

Facultad 8

TRABAJO DE DIPLOMA PARA OPTAR POR EL TÍTULO DE INGENIERO EN CIENCIAS INFORMÁTICAS

Título: Guía para la implementación de entornos interactivos sobre tecnologías libres que apoyen los cursos de segundo perfil en la facultad 8.

Autor(es): Anabel Pérez Avila, Anabel Pérez López

Tutor: Lic. Héctor Matías González

Entorno Virtual de Aprendizaje

Curso 2008-2009

moodle

Joomla!



DEDICATORIA

A nuestros padres y hermano por estar siempre presentes.

A nuestra familia por su apoyo y constante preocupación.

*A todos los amigos que han estado para brindarnos su ayuda
incondicional.*

*A nuestro invencible Comandante en Jefe por ser el principal precursor
de la Universidad de las Ciencias Informáticas.*

A todos los que nos han hecho lo que somos hoy.

AGRADECIMIENTOS

A Héctor por haber sido nuestro tutor, apoyarnos y guiarnos en el desarrollo de este Trabajo de Diploma.

Al tribunal por confiar en nosotras y ayudarnos a perfeccionarlo.

A todos aquellos profesores y estudiantes que de una forma u otra aportaron su granito de arena en nuestra formación como profesionales.

A mi compañera de tesis, por su responsabilidad y abnegación al trabajo.

A la Revolución, por seguir con paso firme y habernos dado la oportunidad de ser integrantes de su Tropa del Futuro. Y un agradecimiento especial a nuestro invencible Comandante en Jefe Fidel y a Raúl Castro, por guiarnos por el camino de la verdad y la justicia.

Anabel Pérez Avila

A mis padres que me han brindado todo su amor, dedicación y apoyo incondicional; por haber estado siempre a mi lado y confiar en mí. A mi querido hermano por su carácter tan especial, que sin importar nada siempre estaremos ligados con un lazo indestructible de cariño y admiración. Todo lo que soy hasta ahora y seré se lo debo a ellos, que me han inculcado el mejor de los ejemplos.

A toda mi enorme familia por estar siempre unidos, querernos y ayudarnos tanto.

A mis compañeros y amigos que nunca podré olvidar, con los cuales he compartido buenos y malos momentos durante estos cinco años, a los del aula, los de las fiestas y a los que ya no están.

A todos muchísimas gracias...

Anabel Pérez López

A mis padres que son la razón de mi ser y me han brindado todo su amor, sus enseñanzas, años de sacrificio, consejos, su confianza y apoyo incondicional.

A mi querido hermano, por ser mi alma gemela, compartir conmigo todos los momentos felices y difíciles que hemos tenido que afrontar en la vida, por brindarme todo su amor y apoyo para que pudiese realizar este trabajo.

A toda mi familia por el granito de arena que día a día pusieron en este trabajo, además de brindarme su apoyo incondicional.

A mi novio que vivo enamorada de él, por comprenderme y apoyarme en todo momento, aguantar mis malcriadeces, brindarme todo su cariño y amor en estos años de universidad. Por haber cambiado mi vida y hacerme reír cada día. A su familia por brindarme su amor, cariño y aceptarme como soy.

A mis compañeros y amigos que nunca podré olvidar, con los cuales he compartido buenos y malos momentos durante estos cinco años, haciendo que fueran los mejores de mi vida, a los del aula, los de las fiestas y a los que ya no están.

A todos los que me brindaron su apoyo y comprensión en estos años, para ellos va este trabajo.

DECLARACIÓN DE AUTORÍA

Declaramos ser Anabel Pérez López y Anabel Pérez Avila los únicos autores del presente Trabajo de Diploma, autorizamos a la facultad 8 de la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio y reconocemos a la misma los derechos patrimoniales de este trabajo, con carácter exclusivo.

Para que así conste firmamos la presente a los 12 días del mes de junio del año 2009.

Anabel Pérez López

Anabel Pérez Avila

Lic. Héctor Matías González

RESUMEN

La forma en que están diseñados los cursos de perfil en el Entorno Virtual de Aprendizaje (EVA), con que cuenta la Universidad de las Ciencias Informáticas (UCI), no fomenta la interacción del alumno con la herramienta. A partir de ello se han venido creando multimedias de apoyo a dichos cursos. Estas presentan el inconveniente de que se han desarrollado por separado, lo que ha ocasionado que presenten diferencias significativas en cuanto a implementación, diseño y contenido. Además, no se han publicado en un mismo sitio donde todos los usuarios puedan acceder a ellas y se han confeccionado mediante el empleo de herramientas propietarias. Por todos estos problemas antes mencionados se hace necesaria la elaboración de una guía para la implementación que permita la creación de un ambiente de aprendizaje interactivo, que combine aspectos de la tecnología Multimedia y el Entorno Virtual de Aprendizaje con que cuenta la UCI, además que sirva de apoyo a los cursos de segundo perfil y esté basado en herramientas libres.

Para dar cumplimiento al objetivo antes expuesto primeramente se abordaron conceptos fundamentales como *Software* Educativo, se describieron sus características y aplicación en Cuba, haciendo énfasis en la UCI. Después se realizó un levantamiento de las funcionalidades posibles de reutilizar para la creación del entorno interactivo, se seleccionaron y detallaron las herramientas para dar solución a los requerimientos identificados, para luego describir las funcionalidades más críticas. Se realizó la estructura de módulos y organización arquitectónica para la implementación del entorno. Finalmente se valoraron los resultados de la investigación por un grupo de expertos.

PALABRAS CLAVES: multimedias, entorno interactivo, funcionalidades, herramientas libres, organización arquitectónica.

ABSTRACT

The design of the courses within the Virtual Learning Environment (EVA - by its acronym in Spanish), used at the University of Informatics Sciences (UCI) does not promote student interaction. Because of this, multimedia support has been created for these courses. One of these multimedia disadvantages is that they have been developed separately, which has led to significant differences in terms of implementation, design and content. Besides, it has not been published on the same site where all users can access them and it has been developed using proprietary tools. For all these problems presented above, preparing a guide for implementation that allows the creation of an interactive learning environment, that combines aspects of Multimedia technology and Virtual Learning Environments available to the UCI, that additionally serves as support for the second profile courses and that is based on free tools, becomes a necessity.

In order to satisfy such objectives, concepts such as Educational Software, its characteristics and application in Cuba were described, with emphasis on the UCI. Subsequently, a survey was carried out to determine the functionalities with potential to be reused to create the interactive environment. The tools to solve the identified needs were selected and described, and then the most critical features were described.

The structure of the modules and architectural organization for the implementation of the environment were performed.

TABLA DE CONTENIDOS

ÍNDICE DE TABLAS.....	X
ÍNDICE DE FIGURAS.....	XI
INTRODUCCIÓN.....	1
CAPÍTULO I LA INFORMÁTICA Y EL SOFTWARE EN LA EDUCACIÓN SUPERIOR	
1.1. SOFTWARE EDUCATIVO	7
1.1.1 Tipo Algorítmico.....	7
Software Tutoriales	8
Software de Ejercitación y Práctica	8
1.1.2. Tipo Heurístico.....	8
Simuladores y Juegos Educativos	8
Micro mundos Exploratorios y Lenguaje Sintónico	9
Sistemas Expertos	9
1.2. TECNOLOGÍAS EN PROGRAMAS EDUCATIVOS	9
1.2.1. La Tecnología Multimedia	9
1.2.2. La Tecnología Hipermedia.....	10
1.3. INCLUSIÓN DEL SOFTWARE EDUCATIVO EN LAS UNIVERSIDADES	11
1.3.1. Alice	11
1.3.2. Scratch.....	11
1.3.3. Multimedias	11
1.3.4. Sistemas de Gestión de Aprendizaje (LMS).....	13
1.3.5. Sistemas de Gestión de Contenidos (CMS)	18
1.4. SOFTWARE EDUCATIVO EN LA UCI	19
1.5. CONCLUSIONES PARCIALES	21
CAPÍTULO II DESCRIPCIÓN DE LA PROPUESTA	
2.1. LEVANTAMIENTO DE LAS FUNCIONALIDADES DE LAS HERRAMIENTAS QUE SE PUEDEN REUTILIZAR ..	23
2.1.1. Moodle	23
2.1.2 Antiguas Multimedias	25
2.2. FUNCIONALIDADES DE LA SOLUCIÓN QUE SE PROPONE	27
2.3. SELECCIÓN DE LAS HERRAMIENTAS PARA CUBRIR LAS FUNCIONALIDADES	28
2.3.1. CMS Joomla	28
Plantillas	29

Componentes	30
Módulos	34
Plugin	34
2.3.2. LMS Moodle	35
2.4. DESCRIPCIÓN DE LAS FUNCIONALIDADES	36
Interfaz	36
Contenidos	38
Modulo Cuestionario	41
Glosario de Términos	49
Galerías de Imágenes	50
Galerías de Videos	51
Audio	52
Portabilidad	53
Apariencia	54
2.5. ESTRUCTURA DE MÓDULOS	57
2.6. ORGANIZACIÓN ARQUITECTÓNICA	57
2.7. CONCLUSIONES PARCIALES	60
<i>CAPÍTULO III ANÁLISIS DE LOS RESULTADOS</i>	
3.1. MÉTODO DE EXPERTOS	63
3.2. APLICACIÓN DEL MÉTODO DE EXPERTOS	63
3.3. CONCLUSIONES PARCIALES	67
<i>CONCLUSIONES GENERALES.....</i>	68
<i>RECOMENDACIONES.....</i>	69
<i>TRABAJOS CITADOS.....</i>	70
<i>BIBLIOGRAFÍA.....</i>	72
<i>ANEXOS.....</i>	75
<i>GLOSARIO DE TÉRMINOS.....</i>	83

ÍNDICE DE TABLAS

<i>Tabla 1. Comparación de ATutor/Moodle.....</i>	<i>17</i>
<i>Tabla 2. Archivos de las Plantillas en Joomla.</i>	<i>30</i>
<i>Tabla 3. Archivos para la parte de front-end de los Componentes en Joomla.</i>	<i>32</i>
<i>Tabla 4. Archivos de los Módulos en Moodle.....</i>	<i>35</i>
<i>Tabla 5. Peso promedio/Criterio.....</i>	<i>65</i>
<i>Tabla 6. Tabla para el cálculo de concordancia de Kendall y el estadígrafo Chi cuadrado.....</i>	<i>66</i>
<i>Tabla 7. Cálculo de P*c.</i>	<i>67</i>

ÍNDICE DE FIGURAS

<i>Figura 1. Estructura de directorios para la parte de front-end de los Componentes en Joomla..</i>	32
<i>Figura 2. Estructura de directorios para la parte de back-end de los Componentes en Joomla..</i>	33
<i>Figura 3. Estructura del directorio del Módulo Cuestionario.</i>	43
<i>Figura 4. Estructura parcial del motor de preguntas question.</i>	46
<i>Figura 5. Estructura del directorio principal de Joomla.</i>	58
<i>Figura 6. Estructura de la plantilla.</i>	59

INTRODUCCIÓN

Las nuevas tecnologías están promoviendo una nueva visión del conocimiento y del aprendizaje (1). A través de los años y con el desarrollo de la ciencia y las tecnologías se ha hecho necesario que la educación evolucione y se introduzcan innovaciones metodológicas, nuevas técnicas, vías y herramientas complementarias a la formación de enseñanza tradicional, como señala Martínez, "en los procesos de enseñanza-aprendizaje, como prácticamente en la totalidad de los procesos de comunicación, pueden darse diferentes situaciones espacio-temporales, tanto en la relación profesor-alumno, como en relación a los contenidos" (2).

Las aulas virtuales y la educación a distancia (*e-learning*) a través de redes informáticas, son nuevas formas de difundir los conocimientos a diversos sectores de la sociedad. Los sistemas de comunicación mediados por ordenadores, facilitan el acceso a la información para aquellas personas con dificultades para asistir regularmente a las instituciones educativas presenciales debido a sus obligaciones laborales, familiares o personales y para aquellas que cuentan con la tecnología que le permite usarla como apoyo a sus estudios.

En la educación a distancia se han incorporado entornos de aprendizaje interactivos que facilitan aún más la diversidad, la accesibilidad y la difusión de la educación, para así captar el interés y la atención del estudiante, estimulando su participación que con otros medios no sería posible conseguir. Dentro de estos entornos se encuentran las multimedias, que no son más que la integración de texto, grafismo, imagen real fija, imagen animada, vídeo y audio (3), que al combinarlas adecuadamente mejoran la atención, la comprensión y el aprendizaje, pues se acercan algo más a la manera habitual en que se comunican las personas, empleando varios sentidos para comprender un mismo objeto o concepto.

En Cuba se han incorporado también estas nuevas técnicas de aprendizaje; promovidas por el uso masivo de las tecnologías de la información y las telecomunicaciones, que se han insertado desde la enseñanza primaria hasta la superior por medio de computadoras personales, televisores y reproductores de videos. En la enseñanza superior y principalmente en la Universidad de las Ciencias Informáticas (UCI) existe un mayor aprovechamiento de estos recursos debido a las propias características de auto aprendizaje que exige este nivel educativo.

