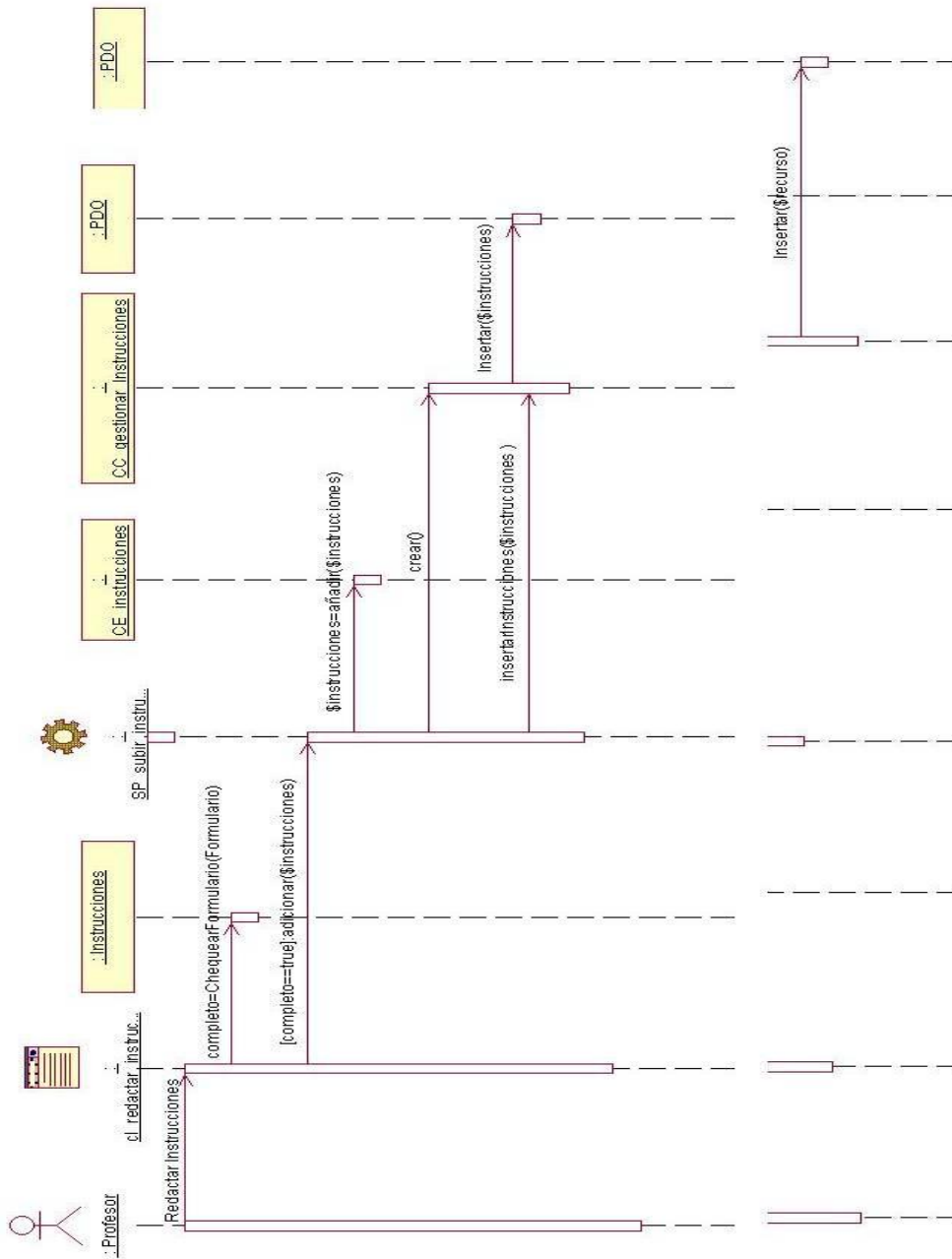
9. SOLANO, A. *SQL Server 2000* de 20/2/2007]. Disponible en:
http://www.netveloper.com/contenido2.aspx?IDC=64_0.
10. VERA, K. L. *Ingeniería de Software – RUP - UML* de 26/6/2007]. Disponible en:
<http://www.mmug.cl/articulos.php?id=287&tod=1>.