El país trabaja fuertemente en el desarrollo de la Industria Cubana del *Software*, de manera que se cubran las necesidades nacionales y se convierta en fuente importante de generación de divisas. Esta

industria no puede estar ajena a la revolución del *Software* Libre, por lo que tiene que estar preparada en esta tecnología, debe crearse la base de desarrolladores capaces de responder a las demandas nacionales e internacionales en esta esfera y es por ello que en la actualidad se encuentra inmerso en la migración de los proyectos ya realizados, a *Software* Libre, evitando con esto la dependencia tecnológica, permitiendo la elaboración de riquezas y recursos nacionales, que quedarían dentro del país. Además da la libertad de estudiar su código fuente, adaptándolo acorde a las necesidades y reutilizar funcionalidades ya existentes en función de los productos que se quieran elaborar.

La UCI tiene como principal característica el estar encaminada a promover esta producción de *software*, es por eso que los estudiantes deben dedicar tiempo de sus estudios a los proyectos productivos en los que se encuentran. Para apoyar la docencia se cuenta con un Entorno Virtual de Aprendizaje (EVA), sencillo, intuitivo y amigable que brinda oportunidades de comunicación y colaboración para la construcción de conocimientos en un ambiente rico en información.

El EVA con que cuenta la universidad está basado en el Sistema de Gestión de Aprendizaje (*LMS Moodle*, *software* diseñado para ayudar a los educadores a crear cursos en línea de alta calidad. Entre sus principales ventajas se pueden nombrar que se distribuye gratuitamente como *Software* Libre, es accesible desde cualquier computadora que posea un navegador Web, permite la importación y exportación de datos y está preparado para migrar desde y hacia cualquier plataforma educativa.

El EVA en la universidad tributa a la formación del claustro de profesores y al aprendizaje de materias necesarias para formar Ingenieros Informáticos. Dentro de estas materias se encuentran la categoría de formación de pregrado; donde están las asignaturas obligatorias que deben cursar los estudiantes en el transcurso de su carrera; y además los cursos de perfil que son los que se deben pasar para acreditarse segundos perfiles que varían dependiendo del perfil de la facultad; en el caso de la facultad 8 cuenta con *Software* Educativo y Multimedia, Gestión de *Software* y Calidad de *Software*.

La forma en que están diseñados los cursos de perfil en el EVA debe fomentar la interacción del alumno con la herramienta, los estudiantes tienen que estar muy motivados para la lectura de todos los documentos que se encuentran adjuntos, es por ello que en la facultad 8 se han venido creando multimedias de apoyo a los mismos, que promueven la interacción y motivación del que lo recibe, así como del profesor que lo imparte, que la crea a partir de sus necesidades y conocimientos. En esta facultad el desarrollo de estas multimedias es mayor que en las demás al contar con el perfil de *Software* educativo y Multimedia.

Dentro de las multimedias creadas existen algunas que al ser ideadas por los propios estudiantes llegan a generar contenidos poco claros que quizás sean muy llamativos, pero que no cumplen con el objetivo sustancial, que es facilitar el conocimiento del alumno. Además, se han desarrollado de manera aislada provocando que presenten diferencias significativas en cuanto a su implementación, diseños y contenido. Tampoco se ha logrado su centralización en un mismo sitio para que sea accesible a todos.

Otro problema que presentan estas multimedias, es que se han confeccionado mediante el empleo de herramientas propietarias, por lo cual se hace necesaria su migración a *Software Libre*, pues así pueden seguir siendo usadas en un futuro, dado que cualquier técnico informático puede continuar desarrollándola, mejorándola o adaptándola.

Conociendo la situación problemática que se plantea, se identificó el siguiente **problema científico**: ¿Cómo estandarizar la producción de *Software* educativos de apoyo a los cursos de segundo perfil de la facultad 8, haciendo uso exclusivo de herramientas y tecnologías libres?

Para darle solución al problema científico mencionado se planteó como **objetivo general**: Elaborar una guía que permita la implementación de un entorno de aprendizaje interactivo sobre tecnologías libres que apoyen los cursos de segundo perfil en la facultad 8.

Conjuntamente se plantearon los siguientes **objetivos específicos**:

- ✓ Realizar el estudio de las funcionalidades de las multimedias de apoyo a los cursos realizadas hasta el momento en la facultad 8 y del EVA disponible en la UCI.
- ✓ Seleccionar las herramientas y lenguajes que puedan ser utilizados en la creación del entorno y respondan a las funcionalidades seleccionadas.
- ✓ Proponer una arquitectura que sirva de base a la implementación de los materiales de apoyo a los cursos de perfil.
- ✓ Valorar los resultados de la investigación.

Se traza como **idea a defender**: Con la propuesta de la guía que permita la implementación de un entorno interactivo sobre tecnologías libres que apoyen los cursos de segundo perfil en la facultad 8, se podrán estandarizar los contenidos a impartir, permitir la incorporación de nuevos cursos, libertad

de licencias de todos los recursos a utilizar y su implementación y facilitar la interacción de los estudiantes con la herramienta.

El problema descrito anteriormente genera como **objeto de estudio**: Herramientas de apoyo al proceso de enseñanza-aprendizaje de cursos a distancia y como **campo de acción**: Proceso de desarrollo de una plataforma de apoyo a los cursos de segundo perfil en la facultad 8.

La investigación científica se guió por el conjunto de **tareas** que se describen a continuación para darle cumplimiento a los objetivos tanto generales como específicos planteados:

1. Recopilar la mayor cantidad de multimedias realizadas para apoyar los cursos optativos.
2. Realizar el estudio de las multimedias de apoyo realizadas hasta el momento en la facultad 8.
3. Realizar el estudio de las funcionalidades que brinda el EVA disponible en la UCI.
4. Identificar las funcionalidades que traen consigo problemas técnicos al implementar las multimedias sobre la Web.
5. Estudiar el estado de arte de las herramientas para la construcción de productos colaborativos y el montaje de cursos a distancia.
6. Seleccionar las herramientas, lenguajes y tecnologías a utilizar de acuerdo a los parámetros que se definen para su comparación.
7. Describir el proceso de desarrollo de las funcionalidades mediante el uso de las herramientas seleccionadas.
8. Analizar los módulos más comunes que se proponen en la construcción de cursos a distancia, así como su estructura.
9. Diseñar una estructura de módulos, capaz de responder a las necesidades de los cursos de segundo perfil y que cumpla un estándar adecuado basado en elementos del diseño instruccional y demás campos que ayudan a la correcta elaboración de *software* de este tipo.
10. Proponer una organización arquitectónica de la forma en que se deba implementar el producto.
11. Valorar los resultados de la investigación mediante la realización de encuestas a expertos.

El presente trabajo estará estructurado de la siguiente manera:

Capítulo 1: La informática y el *Software* en la Educación Superior

Este primer capítulo aborda acerca de la Fundamentación Teórica, en la cual se hace una conceptualización de lo que se entiende por *Software* Educativo, se estudia el estado del tema a nivel internacional, se hace mención a las características y la aplicación en Cuba y fundamentalmente en la UCI.

Capítulo 2: Descripción de la Propuesta

En el presente capítulo se realiza un levantamiento de las funcionalidades del *Moodle* presente en la UCI y las antiguas multimedias creadas para el apoyo de los cursos de segundo perfil, se seleccionan las que pueden presentar problemas técnicos a la hora de implementarlas sobre la Web, para luego escoger las herramientas que las puedan solucionar. Además se describen las funcionalidades de la solución propuesta y para concluir el capítulo se plantea la estructura de los módulos y una organización arquitectónica de la forma en que se deba implementar el producto.

Capítulo 3: Análisis de los Resultados

En este último capítulo se analizan los resultados de la investigación realizada; mediante una selección de expertos que emitieron su criterio.

Capítulo I

La informática y el *Software* en la
Educación Superior

En la actualidad y con el incremento de las tecnologías, la educación ha incorporado nuevas técnicas como herramientas pedagógicas, para que puedan ser empleadas por estudiantes y profesores. A partir de esto han surgido conceptos como Informática Educativa y *Software Educativo*.

Se entiende como *Informática Educativa*, al arte de enseñar a los alumnos y personas en general utilizando como herramienta fundamental el computador, el cual se conecta a una gran red mundial que es *Internet*. También a través de ella se pueden extraer recursos de mucho aporte para complementarlos con nuestra educación, por ende la informática es la nueva enseñanza del futuro presente (4).

Software Educativo se puede definir como cualquier programa computacional cuyas características estructurales y funcionales sirvan de apoyo al proceso de enseñar, aprender y administrar (5).

1.1. SOFTWARE EDUCATIVO

Todos los materiales pueden ser abordados por los programas educativos (matemáticas, idiomas,...) y de muy diversas formas (mediante preguntas, proporcionando contenidos, motivando a la investigación) brindando un ambiente de trabajo más sensible que individualiza la labor de los estudiantes y siendo más rico en posibilidades de interacción; pero todos comparten cinco **características esenciales**:

- ✓ Son materiales elaborados con una **finalidad didáctica**.
- ✓ **Utilizan el ordenador** como soporte en el que los alumnos realizan las actividades que ellos proponen.
- ✓ **Son interactivos**, contestan inmediatamente las acciones de los estudiantes y permiten un diálogo y un intercambio de informaciones con el ordenador.
- ✓ **Individualizan el trabajo** de los estudiantes, pues se adaptan al ritmo de trabajo de cada uno y pueden adecuar sus actividades según las actuaciones de los alumnos.
- ✓ **Son fáciles de usar**. Los conocimientos informáticos necesarios para utilizar la mayoría de estos programas son mínimos, aunque cada programa tiene unas reglas de funcionamiento que son necesarias conocer (6).

El *Software Educativo* se puede ser clasificado de tipo Algorítmico o Heurístico:

1.1.1. Tipo Algorítmico

Predomina el aprendizaje vía transmisión de conocimiento, desde quien sabe, hacia quien lo desea aprender y donde el diseñador se encarga de encapsular secuencias bien diseñadas de actividades

de aprendizaje que conducen al interesado desde donde está hasta donde se desea llegar. Dentro de este tipo se encuentran (7):

Software Tutoriales

Incluye cuatro fases que deben formar parte de todo proceso de enseñanza-aprendizaje: La fase introductoria, en la que se genera la motivación, se centra la atención y se favorece la percepción selectiva de lo que se desea que el usuario aprenda. La fase de orientación inicial, en esta se da la codificación, almacenaje y retención de lo aprendido. La fase de aplicación, en la cual hay evocación y transferencia de lo aprendido y por último la fase de retroalimentación, donde se demuestra lo aprendido, se ofrece retroinformación y esfuerzo o refuerzo (7).

Software de Ejercitación y Práctica

Refuerzan las dos fases finales del proceso de instrucción: aplicación y retroalimentación. Se parte de la base que el usuario tiene un conocimiento previo del tema relacionado con el *software* final. Donde el *software* le servirá para probar sus destrezas y conocimientos adquiridos previamente. Estos sistemas sirven como motivación y refuerzo para el usuario (7).

1.1.2. Tipo Heurístico

Predomina el aprendizaje experimental y por descubrimiento, donde el diseñador crea ambientes ricos en situaciones que el usuario debe explorar conjeturablemente. El usuario debe llegar al conocimiento a partir de experiencias y creando sus propios modelos de pensamiento. Dentro de este tipo se encuentran (7):

Simuladores y Juegos Educativos

Ambos poseen la cualidad de apoyar el aprendizaje de tipo experimental conjetural, como base para lograr aprendizaje por descubrimiento. El usuario resuelve problemas, aprende procedimientos, llega a entender las características de los fenómenos y cómo controlarlos o aprende qué acciones tomar en diferentes circunstancias. Lo esencial en ambos casos es que el usuario es un agente necesariamente activo que, además de participar en la situación debe continuamente procesar la información que se le proporciona en forma de situación problemática, condiciones de ejecución y resultado (7).

Micro mundos Exploratorios y Lenguaje Sintónico

Una forma particular de interactuar con micro mundos es haciéndolos con ayuda de un lenguaje de computación, en particular si es de tipo sinfónico con sus instrucciones y que se puede usar naturalmente para interactuar con un micro mundo en el que los comandos sean aplicables (7).

Sistemas Expertos

Capaces de representar y razonar acerca de algún dominio rico en conocimientos, con el ánimo de resolver problemas y dar consejos a quienes no son expertos en la materia. Además, de demostrar gran capacidad de desempeño en términos de velocidad, precisión y exactitud, tiene como contenido un dominio de conocimientos que requiere gran cantidad de experiencia humana, no solo principios o reglas de alto nivel y que es capaz de hallar o juzgar la solución de algo, explicando o justificando lo que haya o lo que juzgue; de modo que es capaz de convencer al usuario que su razonamiento es correcto (7).

1.2. TECNOLOGÍAS EN PROGRAMAS EDUCATIVOS

Conjuntamente al desarrollo de estos tipos de *software* se han generado nuevas tecnologías en los programas educativos: multimedia e hipermedia.

1.2.1. La Tecnología Multimedia

La tecnología multimedia aparece en el año 1984, cuando la *Apple Computer* lanza la primera variante de Macintosh, la cual tenía amplias capacidades de reproducción de sonidos; apoyándose en su sistema operativo, propicio para el diseño gráfico y la edición. En 1987, con los videojuegos, los avances en las Tecnologías de la Información y las Comunicaciones (TIC) y el desarrollo por la *Philips* del Disco Compacto (CD), aumentan las posibilidades para el desarrollo de la tecnología multimedia. En 1992, se presenta en la feria "*Consumer Electronic Show*" (CES) de Las Vegas, el CD multiusos, que luego dio paso a la televisión interactiva. En 1993, el nuevo término de multimedia, obliga a una revisión de los conceptos que permiten el funcionamiento de la nueva tecnología, buscando el desarrollo de estándares para lograr la uniformidad en el avance tecnológico.

El concepto *multimedia* engloba diferentes maneras de expresión como el video animado, imágenes fijas, texto y sonido que pueden ser impartidas para transmitir un significado. También encapsula la maquinaria utilizada para almacenar, editar, proyectar y transmitir los datos, que son la materia prima de las ideas (8). Entre algunas de las características que se le atribuyen se mencionan:

- ✓ Interactividad
- ✓ Dinamismo
- ✓ Flexibilidad
- ✓ Multidimensional
- ✓ Funcionalidad
- ✓ Modular la información
- ✓ Acceso Multiusuario

1.2.2. La Tecnología Hipermedia

El término *hipermedia* surge como resultado de la fusión de dos tecnologías, el *hipertexto* y la *multimedia*, la definición de este último fue abordada anteriormente. El *hipertexto* es una tecnología que organiza una base de información en bloques distintos de contenidos, conectados a través de una serie de enlaces cuya activación o selección provoca la recuperación de información (9).

La *hipermedia* es la tecnología que permite estructurar la información de una manera no-secuencial, a través de nodos interconectados por enlaces. La información presentada en estos nodos puede integrar diferentes medios: texto, sonido, gráficos y otros (10). Los sistemas de *hipermedia*, por tanto, se pueden entender como "Organización de información textual, visual gráfica y sonora a través de vínculos que crean asociaciones entre información relacionada dentro del sistema" (9).

Los términos *Hipermedia* e *Hipertexto* son en ocasiones motivo de confusión, se pueden mencionar características propias del primero:

- ✓ Utiliza el modo de escritura *HTML*¹.
- ✓ Textos hipermediales no lineales sino multisequenciales.
- ✓ La información es presentada de manera multisensorial.
- ✓ Múltiples rutas para encontrar la información requerida.
- ✓ Pueden ser imagen, sonido o video.
- ✓ Interfaz audio-visual, basada en iconos, que pudiesen animarse.

¹**HTML**, (*Hypertext Markup Language*), en español Lenguaje de Marcas de Hipertexto.

1.3. INCLUSIÓN DEL SOFTWARE EDUCATIVO EN LAS UNIVERSIDADES

La continua evolución de las tecnologías, la sociedad de la información, la tendencia a la comercialización del conocimiento y la demanda de sistemas de enseñanza-aprendizaje más flexibles y accesibles a los que pueda incorporarse cualquier ciudadano a lo largo de la vida; han provocado que las instituciones de educación superior apuesten decididamente por el empleo de las TIC. Respondiendo a estos cambios y demandas, las universidades están cambiando con gran frecuencia al uso de *Internet* o *Intranets* para distribuir cursos a estudiantes a distancia y mejorar el *software* educativo que se distribuye en ella, con el fin de incrementar su uso. En la educación superior se han insertado diversos *software* educativos, dentro de los que se encuentran:

1.3.1. Alice

Innovador entorno de programación 3D que facilita la creación de una animación. Permite a los estudiantes aprender los conceptos fundamentales de programación sencillos para la creación de videojuegos y películas animadas, además de crear programas para animar los objetos empleando lenguajes de programación orientados a la producción, tales como *Java*², *C + +*³ y *C #*⁴ (11).

1.3.2. Scratch

Orientado a la programación, la idea es que los alumnos puedan hacer sus programas de una forma visual e intuitiva, mediante bloques que se arrastran y combinan para realizar ciertas acciones predefinidas: escribir texto en la pantalla, reproducir sonidos y otras. Con este *software* se espera que los estudiantes comiencen a introducirse en el mundo de la programación (12).

1.3.3. Multimedia

Presentan numerosas funciones y ventajas que se aprovechan en la educación. Por ejemplo su función innovadora, motivadora, estructuradora de la realidad e informativa. Con respecto a las ventajas, se dice que las multimedia fomentan la variabilidad metodológica, favorecen la

² **Java**, lenguaje de programación de alto nivel, orientado a objetos.

³ **C + +**, lenguaje de programación que permitan la manipulación de objetos, se caracteriza por ser multiparadigma e híbrido.

⁴ **C #**, lenguaje de programación orientado a objetos, su sintaxis básica se deriva de C/C++ y utiliza el modelo de objetos de la plataforma.NET, la cual es similar al de Java.

participación y la actividad, el tratamiento y presentación de la información e incrementa su comprensión y retención, favoreciendo el aprendizaje educativo mediante la interactividad.

Para la creación de Multimedias existen herramientas, tales como: *Macromedia Director*⁵ y *Macromedia Flash*⁶, las cuales permiten hacer animaciones *flash*, enriquecidas con imágenes, textos, videos y sonidos, disponiendo de posibilidades de programación mediante la utilización del lenguaje *ActionScript*⁷. Estas herramientas presentan la desventaja de ser propietarias, por lo cual se hace necesaria la búsqueda de otras variantes que estén basadas en *Software Libre*, entre las que resaltan: *Gimp*, *Inkscape*, *Wink*, *Synfig*, *Jahshaka*, *Audacity* y la más reciente el Adobe Flex.

Mediante el uso del *Wink* se podrían producir aplicaciones o presentaciones exportadas en archivos *.swf*, esta es una herramienta muy eficiente que permite una integración de medios bastante poderosa y una navegación básica; pero no el desarrollo de aplicaciones más complejas. Es posible crear y editar imágenes, capturas de pantalla y hacer diseños a nivel profesional mediante el *Gimp*, así como, el uso del *Inkscape*, cuando sea necesaria la creación de diseños vectoriales para ilustraciones y dibujos. Digitalizar y editar video con el empleo de *Jahshaka*, editar audio mediante *Audacity* y crear animaciones con el *Synfig*.

A la hora del montaje de todas estas herramientas mencionadas anteriormente para conformar las Multimedias, se hace necesario el empleo del lenguaje *Java* en una versión libre, lo cual implicaría un estudio profundo de su programación. Aunque las posibilidades que permite dicho lenguaje son vastas y ricas, no permitiría que usuarios no expertos en la programación *Java* produzcan materiales multimedia. Es por ellos que dichas herramientas libres no son la solución perfecta para desarrollar multimedias.

⁵ **Macromedia Director**, es una aplicación en forma de estudio destinado para la producción de películas ejecutables.

⁶ **Flash**, es una aplicación en forma de estudio de animación que trabaja sobre "*Fotogramas*" destinado a la producción y entrega de contenido interactivo.

⁷ **ActionScript**, lenguaje de programación orientado a objetos (OOP), utilizado en aplicaciones web animadas.

En estos momentos una de las herramientas libres más usada en la confección de multimedias es el *Adobe Flex*, creado con el fin de integrar toda la *Creative Suite* de *Adobe* (incluyendo *Photoshop*⁸ e *Illustrator*⁹) y otros programas adquiridos recientemente como *Flash* y *Coldfusion*¹⁰, en un entorno muy amigable. Este nuevo *framework* permite la realización de *RIA's*, que no son más que aplicaciones muy ricas en multimedia y por ello son de máxima interactividad con el usuario. Además, dichas aplicaciones tienen la ventaja de ser ejecutadas en tiempo real y ser compatibles en la mayoría de dispositivos y plataformas. *Flex* es un *software* basado *open source* (código abierto), lo que aumenta su velocidad de desarrollo gracias a la posible colaboración de expertos en programación de todos los lugares del mundo (13).

El *framework Adobe Flex*, aporta grandes ventajas a los desarrolladores de aplicaciones Web, de la misma forma que proporcionaba desventajas, pues todas las herramientas que lo componen no llegan a ser del todo libres, en este caso se encuentra el ID de desarrollo **Adobe Flex Builder 3**¹¹, sin este la confección de aplicaciones se hace bastante trabajosa.

1.3.4. Sistemas de Gestión de Aprendizaje (LMS)

Un *LMS* (*Learning Management System*) es un programa o aplicación de *software* instalado en un servidor, que se emplea para administrar, distribuir y controlar las actividades de formación presencial o a distancia de una institución u organización. No hay gestión de los contenidos, sino simplemente administración del curso (14). Generalmente incorporan las siguientes funciones:

- ✓ Gestión y registro de cursos y alumnos.
- ✓ Control de acceso y seguimiento del progreso de los alumnos.

⁸ **Adobe Photoshop**, es una aplicación en forma de taller de pintura y fotografía que trabaja sobre un "lienzo" y que está destinado para la edición, retoque fotográfico y pintura a base de imágenes de mapa de bits.

⁹ **Adobe Illustrator**, es una aplicación en forma de taller de arte que trabaja sobre un tablero de dibujo, conocido como "mesa de trabajo" y está destinado a la creación artística de dibujo y pintura para una ilustración.

¹⁰ **ColdFusion**, es una herramienta que corre en forma concurrente con la mayoría de los servidores web de *Windows*, *Mac OS X*, *Linux* y *Solaris*.

¹¹ **Adobe Flex Builder 3**, es una herramienta de desarrollo basada en Eclipse que incorpora las funciones de: códigos inteligentes, depuración interactiva estratificada, además del diseño visual del aspecto y comportamiento de la interfaz de usuario de las aplicaciones de Internet sofisticadas.

- ✓ Administración y programación de cursos.
- ✓ Gestión de informes.

Existen varios dentro de los cuales se pueden mencionar: *Moodle*, *Sakai*, *Ganesha*, *ATutor*, *Claroline*, *Dokeos*, *.LRN*, sin embargo resaltan dos, siendo los más prometedores y representativos que se emplean en la educación superior para apoyar los programas educativos en las instituciones.

ATutor

Es un Sistema de Gestión de Contenidos de Aprendizaje de código abierto basado en la Web y diseñado con el objetivo de lograr accesibilidad y adaptabilidad. Centrado en la creación, gestión y distribución de contenidos, con algunas herramientas de comunicación añadidas.

Moodle

Es una plataforma para la implementación de cursos a distancia que permite la administración de estos, *software* diseñado para apoyar a los educadores en la creación de cursos en línea de alta calidad. Centrado en la comunicación y las actividades de enseñanza-aprendizaje que incluyen, también, herramientas para gestionar materiales.

Criterios de Selección

Si se fuera a emplear uno de ellos es necesario establecer una comparación entre estos dos antes mencionados por ser los más usados, para poder escoger el que se ajuste a las necesidades. Para ello se tuvo en cuenta los métodos empleados por diferentes universidades como: Universidad Jaime I, Instituto Venezolano de Investigaciones Científicas, Universidad de las Ciencias Informáticas, entre otras. Fueron revisados además los sitios *EduTools*¹² y *Edutech*¹³ que definen una serie de requisitos para que las plataformas puedan ser evaluadas y proporcionan ayuda para la comparación y toma de decisiones entre la gran cantidad de plataformas de *e-learning* existentes.

¹² **EduTools**, sede de herramientas libres para evaluadores y usuarios de aplicaciones educativas.

¹³ **Edutech**, Centro Profesional de Capacitación que ofrece educación de calidad basada en modelos y programas educativos innovadores.

	ATutor	Moodle
<p>Foros</p> <p>Foros de debate.</p>	<p>Se añaden en una sección de "discusiones" que también puede incluir chats. Todos los miembros del curso pueden intervenir e iniciar debates.</p> <p>No se pueden efectuar búsquedas por palabras dentro de los foros.</p> <p>Los usuarios pueden suscribirse a cada debate para recibir los mensajes por correo electrónico.</p>	<p>El profesor puede utilizar un foro en cualquier punto del curso. En cada uno puede decidir si los estudiantes podrán iniciar, contestar debates o si se limitarán a leer sus envíos.</p> <p>Los mensajes pueden escribirse en formato texto o <i>HTML</i>. Se pueden adjuntar archivos.</p> <p>Se pueden efectuar búsquedas por palabras dentro de todos los foros de un curso.</p> <p>Los usuarios pueden suscribirse a cada foro y recibir los mensajes por correo electrónico.</p>
<p>Materiales</p> <p>Posibilidades de gestión de archivos, enlaces externos, entre otros.</p>	<p>El profesor tiene un área privada de archivos en cada curso, que puede organizar por medio de carpetas y subcarpetas.</p> <p>Se pueden subir grupos de archivos en formato <i>Zip</i>¹⁴ y descomprimirlos en el servidor.</p> <p>El administrador puede determinar qué tipos de archivos (extensiones) no se pueden cargar (por defecto no se pueden cargar programas ejecutables).</p> <p>El curso tiene asociada también una sección de enlaces externos ("<i>Links Data Base</i>"), que se puede organizar</p>	<p>El profesor tiene un área privada de archivos en cada curso, que puede organizar por medio de carpetas y subcarpetas.</p> <p>Se pueden subir grupos de archivos en formato <i>Zip</i> y descomprimirlos en el servidor.</p> <p>No se puede definir qué tipo de archivos (extensiones) se pueden cargar o no.</p> <p>También se pueden incluir enlaces externos en un curso, uno por uno, con diversas opciones de visualización: dentro de la ventana del curso y en una ventana nueva.</p>

¹⁴ **Zip**, formato de compresión.