BIBLIOGRAFIA

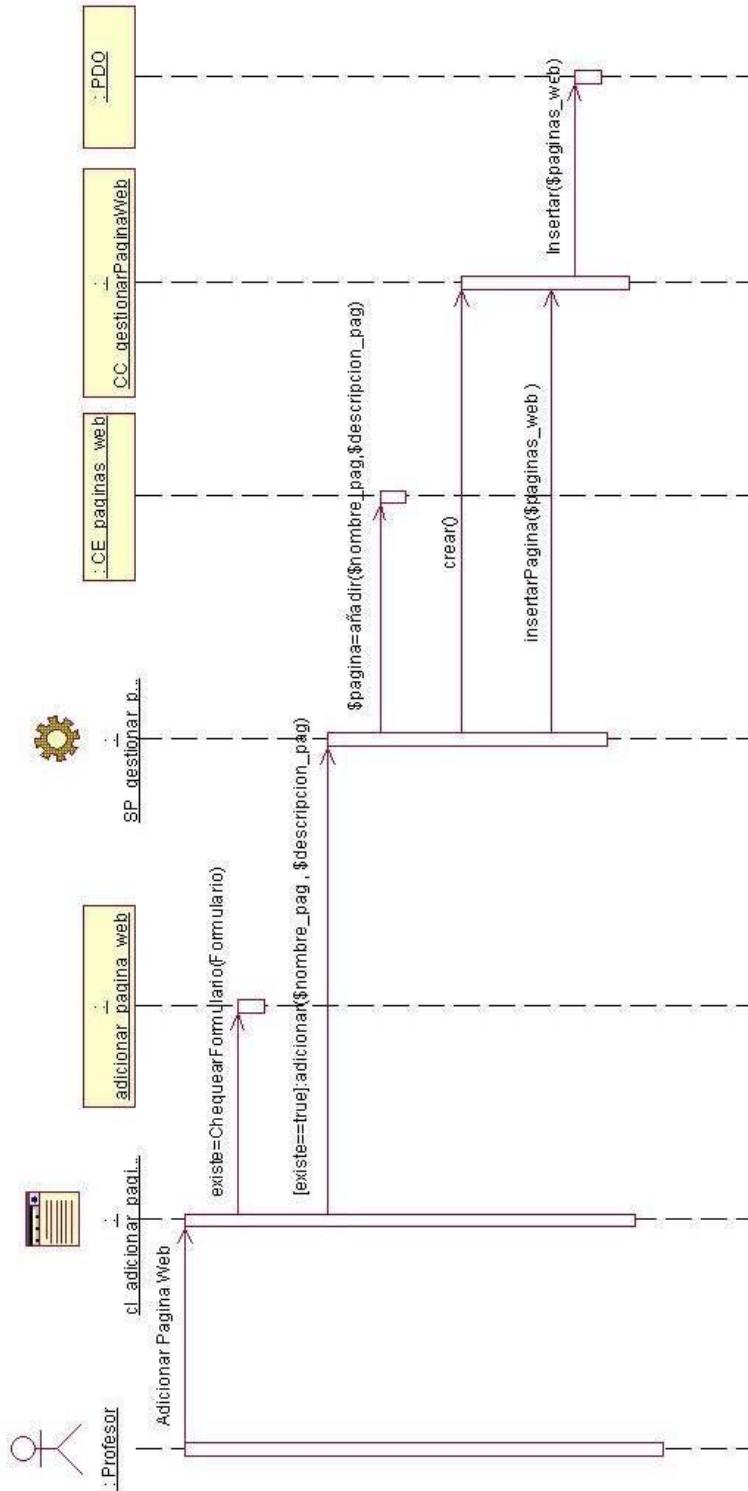
1. Gorga, Gladis; Madoz, Maria C.: Conversión de un Curso de Ingreso a las carreras de la Facultad de Informática de la UNLP en Modalidad Presencial a Modalidad Semipresencial.
<http://www.redespecialweb.org/ponencias3/GORGAMADOZ2.txt>(10/02/2007)
2. RGroup. e-Learning sincrónico o e-learning asincrónico. 2005
<http://www.3rgroup.org/elearning/art03.asp>. (12/02/2007)
3. Comunidades de aprendizaje capaces de crear y crecer en espacios.
http://www.impactalliance.org/ev_es.php?ID=2469_201&ID2=DO_TOPIC(16/02/2007)
4. Trejo Martínez, Janhil A.: Base de Datos.
<http://www.monografias.com/trabajos11/basda/basda.shtml> (26/04/2007).
5. ¿Qué es ASP.NET?
<http://www.zonagratis.com/microsoft/asp/aspnet.htm> (25-04-2007).
6. Metodología Parte II. Justificación y Metodología.
<http://onness.sourceforge.net/proyecto/html/ch05.html>(metodologiaXP) (15-05-2007).
7. Asegurando un servidor MySQL
<http://www.mysql-hispano.org/page.php?id=35&pag=2>(16-05-2007).
8. Restaurar un respaldo o backup de base de datos MySQL
<http://www.desarrolloweb.com/faq/restaurar-respaldo-base-datos-mysql.html>(16-05-2007).
9. Lams su integración con MOODLE
<http://moodle.org/login/index.php>(8-02-2007).