	<p>por categorías y admite aportaciones de los estudiantes (moderadas por el profesor).</p>	
<p>Ejercicios ¿Se pueden crear test, cuestionarios o ejercicios auto-correctivos?</p>	<p>El profesor puede crear cuestionarios con preguntas de verdadero/falso, de respuestas múltiples y de repuesta libre. El <i>test</i> está activo durante un período de tiempo determinado.</p> <p>El profesor debe revisar las respuestas para poner la puntuación.</p>	<p>El profesor puede crear cuestionarios con preguntas de verdadero/falso, de respuestas múltiples, de repuesta libre, de relacionar y de otros tipos. Las preguntas se van acumulando en una base de datos y son reutilizables en otros cuestionarios y en otros cursos. También se pueden importar desde ficheros externos.</p> <p>Se pueden reordenar las preguntas aleatoriamente cada vez que un estudiante rellena el cuestionario, se puede mostrar la respuesta correcta después de que la conteste el estudiante, entre otras.</p> <p>El sistema puede calcular y mostrar las puntuaciones automáticamente.</p>
<p>Otros Otras herramientas de enseñanza-aprendizaje relevantes.</p>	<p>Glosario: el profesor puede crear un glosario con términos utilizados a lo largo del material del curso. No se crean enlaces automáticos a los términos del glosario.</p>	<p>Glosario: el profesor puede crear diferentes glosarios. Hay una opción para activar enlaces automáticos al glosario que aparecen cada vez que se utiliza un término de él.</p> <p>Diarios: el profesor puede activar diarios de aprendizaje privados donde el alumno escribe sobre un tema de estudio. El profesor puede leer, comentar y evaluar estas entradas.</p>

Extensibilidad ¿Cabe la posibilidad de incorporar otras herramientas de aprendizaje adicionales?	No existe esta posibilidad.	La arquitectura modular de <i>Moodle</i> permite crear fácilmente herramientas adicionales programadas en <i>PHP</i> ¹⁵ . Hay una plantilla y documentación.
Multimedia ¿Hay alguna facilidad especial para distribuir contenidos de audio o vídeo?	No existen facilidades especiales.	Incorpora un sistema de reproducción (<i>streaming</i>) de audio en formato <i>MP3</i> ¹⁶ basado en <i>Flash</i> . Puede ser útil para la enseñanza de idiomas, entrevistas y otras.
Facilidad de uso Facilidad de uso percibida por los usuarios	Baja. Percepción del entorno: confuso, embrollado. Tiene una lógica propia difícil de captar de entrada.	Media-alta. Percepción del entorno: sencillo, amigable, bastante intuitivo. Las operaciones básicas se realizan sin dificultad.

Tabla 1. Comparación de ATutor/Moodle.

De acuerdo a la comparación antes expuesta se puede llegar a la conclusión que los dos EVA son técnicamente viables en lo que respecta a sus posibilidades de integración; ambos ofrecen las funcionalidades básicas suficientes para utilizarse como entornos virtuales de enseñanza-aprendizaje en una universidad. Sin embargo, en el momento presente la ventaja de *Moodle* parece clara en casi todos los aspectos comparados:

¹⁵ **PHP**, es un lenguaje de programación interpretado.

¹⁶ **MP3**, es un formato de audio digital comprimido con pérdida.

- ✓ Ofrece más funcionalidades didácticas y éstas son más sofisticadas y ricas en opciones. Al mismo tiempo, el diseño modular del entorno garantiza su flexibilidad: según los módulos empleados puede dar soporte a cualquier tipo de estilo docente o modalidad educativa.
- ✓ Gracias a su diseño modular y a una mayor atención a la interfaz de usuario, el índice de usabilidad de *Moodle* es superior al de *ATutor*.
- ✓ El grado de apertura y el dinamismo del proyecto son también más elevados.

1.3.5. Sistemas de Gestión de Contenidos (CMS)

Los sistemas de gestión de contenidos (*Content Management Systems*) o "sistema de soporte a la Gestión de contenidos", son estrategias de comunicación que gestionan los contenidos de forma efectiva, se utiliza principalmente para facilitar la gestión de Webs. El sistema permite manejar de manera independiente el contenido y el diseño. Así, es posible manejar el contenido y darle en cualquier momento un diseño distinto al sitio sin tener que darle formato nuevamente (15).

Dentro de estos se encuentran el *OpenCMS*, *TYPO3*, *Joomla*, *Drupal*, *Nuxeo*. Resaltando el *Joomla* debido a sus cautivantes ventajas:

- ✓ *Software Open Source. Software Libre* (Licencias *GNU/GPL*), ampliable al disponer el código fuente.
- ✓ Completa y fácil administración por Web.
- ✓ Creación de la Web por inserción de módulos y componentes independientes.
- ✓ Creación y actualización dinámica de secciones, categorías y contenidos (públicos y privados).
- ✓ Creación de perfiles y privilegios con niveles jerárquicos para diferentes niveles de usuarios (súper administrador, administrador, *mánager*, *publisher* y usuario registrado).
- ✓ Plantillas (*templates*) para modificar el diseño gráfico de la Web de forma automática en pocos minutos.
- ✓ Zonas configurables y personalizables.
- ✓ Administrador gráfico de fácil utilización para editar y eliminar artículos/temas.
- ✓ Fácil inserción de imágenes y galerías de fotos.
- ✓ Fácil creación de enlaces.
- ✓ Sistema de gestión de banners integrado.
- ✓ Motor de búsqueda global integrado.

- ✓ Editores de contenido basados en *WYSIWYG*¹⁷.

El sistema permite generar de forma automática los documentos como archivos *PDF*¹⁸, en vista de poderse imprimir o enviar por correo electrónico.

1.4. SOFTWARE EDUCATIVO EN LA UCI

La Universidad de las Ciencias Informáticas debe ser una institución de avanzada en la aplicación de las Tecnologías de Información y las Comunicaciones, el desarrollo y perfeccionamiento del proceso de enseñanza-aprendizaje ocupa un lugar prioritario y como universidad dinámica e innovadora; en su misión de entregar a la sociedad profesionales en una especialidad tan vinculada a estas tecnologías como es la Ingeniería Informática, han comenzado a utilizar las TIC en los procesos educativos. Esto ha representado el surgimiento de nuevos modelos y sistemas de formación.

Uno de los factores fundamentales es la selección del EVA, mediante el cual se realiza la distribución y gestión de cursos a través de la red. Teniendo en cuenta que no existe la mejor plataforma, sino que la selección debe estar ante todo basada en las prioridades que identifique cada universidad, a partir de las características de su Modelo de Formación se evidenció que la plataforma **Moodle** que supera en varios aspectos al resto está seleccionada como el **Entorno Virtual de Aprendizaje de la Universidad de las Ciencias Informáticas**.

Estas funcionalidades que hacen del *Moodle* una plataforma de avanzada no están siendo usadas del todo en los cursos de segundo perfil que se encuentran publicados, solo están disponibles los documentos, no hay cuestionarios y algunos de los publicados no funcionan. Esto hace que los estudiantes tengan que estar muy interesados en la lectura de estos contenidos. Se debe tener muy en cuenta que estos cursos tienen un carácter menos presencial, la sola disposición de los documentos en la plataforma, no basta.

Se observa que carece de diseño, pues esta herramienta se encarga mayormente de la administración de los cursos, no así de los contenidos de estos, y considerando que en el desarrollo

¹⁷ **WYSIWYG**, (*What You See What You Get*) se aplica a los procesadores de texto y otros editores de texto con formato (como los editores de *HTML*).

¹⁸ **PDF**, (*Portable Document Format*), en español, formato de documento portátil. Es un formato de almacenamiento de documentos de tipo compuesto (imagen vectorial, mapa de bits y texto).

de un determinado material educativo se deben cuidar con especial atención los aspectos didácticos; y más cuando se utiliza como medio de transmisión de conocimientos la metodología de *e-learning*, se tienen que tener en cuenta las limitaciones y posibilidades que ofrece este medio.

Una misma información se puede presentar de formas muy diferentes y dependiendo de los objetivos que intente cubrir y del medio en el que se vaya a implementar, se deben transformar los contenidos que se pretenden transmitir. A partir de esto se han venido creando multimedias en apoyo a estos cursos en la facultad 8, como Trabajos de Tesis, para Jornadas Científicas o por iniciativa propia de los estudiantes y profesores que se enfrentan a este dilema. Por nombrar algunas: Curso optativo interactivo *Flash Basic*, Multimedia interactiva para el Bloque Básico General del Perfil de Calidad de *Software*, Multimedia curso *XML*¹⁹, Multimedia interactiva de *Adobe Photoshop*, Multimedia de Cursos Optativos *Macromedia Director*.

Estas multimedias son muy interactivas y dinámicas, muestran los contenidos de una mejor forma para los estudiantes, pero presentan algunas dificultades como: el empleo de herramientas propietarias en su confección pues usan *Macromedia Flash*, *Macromedia Director* y algunas complementarias como *Adobe Premier*²⁰, *Adobe Photoshop*; el acceso de los estudiantes a ellas no es muy habitual, pues no se encuentran centralizadas en un mismo sitio; con los avances en las tecnologías se producen cambios en los lenguajes a impartir, en los programas que se emplean en la dinámica producción de la UCI y no pueden ser actualizados, por la razón de no estar disponibles. Otra desventaja que se le atribuye es que el diseño en el cual están montados sus contenidos difiere de un producto a otro, lo cual tiende a confundir a los usuarios.

Para lograr una mejora en la forma en que llegan estos contenidos a los estudiantes y que los profesores que imparten estos cursos se sientan respaldados por una buena aplicación se hace necesario que exista un entorno interactivo sobre tecnologías libres, que se obtendría a partir del uso del diseño multimedia en la forma en que se exponen los contenidos mediante el *Joomla* y el empleo de los recursos educativos que brinda *Moodle*: cuestionarios.

¹⁹ **XML**, (*Extensible Markup Language*), en español, lenguaje de marcas ampliable, es un metalenguaje extensible de etiquetas.

²⁰ **Adobe Premiere**, es una aplicación en forma de estudio destinada a la edición de vídeo en tiempo real.

Cuando se habla de la migración a *Software Libre* en la UCI, es necesario mencionar al proyecto Multisaber, el cual se encuentra inmerso en este proceso, teniendo como base el producto de la colección cubana de Multisaber, compuesta por 14 multimedias que apoyan el proceso de aprendizaje de los niños, las cuales están programadas con herramientas propietarias y utilizan recursos mediáticos no libres. Actualmente se encuentran trabajando en el Gestor de Contenidos Joomla, en el cual han ido desarrollando y mejorando algunas de las herramientas que lo conforman con fin de satisfacer las necesidades del nuevo producto; esto ha servido como guía y apoyo en la elección de la solución de las problemáticas planteadas en este trabajo.

1.5. CONCLUSIONES PARCIALES

Al terminar este capítulo se han expuesto conceptos de gran ayuda para el entendimiento del problema a resolver y su solución, como son: *Software Educativo*, sus tipos, su inclusión en las universidades y dentro de estas se resalta la importancia de los Sistemas de Gestión de Aprendizaje y de Contenidos además de las multimedias, su empleo en la Universidad de las Ciencias Informáticas y específicamente en la facultad 8. Luego de su estudio se puede notar como algunas de las herramientas que se encuentran en uso en la UCI no satisfacen del todo las necesidades del proceso enseñanza-aprendizaje de los cursos de segundo perfil y se muestran los primeros indicios de una posible solución.

Capítulo 2

Descripción de la Propuesta

Para crear un ambiente de aprendizaje interactivo que emplee las ventajas que brindan los aspectos de la tecnología Multimedia y obvие las principales dificultades que presenta, fundamentalmente el uso de herramientas propietarias, se describe el proceso de montaje de los cursos de segundo perfil en el *CMS Joomla*, empleando algunas de las funcionalidades que ofrece el *LMS Moodle*, basado en *Software Libre* y que brinda a sus usuarios actividades para los cursos que incluyen muy diversos tipos de cuestionarios.

Tanto el *CMS Joomla* como el *LMS Moodle* cuentan con un amplio número de librerías y códigos disponibles y están respaldados por una amplia comunidad que los hace proyectos en continuo desarrollo al trabajar permanentemente para ofrecer cada día un *software* mejor y más seguro, donde se incluye la vigilancia exhaustiva de posibles vulnerabilidades en el código. Igualmente, pueden ser aprovechados por un inmenso número de usuarios facilitando su distribución por medio de la red, *CDs*, entre otros.

Con el uso de la Web para la distribución de estos cursos se puede llegar a un público más numeroso, pues son muy fáciles de usar al no exigir ningún conocimiento de informática adicional, permite acceder a todo tipo de documentos multimedia: no solamente texto, sino también información sonora y audiovisual. Además, abre nuevos espacios de comunicación entre profesores y alumnos, aumentando las capacidades sociales y de colaboración humana y fomentando el aprendizaje constructivista. Es una forma de salir del entorno del aula, lo que se hace en clase se podrá continuar trabajando desde cualquier otro sitio por las facilidades de conexión que brinda y con relación a los cursos que no son presenciales fomentará el auto aprendizaje del estudiante.

2.1. LEVANTAMIENTO DE LAS FUNCIONALIDADES DE LAS HERRAMIENTAS QUE SE PUEDEN REUTILIZAR

De acuerdo a lo antes expuesto se muestran las funcionalidades que brindan el *Moodle* y las antiguas multimedias elaboradas en la facultad 8, para a partir de estas elaborar los requisitos que permitirán lograr un mejor diseño que satisfaga las necesidades del usuario.

2.1.1. Moodle

- ✓ Los cursos pueden clasificarse por categorías y también pueden ser buscados.
- ✓ Los "temas" permiten al administrador personalizar los colores del sitio, fuentes, presentación, entre otros, para ajustarse a sus necesidades.
- ✓ Pueden añadirse nuevos módulos de actividades a los ya instalados en *Moodle*.

- ✓ Soporta un rango de mecanismos de autenticación a través de módulos para esto, que permiten una integración sencilla con los sistemas existentes.
- ✓ Cada persona necesita sólo una cuenta para todo el servidor. Por otra parte, cada cuenta puede tener diferentes tipos de acceso.
- ✓ Una cuenta de administrador controla la creación de cursos y determina los profesores, asignando usuarios a los cursos.
- ✓ Una cuenta como autor de curso permite sólo crear cursos y enseñar en ellos.
- ✓ Los profesores pueden dar de baja a los estudiantes manualmente si lo desean, aunque también existe una forma automática de dar de baja a los estudiantes que permanezcan inactivos durante un determinado período de tiempo (establecido por el administrador).
- ✓ Permite colocar recursos variados para formar una unidad de contenidos: etiquetas, archivos en formato variable (texto, audio, vídeo, hoja de cálculo, documento, presentación), Web externas, edición de Webs.
- ✓ Ofrece una serie flexible de actividades para los cursos: foros, glosarios, cuestionarios, recursos, consultas, encuestas, tareas, diarios, *wikis*, lecciones, *chats* y talleres.
- ✓ Registro y seguimiento completo de los accesos del usuario. Se dispone de informes de actividad de cada estudiante, con gráficos y detalles sobre su paso por cada módulo (último acceso, número de veces que lo ha leído) así como también de una detallada "historia" de la participación de cada estudiante, incluyendo mensajes enviados, entradas en el glosario, etc. en una sola página.
- ✓ Puede funcionar en cualquier ordenador en el que pueda correr *PHP* y soporta varios tipos de bases de datos (en especial *MySQL*²¹).
- ✓ La mayoría de las áreas de introducción de texto (recursos, mensajes de los foros y otros) pueden ser editadas usando el editor *HTML*.
- ✓ Los cursos se pueden empaquetar en un único archivo *Zip* utilizando la función de "copia de seguridad".
- ✓ Logra una interfaz y navegación funcional, tanto para usuarios expertos, como para los que no tienen conocimientos profundos de informática, de forma que los usuarios que interactúen con la aplicación, no presenten dificultades en la asimilación de la herramienta.

²¹ **MySQL**, sistema de gestión de base de datos relacional, multihilo y multiusuario que ofrece cierta biblioteca para ser utilizado por otro software.

- ✓ Garantiza la mayor eficiencia posible, se requiere que la velocidad de procesamiento sea tan alta como se pueda, se asegura un tiempo óptimo de respuesta.
- ✓ Las funcionalidades del sistema se muestran de acuerdo al nivel de usuario que este activo.

2.1.2. Antiguas Multimedias

- ✓ Mostrar y saltar presentación del producto.
- ✓ Controlar audio del producto.
- ✓ Mostrar pantalla principal del programa, donde el usuario debe recibir información general acerca de las especificaciones y objetivos del perfil.
- ✓ Mostrar menú de contenidos.
- ✓ Mostrar materiales complementarios en el menú de contenidos.
- ✓ Mostrar ejercicios en el menú de contenidos: basados en preguntas, de verdaderos y falsos, de selección múltiple, de ordenar frases, de asociación, de completar espacios en blancos con la palabra o frase correcta, entre otros.
- ✓ Mostrar juegos interactivos en el menú de contenidos: crucigramas, de preguntas y respuestas.
- ✓ Mostrar plan de la asignatura en el menú de contenidos.
- ✓ Mostrar glosario de términos.
 - Permitir seleccionar palabra caliente.
- ✓ Mostrar la Ayuda del sistema al seleccionar el botón de ayuda.
- ✓ Visualizar la galería de imágenes.
 - Permitir la visualización de la galería al seleccionar las palabras calientes correspondientes a imágenes situadas en la galería.
- ✓ Mostrar un listado con los videos del tema.
 - Reproducir vídeo.
 - Pausar vídeo.
 - Parar reproducción de vídeo.
 - Permitir controlar la barra de estado de reproducción del vídeo.
 - Controlar volumen del vídeo.
- ✓ Permitir cerrar la aplicación en cualquier momento.
- ✓ Mostrar créditos al confirmar salida de la aplicación.
- ✓ El formato *.swf* de *Macromedia Flash* es independiente del sistema operativo por lo que puede ser ejecutado sin dificultad en cualquier sistema operativo.

- ✓ El sistema debe contener una interfaz amigable al usuario, brindándole todas las facilidades necesarias para que esta sea lo más cómoda y agradable posible.
- ✓ Las medias siempre se visualizarán utilizando la misma área de la interfaz para evitar el movimiento innecesario entre interfaces y la posible pérdida del usuario en la aplicación.
- ✓ Tiene que existir consistencia entre la presencia de elementos de la interfaz y la funcionalidad prevista en cada situación (no deben existir elementos interactivos activos o visibles cuya funcionalidad no sea necesaria en un contexto dado).
- ✓ La aplicación debe funcionar tanto desde el *CD* como del Disco Duro (*HDD*).
- ✓ El color de la letra debe estar en contraste con el fondo del área de texto para facilitar su lectura.
- ✓ El *software* tendrá siempre la posibilidad de ayuda disponible al usuario, lo que le permitirá un avance considerable en la explotación de la aplicación en todas sus funcionalidades.
- ✓ Desde cualquier pantalla se podrá acceder a cualquier módulo.
- ✓ Se podrá abandonar el programa desde cualquier pantalla.
- ✓ Existirá un elemento localizador que le indicará al usuario, que subtema del menú está viendo en ese instante.
- ✓ Todas las pantallas presentarán los siguientes servicios generales con los botones interactivos siguientes: *música de fondo, inicio, menú, imprimir, ayuda, salir*. Los cuales se activarán o desactivarán según el contexto.
- ✓ El instalador debe garantizar de manera transparente la corrida exitosa de todos los componentes del programa.
- ✓ El instalador debe crear un directorio para la instalación del producto.
- ✓ El instalador debe crear un acceso directo a la aplicación en el menú principal del sistema operativo.
- ✓ La aplicación debe ser protegida para que no sea posible hacerle modificaciones al código fuente.

En vista a plantear una guía de implementación que permita la creación de un entorno virtual interactivo sobre tecnologías libres que apoyen los cursos de segundo perfil y tratar de dar una solución que cubra la mayor cantidad de los requerimientos metodológicos propuestos en las multimedias anteriores; se listan las funcionalidades que consiguen traer problemas técnicos, sobre la forma en que se puedan desarrollar sobre la Web en el momento de su migración.

2.2. FUNCIONALIDADES DE LA SOLUCIÓN QUE SE PROPONE

Con el fin de entender mejor la propuesta de solución que se propone se agruparon las funcionalidades que tendrá la misma de la siguiente manera:

Interfaz

- ✓ Diseño de la interfaz.

Contenidos

- ✓ Mostrar los contenidos en formato texto.
- ✓ Mostrar menú de contenidos.
- ✓ Mostrar en la primera página las especificaciones y descripción del curso.
- ✓ Permitir la creación de palabras calientes que accedan al glosario de términos y galería de imágenes y videos.

Módulo Cuestionario

- ✓ Configurar de manera general el cuestionario.
- ✓ Editar categorías en el cuestionario.
 - Editar preguntas en cada categoría.
 - Las categorías contendrán preguntas que pueden ser de tipo:
 1. Calculadas
 2. Descripción
 3. Ensayo
 4. Emparejamiento
 5. Respuestas incrustadas (*Cloze*)
 6. Opción múltiple
 7. Respuesta corta
 8. Numérico
 9. Emparejamiento aleatorio de respuestas cortas
 10. Verdadero/Falso
- ✓ Adicionar preguntas al cuestionario.
- ✓ Mostrar los resultados de los estudiantes.

Glosario de Términos

- ✓ Acceder al Glosario de Términos.

Galerías de Imágenes

- ✓ Acceder a la galería.

Galerías de Videos

- ✓ Acceder a la Galería de Videos
- ✓ Permitir la reproducción del video, su pausa y detención, control de su barra de estado y volumen.

Audio

- ✓ Reproducir música de fondo, poder detenerla en cualquier momento.

Portabilidad

- ✓ La aplicación debe funcionar tanto desde el *CD* como del Disco Duro (*HDD*).

Apariencia

- ✓ Mostrar y saltar presentación del curso.
- ✓ Permitir cerrar la aplicación en cualquier momento.

2.3. SELECCIÓN DE LAS HERRAMIENTAS PARA CUBRIR LAS FUNCIONALIDADES

La selección de las herramientas debe responder a las funcionalidades que se planteen, una vez realizada la misma y conocidos los elementos que las componen se puede pasar a describir las funcionalidades basadas en el empleo de las herramientas elegidas.

2.3.1. CMS Joomla

Dentro de las funcionalidades que se deben cumplir se encuentra la Gestión de Contenidos, para darle solución a la misma existen los Sistemas de Gestión de Contenidos (*CMS*), dentro de los que resalta Joomla, pues sirve para construir soluciones de gestión de contenidos y cuenta con diversas funcionalidades destacándose entre muchas otras el ser ampliable al disponer el código fuente, permitir una completa y fácil administración por la Web y contener zonas configurables y personalizables.

La versión del *CMS Joomla* que se va a emplear es la 1.x debido a que presenta un panel de administración más desarrollado, otorgando una mejor gestión de contenidos y facilidad de uso,

además utiliza el “soporte *Legacy*”, el cual posibilita la adaptación de componentes y módulos desarrollados para versiones anteriores de *Joomla* y una de las razones más importantes por la cual se escoge esta versión es que contiene un gran número de colaboradores que continúan dándole soporte al mismo, razón por la cual no se escoge la versión 1.0.x.

El *CMS Joomla* 1.x contiene diferentes elementos de configuración que son necesarios conocer:

Plantillas

La plantilla (*template*) y sus archivos asociados proporcionan el aspecto visual y el manejo del sitio Web y se mantienen separados del contenido del mismo, esta se almacena en una base de datos *MySQL* (16) y trabaja básicamente del mismo modo que una página *HTML* única, lo que simplifica mucho el trabajo para el diseñador gráfico. El contenido dinámico se inserta mediante etiquetas *PHP* en la posición en la cual se desea que se visualice en el diseño de la página. La instalación habitual de *Joomla* incluye varias plantillas pre-instaladas. Los archivos de la plantilla se ubican en la carpeta “*templates*” en el directorio principal de instalación.

Para que el sitio Web tenga un aspecto de multimedia lo fundamental es el cómo va a estar diseñada la plantilla que está en uso, para que se pueda personalizar es necesario conocer cómo funciona y los archivos que la componen:

Nombre del Archivo	Detalles
index.php	Archivo que contiene código <i>HTML</i> , <i>PHP</i> y posiblemente <i>JavaScript</i> , proporciona el soporte para el diseño del sitio Web. Además, en combinación con los archivos <i>CSS</i> ²² y las imágenes, determinan la estructura del diseño y los elementos de contenido del sitio.
templateDetails.xml	Contenedor de los detalles descriptivos de la plantilla y las referencias a todos los archivos usados, ejemplo: <i>index.php</i> , el archivo <i>CSS</i> .

²² **CSS**, (*Cascading Style Sheets*), en español, hojas de estilos en cascadas, lenguaje formal usado para definir la presentación de un documento estructurado escrito en *HTML* o *XML*.

template_thumbnail.png	Se usa, en el Administrador de Plantillas para proporcionar, si está activado, una imagen en miniatura del diseño del sitio Web. Debe estar situado en el directorio raíz de la plantilla.
Template_css.css	Situado en la carpeta "CSS" del directorio de la plantilla, esta Hoja de Estilos en Cascada (CSS) contiene el código que define el estilo visual del sitio Web, ejemplo: tamaños de letra, colores. Pueden existir tantos archivos CSS como se desee proporcionar en la referencia <i>HTML</i> del archivo.
Archivos de Imagen	Estos archivos están situados en la carpeta " <i>images</i> " y pueden ser <i>.gif</i> , <i>.jpg</i> o <i>.png</i> . Constituyen los elementos gráficos del diseño del sitio Web.

Tabla 2. Archivos de las Plantillas en Joomla.

Componentes

Los Componentes son elementos funcionales del núcleo de *Joomla*, reconocidos como la extensión fundamental y más compleja, se muestran en el cuerpo principal de la plantilla del sitio Web (17). El código para el cuerpo principal aparecerá en la plantilla del siguiente modo: `<?php mosMainBody.php ();?>`. Dependiendo del diseño de la plantilla utilizada, suelen estar en el centro de la página Web o en cualquier otro lugar determinado.