ANEXOS

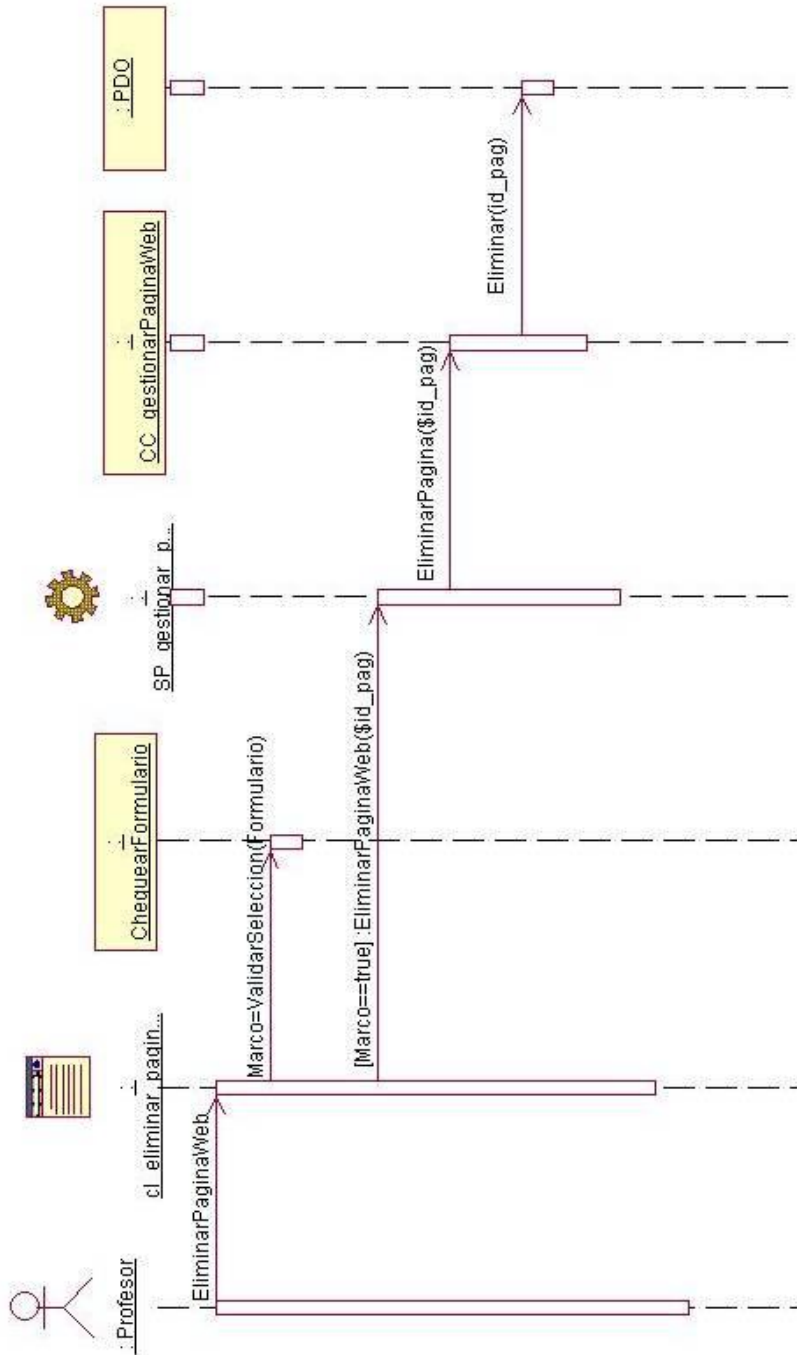
Anexo 1. Diagrama de secuencia del caso de uso "Elaborar Instrucciones".



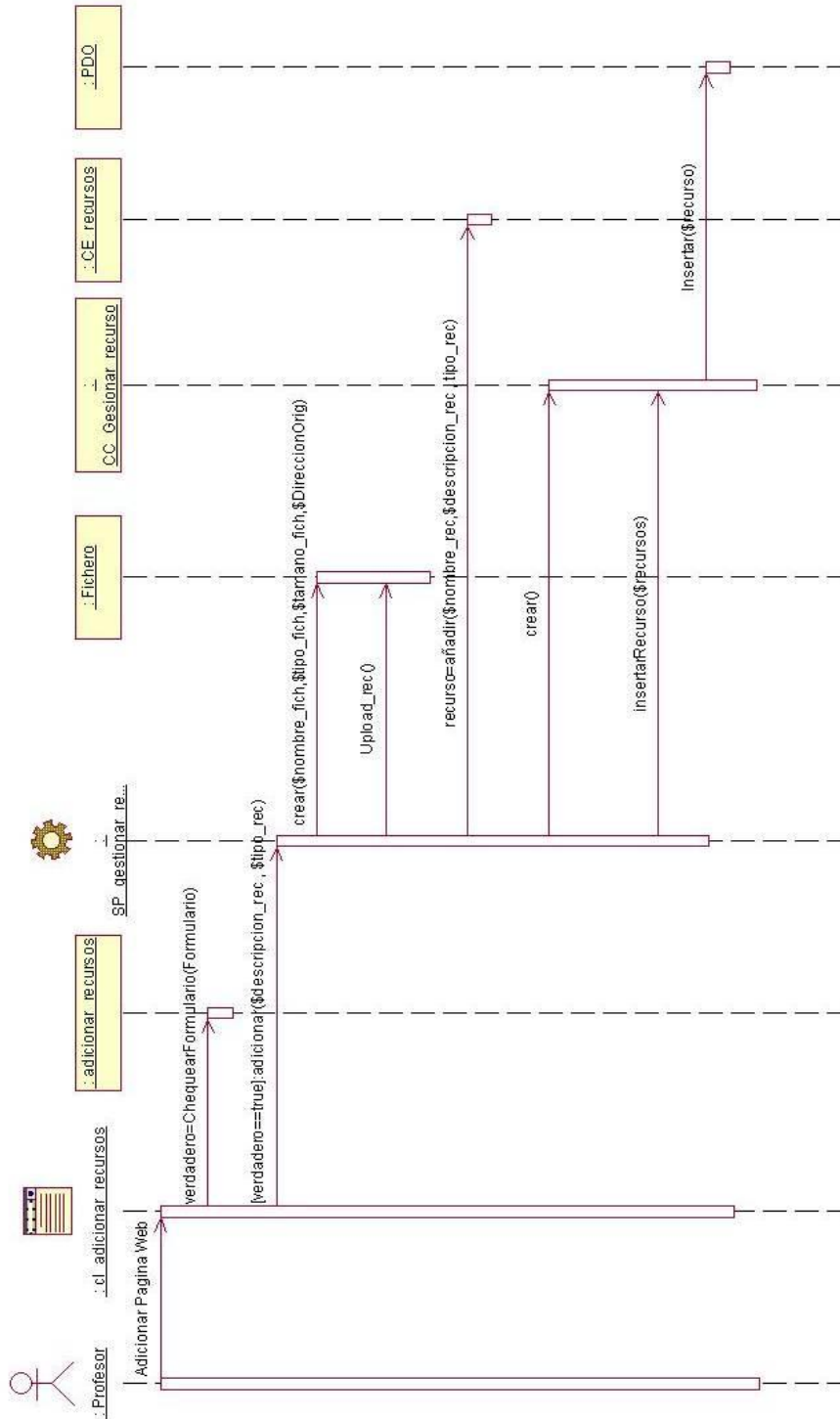
Anexo 2. Diagrama de secuencia del caso de uso “Gestionar Paginas Web”. Sección “Adicionar Página”