Son pequeñas aplicaciones independientes entre sí que gestionan la información dentro del sitio y algunos están relacionados a las funciones principales de la Web, como por ejemplo la gestión de artículos o el manejo de usuarios. Siempre se debe designar el nombre de los componentes con el prefijo "**com_** nombre". Una instalación típica de *Joomla* contiene diversos componentes: *Banners*, *Contactos*, *Noticias Externas*, *Encuestas* y *Enlaces Web*. Estos se pueden quitar o instalar otros en caso que sea necesario.

Para la creación de un componente en la versión 1.x de *Joomla* se ha incorporado el uso del patrón de diseño *MVC*²³, con lo cual se facilita el mantenimiento del componente y lo hace más fácil de

²³ **MVC**, (*Model View Controller*), en español modelo vista controlador.

extender y leer por otras personas. Una aplicación Web basada en este patrón separa su código en tres partes diferenciadas:

- ✓ **El controlador:** es el punto de entrada de la aplicación, se mantiene a la escucha de todas las peticiones, ejecuta la lógica de la aplicación y muestra la vista apropiada para cada caso.
- ✓ **El modelo:** contiene todo el código relacionado con el acceso a datos. Es importante que sea un código lo más genérico posible y se pueda reutilizar en otras situaciones y proyectos. Nunca se incluye la lógica en el modelo, solamente consultas a la base de datos y validaciones de entrada de datos.
- ✓ **La vista:** contiene el código que representará, lo que se muestra por pantalla, en este caso se trata del código *HTML*.

El componente contiene dos partes diferenciadas: el *front-end* y el *back-end*:

- ✓ **Front-end:** es la parte del componente que es visible para el usuario del sitio Web. Se visualiza donde se haya incluido la etiqueta `<jdoc:include type="component" />` en la plantilla. Su código fuente se encuentra en el directorio *components* dentro de la carpeta principal de *Joomla*.
- ✓ **Back-end:** es la parte del componente que se verá cuando se entra al sitio como administrador. El componente se mostrará donde se haya especificado en la plantilla del *back-end*. La plantilla para la parte del *back-end* y la parte del *front-end* es diferente. Su código fuente se encuentra en la carpeta *components* dentro de la carpeta *administrator* ubicado en el directorio principal de *Joomla*.

La parte que se pondrá en el *front-end* tendrá básicamente una estructura como la mostrada en la Figura 1.

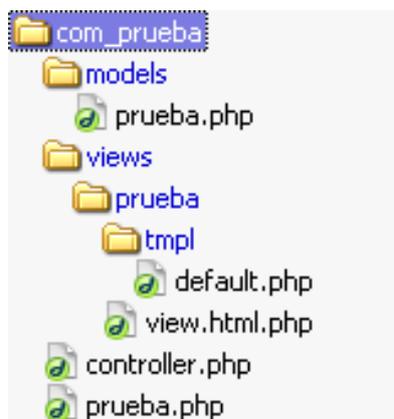


Figura 1. Estructura de directorios para la parte de front-end de los Componentes en Joomla.

Nombre del Archivo	Detalles
prueba.php (com_prueba/prueba.php)	Este archivo es el punto de entrada, es el que busca <i>Joomla</i> y ejecuta cuando se llama al componente. Es donde se crea el controlador y se ejecutan sus métodos <i>execute</i> y <i>redirect</i> , con este último se redirigirá el flujo del programa a la vista adecuada.
controller.php	Aquí se crean o importan las clases que harán función de controlador y se da paso a la vista correspondiente.
models	Directorio donde se ubicará el modelo.
prueba.php (com_prueba/models/prueba.php)	Archivo ubicado dentro de la carpeta <i>models</i> , es donde se hacen las consultas a la base de datos y las validaciones de entrada de datos. La clase declarada dentro de este tendrá el nombre <i>NombreVista/Model/NombreModelo</i> .
views	Directorio que contendrá las vistas. Debe existir una carpeta para cada vista en particular y llevar el nombre de la misma.
view.html.php	Archivo ubicado dentro de cada vista, en el se crea la clase con el nombre <i>NombreComponente/View/NombreVista</i> .
tmpl	Directorio ubicado dentro de <i>views/prueba</i> donde se guardarán los <i>layouts</i> de la vista. El <i>layout</i> es el fichero final que se mostrará por el navegador.

Tabla 3. Archivos para la parte de front-end de los Componentes en Joomla.

Hasta aquí se describe cómo se estructuran los archivos que van en el front-end, la parte de back-end se desarrolla de forma similar, pero bajo la carpeta *administrator/components* del sistema de directorios de *Joomla*.

La parte que se pondrá en el *back-end* tendrá básicamente una estructura como la que se muestra en la Figura 2.

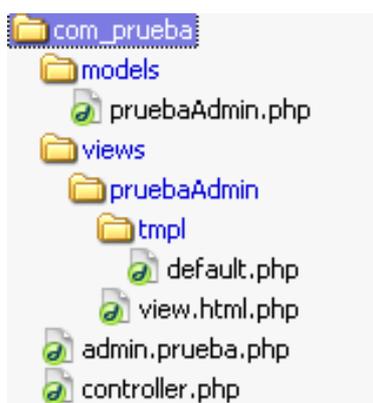


Figura 2. Estructura de directorios para la parte de back-end de los Componentes en Joomla.

La funcionalidad de cada uno de los archivos del *back-end* corresponde a su similar en la parte del *front-end*, lo que en este caso las vistas contienen lo que se va a mostrar cuando se entra al sistema como administrador, cuando se ejecuta el componente desde el *back-end* de *Joomla* y en el caso del modelo serían las consultas a la base de datos mediante las cuales se introducen los datos que serán vistos desde el *front-end*. Se pueden crear los archivos que se necesiten para la correcta funcionalidad del componente a desarrollar siempre respetando el patrón de diseño *MVC*.

El *API*²⁴ de *Joomla* provee objetos específicos para operar con bases de datos, este es el *JTable* usado para crear, leer, actualizar y borrar registros de una tabla. Siempre que se pueda es recomendable usarlo, en lugar de conectar directamente con las funciones de *MYSQL* que tiene *PHP*, este objeto hace mucho trabajo por debajo, como protección contra la inyección *SQL*²⁵ y

²⁴ **API**, (**A**pplication **P**rogramming **I**nterface), es un conjunto de funciones y procedimientos o métodos.

²⁵ **SQL**, es un sistema de gestión de bases de datos relacionales (*SGBD*) basado en el lenguaje *Transact-SQL*, y específicamente en *Sybase IQ*, capaz de poner a disposición de muchos usuarios grandes cantidades de datos de manera simultánea.

optimización. Si se necesitara un acceso a datos más profundo y este *API* no lo satisface se pueden realizar las consultas de la forma tradicional mediante el lenguaje *SQL*.

Para la creación final del componente instalable hay simplemente que generar un archivo *XML* con nombre igual al del componente y con un formato específico, en este se especificarán características propias del componente como su creador, descripción; se llama a los *Scripts PHP* necesarios para la instalación y desinstalación del componente, se especifican los puntos de entrada al componente y todos los archivos que contiene y en qué lugar se copiarán dentro del directorio principal de *Joomla*.

Luego de generado este archivo se comprime en formato *Zip* junto a todos los demás necesarios para el correcto funcionamiento del componente. Este fichero será leído por el instalador de *Joomla* y de ahí sacará la información para proceder a instalarlo.

Módulos

Los módulos proporcionan una nueva funcionalidad al *software*, son los que muestran generalmente información secundaria, simplificada y breve que toman de la base de datos y se cargan sobre los laterales del sitio o en cualquier parte que la plantilla lo permita. Por ejemplo, los módulos nos dicen quién está *online*, cuáles son los últimos artículos publicados o los más vistos; el formulario de ingreso de usuarios es también un módulo. Estos son a veces un complemento de los componentes. Siempre el nombre de los componentes debe contener el prefijo "**mod_** nombre" y un ejemplo del código de un módulo es: `<?php mosLoadModules ("module position");?>`.

Plugin

Una de las extensiones más avanzadas de *Joomla* es el *plugin*. En las versiones de *Joomla* 1.0.x los *plugin* se conocían como *mambots*²⁶. Con el desarrollo de *Joomla* 1.x, los *mambots* fueron renombrados a *plugin* y se ha ampliado su funcionalidad. Un *plugin* es una porción de código que se ejecuta cuando tiene lugar un evento predefinido en *Joomla*. Los editores son *plugins*, por ejemplo, que se ejecutan cuando tiene lugar el evento "*onGetEditorArea*" de *Joomla*. El uso de *plugins* permite al desarrollador cambiar el modo en que se comporta el código, dependiendo de los *plugins* instalados para reaccionar a un determinado evento (18).

²⁶ **Mambots**, son pequeños códigos *PHP* que se ejecutan en el momento que se va a mostrar el contenido en la salida de la página web.

2.3.2. LMS Moodle

Otra de las funcionalidades que se requiere es la diversidad de tipologías de cuestionarios, para darle solución a esta, la mejor herramienta que se propone es el *LMS Moodle*, pues dentro de los Sistemas Gestores de Aprendizaje es uno de los que cuenta con más funcionalidades didácticas y éstas son más sofisticadas y ricas en opciones.

Este se distribuye gratuitamente como *Software* libre (bajo la Licencia Pública *GNU*); esto significa que *Moodle* tiene derechos de autor, pero que al mismo tiempo posibilita algunas libertades. Puede ser copiado, usado y modificado siempre que acepte: proporcionar el código fuente a otros, no modificar o eliminar la licencia original y los derechos de autor y aplicar esta misma licencia a cualquier trabajo derivado de él. Estas características de *Moodle* son las que permitirán el uso del código fuente y librerías ya existentes para el desarrollo de cuestionarios en otras plataformas Web.

La instalación viene con un paquete estándar de módulos, estos se encuentran en el directorio principal de *Moodle*, en una carpeta nombrada “mod”, cada uno en un subdirectorio separado y consta de los siguientes elementos obligatorios (más los *scripts* extras que son de cada módulo en particular):

Nombre del Archivo	Detalles
mod.html:	Formulario para establecer o actualizar una instancia del módulo.
version.php:	Define alguna meta-información, proporciona código de actualización.
icon.gif:	Un ícono de 16x16 para el módulo.
db/:	Contiene los archivos correspondientes a las tablas y datos requeridos de una base de datos (para cada tipo de base de datos).
index.php:	Página para presentar la lista de todas las instancias en un curso.
view.php:	Página para ver una instancia en particular.
lib.php:	Contiene todas las funcionalidades definidas para el módulo.

Tabla 4. Archivos de los Módulos en Moodle.

Las funciones requeridas en cada módulo incluyen (nombre es el propio del módulo):

nombre_add_instance(): para añadir una nueva instancia del módulo.

nombre_update_instance(): para actualizar una instancia existente.

nombre_delete_instance(): para borrar una instancia.

nombre_user_outline(): dada una instancia, devuelve un resumen de una contribución de un usuario.

widget_user_complete(): dada una instancia, imprime detalles sobre la contribución de un usuario.

Para evitar posibles conflictos, cualquiera de las funciones de un módulo debe ser nombrada comenzando con nombre_ (el nombre del módulo seguido por un guión bajo) y cualquier constante que usted defina debe comenzar con NOMBRE_.

Además de estos archivos, cada módulo porta unos de idioma que contienen cadenas para este; si se desea comenzar un nuevo módulo el sitio oficial de *Moodle* ofrece en la zona de descargas una plantilla de código para desarrollar uno nuevo, se debe descomprimir y seguir las indicaciones que hay en el documento "*README*". De forma general los desarrolladores en *Moodle* se deben guiar por un Manual de Estilo de Código que ofrece el mismo, el cual se encuentra en su sitio de documentación. Mediante el mismo se puede además entender el código del *LMS*.

2.4. DESCRIPCIÓN DE LAS FUNCIONALIDADES

A continuación se realiza una detallada descripción de las funcionalidades seleccionadas que pueden presentar problemas técnicos a la hora de construirlas en la Web.

Interfaz

✓ Diseño de la interfaz

Para el manejo del diseño de la interfaz del sitio Web se puede modificar una plantilla ya existente o crear una nueva, en el epígrafe anterior se describieron los archivos que la componen y su funcionalidad esencial. El diseño de la interfaz es programado mediante el lenguaje *JavaScript*, el cual contiene una buena cantidad de librerías que se emplean en el desarrollo de aplicaciones Web, entre las que se mencionan:

Dojo, que contiene *APIs* y *widgets* (controles) para facilitar el desarrollo de aplicaciones Web que utilizan tecnología *AJAX*²⁷, además de un sistema de empaquetado inteligente, efectos de interfaz de usuario, abstracción de eventos, almacenamiento de *APIs* en el cliente e interacción de estos con *AJAX*. *Dojo* resuelve asuntos de usabilidad comunes como pueden ser la navegación y detección del navegador (19). Se encuentra respaldado por las licencias *BSD*²⁸ y *AFL*²⁹. Es muy eficiente para programar el diseño de las interfaces, pero requiere del uso de *IDEs* avanzados para el desarrollo del contenido Web y no funciona fácilmente en varios *browsers*³⁰ a la vez.