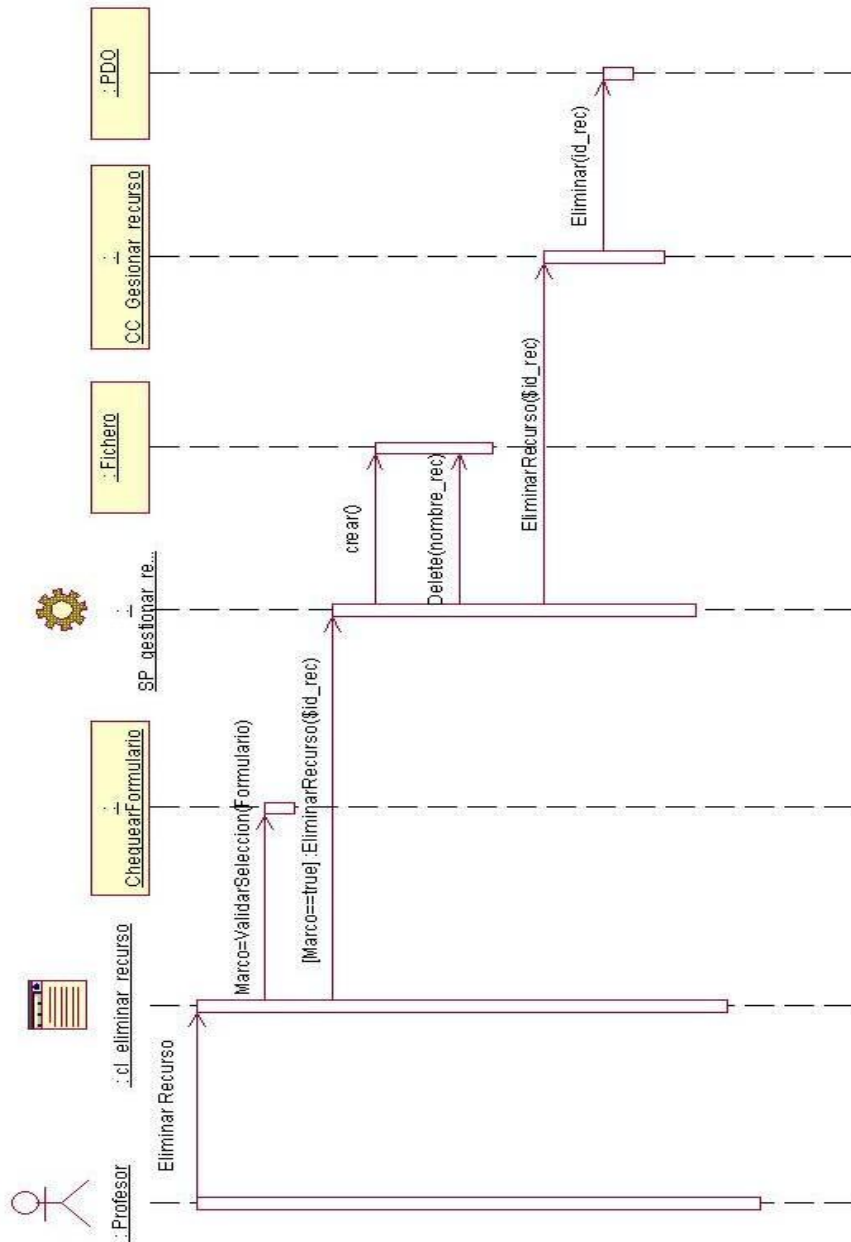
Anexo 3. Diagrama de secuencia del caso de uso “Gestionar Paginas Web”. Sección “Eliminar Página”



Anexo 4. Diagrama de secuencia del caso de uso “Gestionar Recurso”. Sección “Adicionar Recurso”



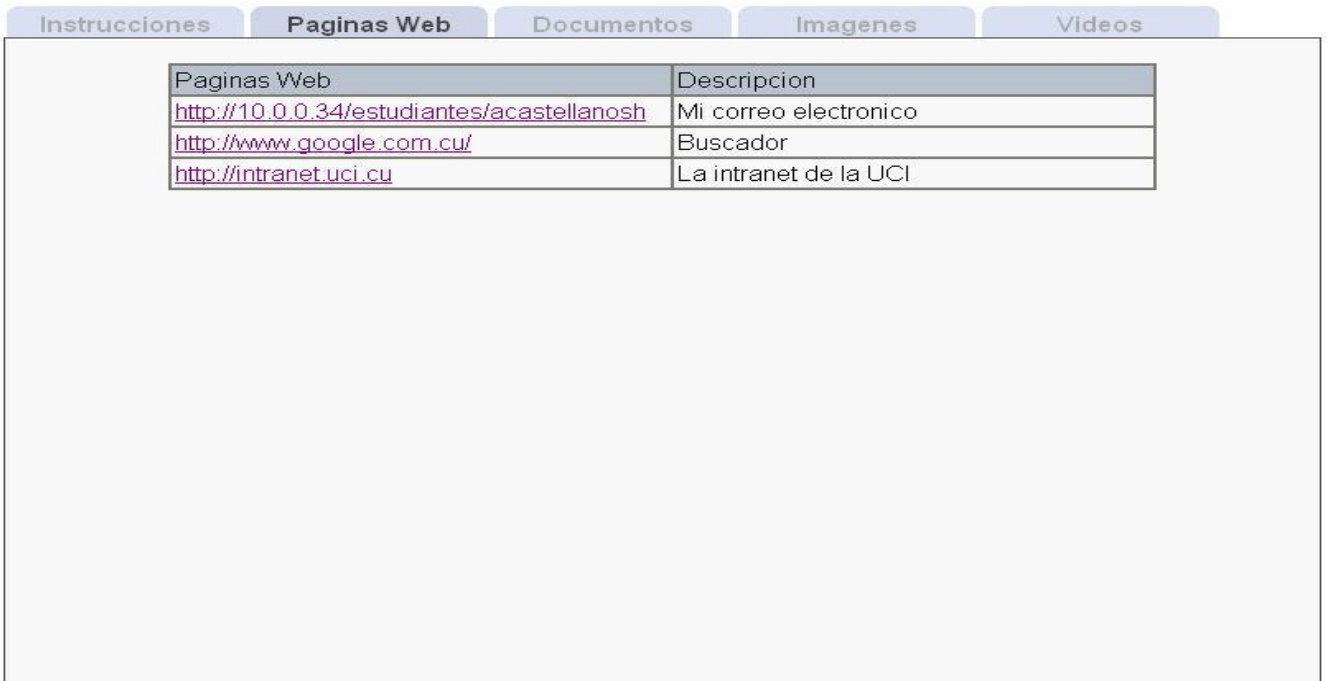
Anexo 5. Diagrama de secuencia del caso de uso “Gestionar Recurso”. Sección “Eliminar Recurso”



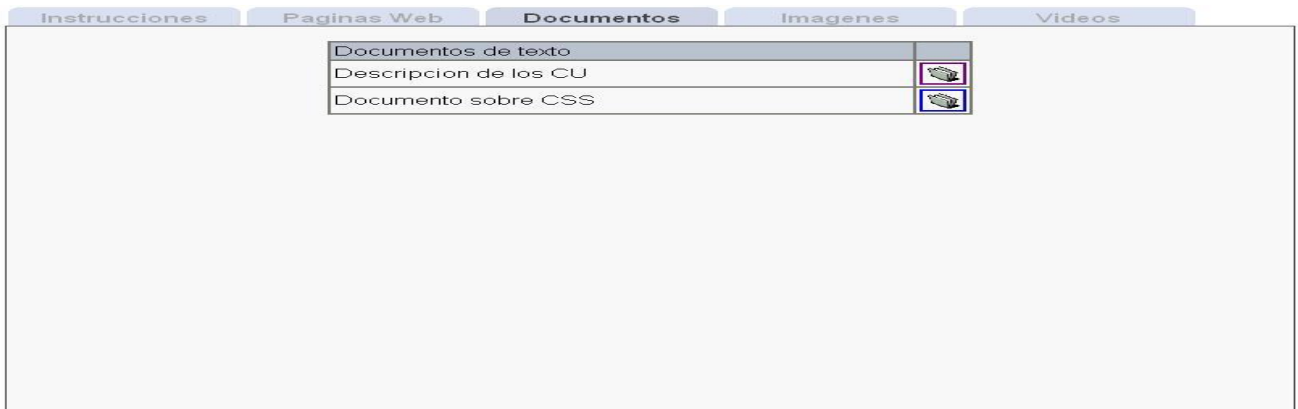
Anexo 6. Reporte en la Interfaz del profesor “Eliminar Página Web”.