Prototype, que no es más que un *framework* desarrollado en *JavaScript* para el desarrollo sencillo y dinámico de páginas Web. Simplifica gran parte del trabajo cuando se pretende desarrollar páginas altamente interactivas. Puede usarse en *Blogger*³¹ para conseguir algunos efectos muy funcionales en distintas partes del *blog* (19). Este *framework* se encuentra sustentado bajo una licencia *open source* muy bien pensada y diseñada. Su potencial es aprovechado al máximo cuando se desarrolla con *Ruby On Rails*³², por lo que no sería recomendable implementarlo como *framework* en proyectos basados en *PHP*.

MooTools, es un *framework* Web orientado a objetos, desarrollado en *JavaScript*, compacto y modular. Permite escribir un código poderoso y flexible gracias a su elegante, bien documentada y coherente *API* (19). Carece de *widgets* (controles) avanzados. Se encuentra sustentado bajo la licencia *Open Source MIT*³³, posibilitando su utilización y modificación en cada circunstancia.

²⁷ **AJAX**, (**A**synchronous **J**ava**S**cript **A**nd **X**ML), en español, *JavaScript* asíncrono y *XML*, es una técnica de desarrollo web para crear aplicaciones interactivas o *RIA*.

²⁸ **BSD**, (**B**erkeley **S**oftware **D**istribution), es un tipo de licencia para software libre permite el uso del código fuente en software no libre.

²⁹ **AFL**, es una licencia de software libre sin *copyleft* e incompatible con la licencia *GNU/GPL*.

³⁰ **Browsers**, también conocido como navegador, navegador de red o navegador web, es un programa que permite visualizar la información que contiene una página web.

³¹ **Blogger**, es un servicio para crear y publicar un blog de manera fácil.

³² **Ruby On Rails**, es un *framework* de aplicaciones web de código abierto escrito en el lenguaje de programación *Ruby*.

³³ **MIT**, (**M**assachusetts **I**nstitute of **T**echnology), licencia que permite reutilizar el *software* licenciado tanto para ser *software* libre como para ser *software* no libre, permitiendo no liberar los cambios realizados al programa original.

JQuery, librería de *JavaScript* muy rápida y ligera que simplifica el desarrollo de la parte de cliente de las aplicaciones Web. Con esta librería es posible la manipulación de código *HTML*, creación de animaciones, desarrollo de páginas Web con *AJAX* y muchas otras. Es una buena alternativa para *Prototype* y *MooTools*, pues simplifica tareas y optimiza la cantidad de tiempo que se emplea escribiendo códigos frente al monitor (19). Usa las licencias *MIT* y *GPL* permitiendo su empleo en proyectos personales y comerciales.

Ext Core 3.0, esta nueva versión de *framework*, puede ser comparada con otros como *JQuery*, *MooTools*, *Prototype*, debido a que únicamente dispone de las funcionalidades necesarias para gestionar eventos, aplicar estilos *CSS*, todo lo que se necesite de *AJAX* y *JSON*³⁴ y un pequeño *set* de animaciones. Además, es uno de los *frameworks* de *JavaScript* más rápido en la manipulación de los *DOM*³⁵. Esta librería presenta la desventaja de ser muy grande en tamaño. Se encuentra disponible bajo una licencia *MIT*.

Sound Manager 2, esta librería se utiliza en el diseño de la interfaz para cargar el sonido. Presenta una licencia *BSD*.

SWFObject, pequeña librería en *JavaScript* que permite incluir contenido *flash* de forma práctica y sencilla, permitiendo agregar algún otro contenido alternativo en caso de que el navegador no tenga habilitado *JavaScript* o *Flash Player*. Es empleada en el diseño de la interfaz para poder cargar las animaciones o contenidos *flash*. Requiere que el usuario tenga activado *JavaScript* para poder insertar el contenido, si no es así mostrará el contenido alternativo. Esta librería es recomendada únicamente para películas *Flash*. Se encuentra sustentada bajo una licencia *MIT*.

Contenidos

✓ Mostrar los contenidos en formato texto

Los contenidos en el *CMS Joomla* están organizados en secciones, categorías y artículos y son completamente editables y modificables. Estos 3 niveles deben respetarse, una sección contiene

³⁴ **JSON**, (*JavaScript Object Notation*), es un formato ligero para el intercambio de datos.

³⁵ **DOM**, (*Document Object Model*), en español, Modelo en Objetos para la representación de Documentos, es esencialmente una interfaz de programación de aplicaciones que proporciona un conjunto estándar de objetos para representar documentos *HTML* y *XML*.

categorías y cada categoría contiene artículos. Estos se crean directamente en el administrador de *Joomla*, desde el menú **Contenido**, a continuación los pasos a seguir:

1. Definir y crear las secciones (las que engloban todo).
2. Crear las categorías (que pertenecen a las secciones).
3. Introducir la información perteneciente a las secciones/categorías, es decir los artículos: en este caso que se pretende montar un curso no serían más que las conferencias, clases prácticas y demás contenidos que lo conforman. Esto se hace mediante el editor *Wysíwyg*, el cual permitirá darle casi cualquier tipo de formato al texto.

Otra forma de administrar el contenido es mediante los Artículos de Contenido Estático donde la diferencia entre este y un artículo de contenido normal es que el contenido estático:

- No pertenece a una estructura de Sección y Categoría.
- No puede publicarse en la Página de Inicio.

✓ **Mostrar menú de contenidos**

La creación de menús de navegación en el *CMS Joomla* proporciona un método potente y flexible para ayudar a los visitantes a navegar con sencillez. Los menús son módulos y se pueden considerar como grupos de enlaces al contenido.

En la instalación de *Joomla* vienen instalados por defecto:

- *Mainmenu*: es el menú principal del sitio.
- *Topmenu*: este es el menú horizontal que aparece sobre (o debajo) de la cabecera del sitio.
- *Usermenu*: menú de usuario (solo aparece cuando un usuario se registra e ingresa en el sistema).
- *Othertextmenu*: es simplemente "otro menú" cualquiera.

Estos menús se pueden modificar y borrar en caso de que no se necesiten, no se recomienda borrar el principal sino solamente cambiarle el nombre. Esto se hace accediendo a **Menús >> Gestor de menús**. Se pueden crear un número indeterminado de menús, que se visualizarán en Módulos individuales y posicionarlos en cualquier lugar del diseño del sitio.

Cada Artículo de Menú (enlace) se configura independientemente, estos pueden enlazar a Secciones, Categorías, Componentes, Páginas Estáticas y Sitios Externos. También pueden anidarse indefinidamente creando diferentes niveles de menú. Para crearlos se accede a **Menús >>**

Menú principal, el último depende del menú en el cual se agregarán los Artículos de Menú: se debe hacer *click* en el ícono "*nuevo*", ahí elegir el "*tipo*" de enlace o acceso que se quiere crear y eso lleva a un último paso donde se pone el nombre del enlace (el texto), se selecciona el elemento a vincular y demás parámetros asociados, que permiten controlar el modo en que se presenta el contenido de destino como: el orden en que aparecerá, si es un *subítem* de otro elemento del menú, el nivel de acceso de los usuarios a este, entre otros que ayudan a personalizarlo.

Además de todos estos módulos para menús que tiene por defecto el *Joomla*, se han desarrollado otro número de ellos que muestran diversas funcionalidades y apariencias para mejorar el diseño de los sitios:

Menú acordeón, es un módulo sencillo para la versión 1.5.x, que crea un menú usando listas desordenadas y la librería *MooTools*, además brinda la posibilidad de escoger cuales de los menú existentes se mostrará con efecto acordeón.

Menú deslizante Kwick para Joomla 1.5.x, este modulo es muy sencillo, posee un agradable efecto deslizante, ideal para el vínculo con otros sitios pues enlaza a direcciones Web y puede mostrar los enlaces mediante imágenes que trae por defecto, las mismas se pueden sustituir o añadir.

✓ **Mostrar en la primera página las especificaciones y descripción del curso**

Los contenidos que se muestran en la página de inicio son los que corresponden al artículo del menú principal que está marcado como predeterminado, este se puede cambiar al que se desee y además crear uno nuevo y marcar en él esta categoría.

En la configuración predeterminada cuando se instala *Joomla*, los contenidos que se muestran en la página principal se administran mediante el componente Página de Inicio (viene este artículo de Menú en el Menú Principal puesto como predeterminado). Los artículos se pueden asignar a este componente siempre y cuando sea seleccionado el parámetro Mostrar en la Página de Inicio en la página de edición de cada Artículo de Contenido.

✓ **Permitir la creación de palabras calientes que accedan al glosario de términos y galería de imágenes y videos**

El editor de texto *TinyMCE 2.0* que usa *Joomla*, de tipo *WYSIWYG*, permite vincular palabras en el texto a direcciones Web, si a ello se le agrega la función de poder hacerlo a su definición en el glosario de términos y a imágenes y videos situados en la galería, se convierte el término en una

palabra caliente. Se le puede agregar un *tag*³⁶ en el código del editor, donde, al seleccionar una palabra para hacerle el enlace, se muestre además una pestaña en la que se pueda escoger una palabra del glosario o elemento de la galería.

En el proyecto Multisaber está en uso una extensión a este editor para el manejo de las palabras calientes, el cual depende de la existencia de los componentes *EasyGallery* usado para la administración de las medias y *RDGlossary* como glosario de términos, la versión donde trabajan es la 1.0.x de *Joomla*. En el caso del desarrollo del entorno interactivo se desarrollará con la versión 1.x, por lo que esta modificación no se puede usar al no estar disponibles estos componentes para la misma, pero sí se puede emplear como guía de apoyo a la hora de llevar a cabo la modificación. El código de los editores de *Joomla* se encuentran en el directorio *Joomla / plugins / editors*.

Modulo Cuestionario

- ✓ **Configurar de manera general el cuestionario**
- ✓ **Editar categorías en el cuestionario**
- ✓ **Adicionar y editar preguntas en cada categoría**

El **LMS Moodle** es dentro de los Sistemas Gestores de Aprendizaje uno de los que cuenta con los más variados tipos de ejercicios. Existen en el sitio *Pro Moodle.com* componentes y módulos que al instalarlos en el CMS *Joomla* aseguran la integración de ambos, estos son: *com_moodle*, *com_moodleauth*, *com_moologin*: *mod_moodle*, *mod_moodleauth*, además de un manual que explica paso a paso cómo se realiza la misma.

Los contenidos del curso se gestionan a través del *Joomla* y los cuestionarios mediante el *Moodle*. Al integrarlos se inserta el sitio de *Moodle* dentro del *Joomla* a través de un enlace, no se accede directamente a un módulo en particular, en caso de que se desee mostrar solo los cuestionarios, habría que modificar la aplicación; este es un proceso muy engorroso, requiere de un gran dominio del funcionamiento de *Moodle* y se debe tener especial cuidado pues este sistema está muy bien integrado y hay funciones que dependen de otras. Además se debe modificar el aspecto visual para que tenga una mayor armonía con el diseño existente en *Joomla*.

³⁶ **Tag**, es conocida también con el término de **etiqueta** o **baliza**, delimita una región en los lenguajes basados en *XML*.

Otra variante es desarrollar un componente en el *Joomla* que se encargue del manejo de los cuestionarios y preguntas dentro de él, para esto en el epígrafe anterior fueron descritos todos los archivos que se deben incluir en su creación.

Este proceso puede hacerse menos trabajoso si se utilizan las funcionalidades y librerías con las que cuenta el módulo Cuestionario perteneciente al *Moodle*, mediante el cual se administran todas las actividades referentes a los ejercicios de los cursos y contiene características que lo definen:

- Los profesores pueden definir una base de datos de preguntas que podrían ser reutilizadas en diferentes cuestionarios.
- Las preguntas pueden ser almacenadas en categorías de fácil acceso y ser "publicadas" haciéndolas accesibles desde cualquier curso del sitio.
- Los cuestionarios se califican automáticamente y pueden ser recalificados si se modifican las preguntas.
- Los cuestionarios pueden tener un límite de tiempo a partir del cual no estarían disponibles.
- El profesor puede determinar si los cuestionarios pueden ser resueltos varias veces y si se mostrarán o no las respuestas correctas y los comentarios.
- Las preguntas y las respuestas de los cuestionarios pueden ser mezcladas (aleatoriamente) para disminuir las copias entre los alumnos.
- Las preguntas pueden crearse en *HTML* y con imágenes.
- Las preguntas pueden importarse desde archivos de texto externos.
- Los intentos pueden ser acumulativos y acabados tras varias sesiones.
- Pueden crearse diversos tipos de preguntas como: de respuesta corta (palabras o frases), verdadero/falso, emparejamiento, aleatorias, numéricas (con rangos permitidos), incrustada (estilo "*cloze*").
- Pueden crearse textos descriptivos y gráficos.

Este módulo es complejo y no muy simple de entender, aunque cabe destacar que posee gran cantidad de documentación para los desarrolladores y el código disponible se encuentra con abundantes comentarios.