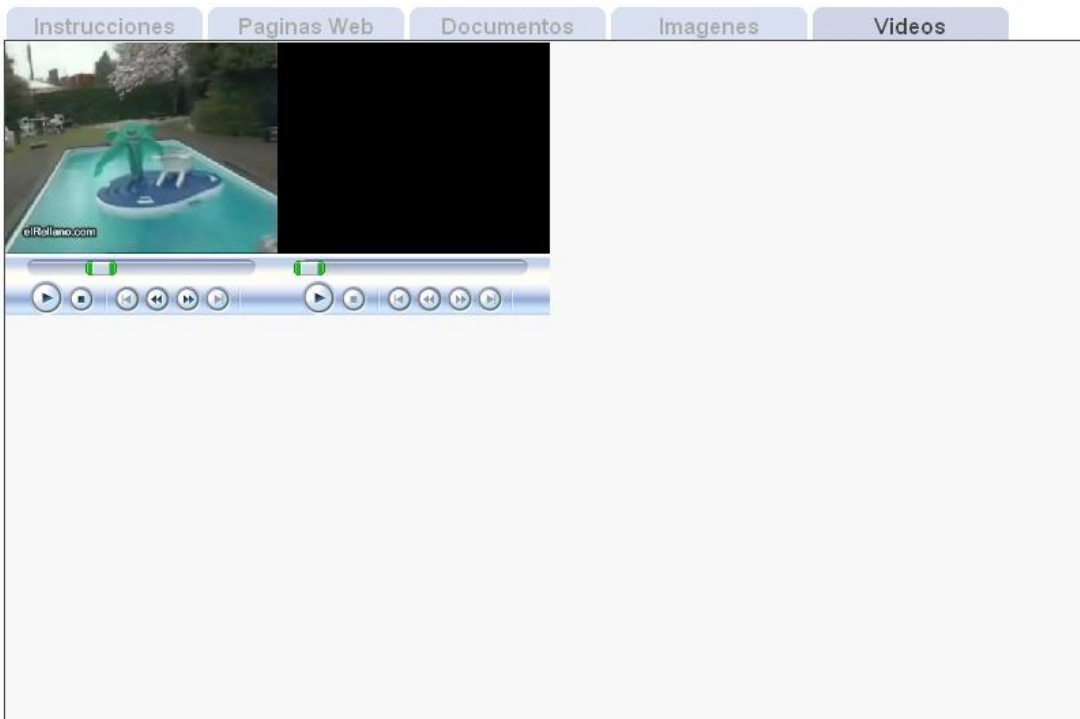
Anexo 7. Reporte en la Interfaz del estudiante “Mostrar Páginas Web”.



Anexo 8. Reporte en la Interfaz del estudiante “Mostrar documentos” .



Anexo 9. Reporte en la Interfaz del estudiante “Mostrar Videos” .



GLOSARIO DE TERMINOS

Apache: es el servidor Web, disponible sin costo alguno, distribuido con licencia Open Source. Es compatible con sistemas operativos basados en Unix y con Windows.

Aplicación: es el programa que el usuario activa para trabajar en el ordenador. Existen muchos programas de ordenador que pueden clasificarse como aplicación. Generalmente se les conoce como Software.

Clases persistentes: representan almacenamientos de datos que persistirán más allá de la ejecución el software.

Comunidad virtual: es aquella comunidad cuyos vínculos, interacciones y relaciones tienen lugar no en un espacio físico sino un espacio virtual.

CSS: Siglas en Inglés de Cascading Style Sheets (Hojas de Estilo en Cascada).

E-Learning: abarca al conjunto de las metodologías y estrategias de aprendizaje que emplean tecnología digital o informática para producir, transmitir, distribuir, y organizar conocimiento entre individuos, comunidades y organizaciones.

HTTP (Hypertext Transfer Protocol): protocolo de transmisión del hipertexto

Internet: red de computadoras alrededor de todo el mundo que comparten información unas con otras por medio de páginas o sitios.

Laboratorios Virtuales: aula que sirve como lugar de prácticas colaborativas.

Tecnologías de la información y de la comunicación (TIC): el concepto es empleado para designar lo relativo a la informática conectada a Internet y especialmente, el aspecto social de éstos, donde se agrupan un conjunto de aparatos necesarios para administrar la información, y especialmente los ordenadores y programas necesarios para convertirla, almacenarla, administrarla, transmitirla y encontrarla.

GLOSARIO DE TERMINOS

URL: Acrónimo de Universal Resource Locator (localizador universal de recursos), método de identificación de documentos o lugares en Internet, que se utiliza principalmente en World Wide Web (WWW). Un URL es una cadena de caracteres que identifica el tipo de documento, la computadora, el directorio y los subdirectorios en donde se encuentra el documento y su nombre.