El módulo en sí mismo es solamente responsable del montaje de las preguntas en los cuestionarios y de presentar los resultados. Está ubicado en el directorio principal del *LMS* dentro de la carpeta *mod*, se nombra *quiz* y tiene una estructura de directorios como la mostrada en la Figura 3:

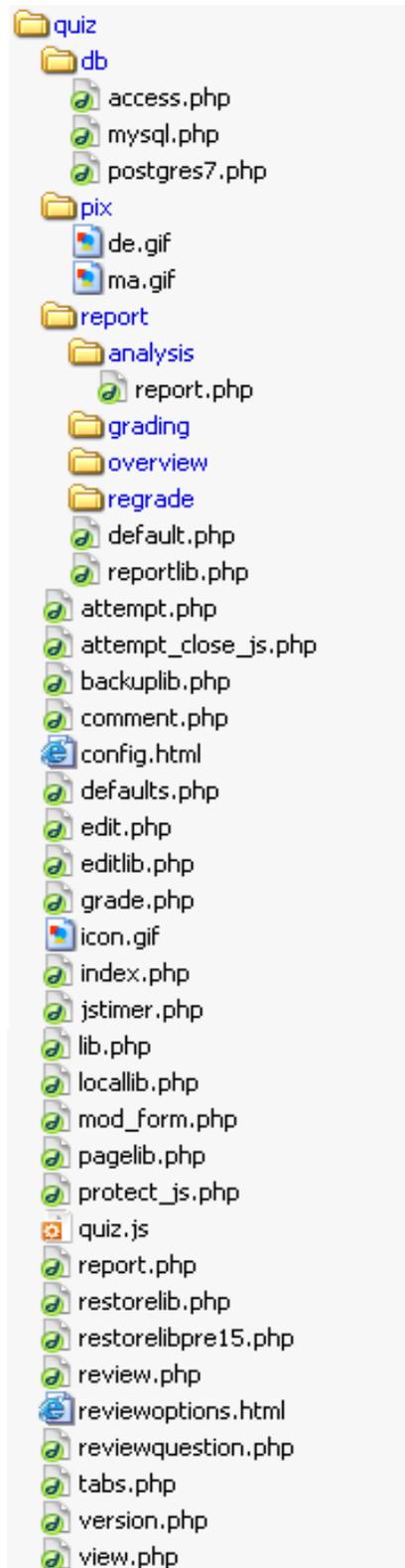


Figura 3. Estructura del directorio del Módulo Cuestionario.

db: Contiene los archivos *PHP* que gestionan la base de datos del *Moodle*.

- *install.xml*, contiene toda la información necesaria para crear la base de datos.
- *upgrade.php*, archivo para posibles actualizaciones de la base de datos.
- *access.php*, controla a qué pueden tener acceso los diferentes grupos (profesores, alumnos, etc.).
- *mysql.php*, *postgres7.php*, en estos archivos se encuentran algunas de las actualizaciones para las bases de datos.

pix: Se almacenan las imágenes usadas en el módulo.

report: Contenedor de los archivos *PHP* encargados de generar los informes del cuestionario, los cuales se encuentran en subcarpetas con el nombre de cada tipo existente:

- *analysis*, muestra los datos procesados estadísticamente del cuestionario a fin de poder analizar y juzgar el desempeño de cada pregunta según su evaluación.
- *grading*, muestra las preguntas que no han sido calificadas posibilitando la evaluación manualmente de las mismas.
- *overview*, muestra una lista de los estudiantes que han realizado el cuestionario.
- *regrade*, recalculará las calificaciones del cuestionario si se ha cambiado su puntuación o la de alguna/s pregunta/s.

Y fuera de todas estas carpetas se encuentran diferentes librerías, dentro de las que se localizan:

- **lib.php:** Funciones que son llamadas a veces por el núcleo de *Moodle*.
- **edit.php:** Empleada para la edición de los cuestionarios y preguntas desde la interfaz del profesor.
- **editlib.php:** Funciones que son utilizadas por la página *edit.php*. Esta carga *locallib.php*.
- **locallib.php:** Todas las funciones que son utilizadas solamente por el módulo Cuestionario. Esta carga *lib.php* y *lib/questionlib.php*.
- **attempt.php:** esta es una de las más complicadas, es la responsable de mostrar las preguntas a un usuario y de evaluar y clasificar su respuesta.
- **jstimer.php:** la cuál imprime un temporizador que cuenta regresivamente y envía las respuestas automáticamente cuando el tiempo establecido se termina.
- **review.php:** dependiendo de los ajustes de la revisión del cuestionario, muestra las preguntas incluyendo las respuestas de los estudiantes y otros elementos como las respuestas correctas.

Para la interpretación de las preguntas y procesar las respuestas del estudiante el módulo Cuestionario del *Moodle* utiliza el motor de preguntas, *Question engine*, ubicado en el directorio principal del *Moodle* en una carpeta nombrada "*question*". Cuando se habla de este motor de preguntas hay ciertos términos que pueden causar confusión pues son usados con diferentes significados, este es el caso de:

Question: es un conjunto de definiciones (nombre de la pregunta, texto, posibles respuestas, puntuación, entre otros.).

Answer: este término es usado específicamente para las respuestas definidas por el profesor a cada pregunta. A las respuestas dadas por los estudiantes se les denomina *responses*.

Attempt: es usado en el sentido de "Intento en un cuestionario".

La estructura del directorio *question* se muestra en la Figura 4:

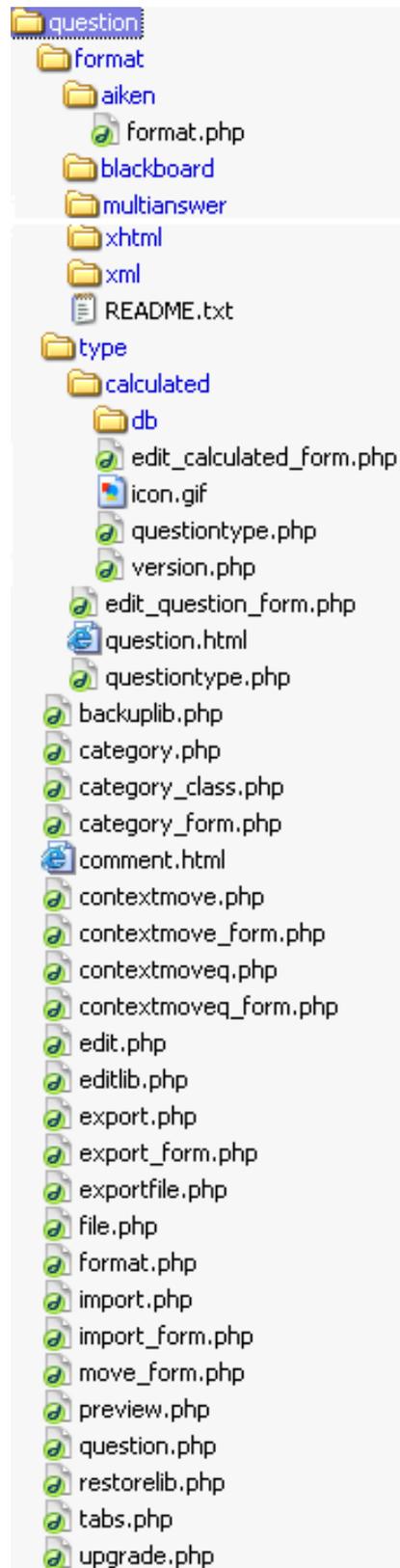


Figura 4. Estructura parcial del motor de preguntas *question*.

type: se ubican las funcionalidades para conformar los diferentes tipos de preguntas, organizadas en subdirectorios nombrados como el tipo y dentro contienen una carpeta **db** con los archivos *PHP* encargados de gestionar la base de datos; y fuera los demás archivos para su configuración en específico, mientras que externamente están los que son usados por todos los tipos en general.

Dentro de los tipos de preguntas que contiene están:

✓ **calculated, “Calculadas”:** Son una generalización del tipo numérico. Presentan una pregunta que requiere que el alumno escriba un número como respuesta. Tanto la pregunta como la respuesta pueden usar una notación algebraica, {a}, {b}, {c}, indicando valores no concretos, variables; estos símbolos serán reemplazados por números arbitrarios cuando la pregunta se le presente a los alumnos (en cada intento del cuestionario). Se puede especificar los intervalos y la distribución estadística usada para generar estos valores aleatorios y la tolerancia o el error de la respuesta del estudiante.

✓ **description, “Descripción”:** Este formato no es en sí una pregunta. Todo lo que hace es mostrar un texto sin requerir respuesta. Es útil para presentar un texto seguido de preguntas.

✓ **essay, “Ensayo”:** Este tipo de pregunta permite que los estudiantes redacten textos de buena longitud con respecto al planteamiento propuesto por el profesor. *Moodle* no califica las respuestas a éste tipo de preguntas de manera cuantitativa como lo hace con las otras. Se requiere que el profesor lea detenidamente la respuesta de sus estudiantes y diga a *Moodle* cuál es la respectiva calificación de dicha respuesta.

✓ **match, “Emparejamiento”:** Presenta dos listas de conceptos. Se deben formar parejas de conceptos afines según el enunciado de la pregunta. Una de las listas se presenta como opciones fijas. A cada una de ellas hay que escogerle un compañero de una lista desplegable. La calificación de la pregunta es automática, en función del número de parejas correctas que se formen.

✓ **multichoice, “Opción múltiple”:** A cada pregunta le sigue una lista de posibles respuestas que el alumno debe escoger. Es posible definir si se admite una única respuesta o si se pueden seleccionar varias opciones. A cada opción se le puede atribuir una calificación. La escala es relativa al valor de la pregunta en el cuestionario. Aunque se permita sólo una repuesta, se pueden otorgar calificaciones parciales si hay una respuesta correcta y otras "casi correctas". También se pueden asignar puntuaciones negativas a las opciones incorrectas.

✓ **shortanswer, “Respuesta corta”**: Este tipo de preguntas requiere que el alumno genere una respuesta por sí mismo, se le presenta la pregunta y un cuadro de texto donde debe introducir su respuesta. Por restricciones de interpretación del computador, las respuestas están limitadas a palabras individuales o a una frase muy concisa.

✓ **numerical, “Numérico”**: Desde la perspectiva del estudiante luce igual que una de respuesta corta, la diferencia está que en este caso se formula una pregunta que requiere una respuesta numérica, expresada en cifras. El alumno debe escribir un número en un cuadro de texto. *Moodle* permite especificar el valor exacto correcto que corresponde al cálculo solicitado, pero también permite especificar un margen de error.

✓ **truefalse, “Verdadero/falso”**: el alumno debe seleccionar una de las opciones: verdadero o falso, la calificación de la pregunta se concede al 100% sólo a la opción considerada correcta. Ambas opciones, pueden tener asociado un texto de retroalimentación.

Los demás archivos que están fuera de las carpetas antes mencionadas son las demás librerías que se emplean en la gestión de las preguntas. Además de estas dentro de la carpeta *lib* del directorio principal se encuentra *questionlib.php* que contiene todas las funciones necesarias para usar las preguntas.

Para entender como funciona este motor de preguntas es clave conocer como trabajan juntos los diferentes tipos de objetos. Los más importantes son:

- *Questions* (en español preguntas)
- *Attempts* (en español intentos)
- *States* (en español estados)

Questions son los datos creados por el profesor. *Attempts and States* son los datos creados por el estudiante cuando interactúan con un cuestionario.

Los estudiantes tienen varios intentos para hacer un cuestionario. El motor de preguntas está optimizado para lidiar con ellos y los datos sobre estos intentos son guardados en un objeto *attempt*; este puede contener información de cómo el cuestionario fue organizado, el orden de las preguntas y respuestas. Pueden también interactuar repetidamente con una sola pregunta, cuando el estudiante interactúa con una pregunta que ha salido por primera vez en un intento se crea un *state*.

Ciertamente el *Question engine* no está escrito orientado a objetos, solo los objetos referentes a los tipos de preguntas, ubicados dentro de la carpeta *type* tienen métodos. El módulo Cuestionario inicializa un objeto de estas clases al comenzar y reutiliza sus métodos para las diferentes preguntas.

Toda la información que se maneja en los cuestionarios necesita guardarse en la base de datos, el *Moodle* antepone a los nombres de las tablas '*quiz_*' para las que se emplean por el módulo Cuestionario y '*question_*' para las que usa el motor de preguntas. Cuando se crea el componente para *Joomla* y se van a utilizar las funcionalidades que brinda este sistema se debe asegurar que estas tablas existan en la base de datos para que no ocurran errores al consultarlas. Para ello se puede instalar el *Moodle* y luego importar estas tablas a la base de datos del *Joomla*.

Cuando se utilizan las funciones y librerías que ofrece el *Moodle* para la confección del componente en *Joomla*, se pueden generar errores de dependencia a otras que se encuentran fuera de los directorios del motor de preguntas y del módulo *quiz*, se debe analizar si es más factible incluir estas funcionalidades de las que depende o implementarlas.

✓ **Mostrar los resultados de los estudiantes**

Para la primera variante en que se integran ambos sistemas, los resultados se pueden mostrar mediante el *Moodle*, el profesor accede a un cuestionario dentro del curso y se le muestra un listado de los estudiantes que lo iniciaron, fecha en que lo culminaron, tiempo empleado y la calificación. En la segunda variante se puede incluir esta funcionalidad en el componente que gestione todo lo referente a los cuestionarios; o construir un módulo, que tome la información a mostrar de la Base de Datos y así se puedan visualizar los resultados de los estudiantes.

Glosario de Términos

✓ **Acceder al Glosario de Términos**

En los glosarios se incluyen las palabras de difícil comprensión relacionadas con los temas abordados en el *software* y las palabras calientes que aparecen en los distintos módulos, las cuales ofrecen el significado de las mismas y en muchos casos ilustrados a través de imágenes. Para la administración de un glosario de términos en *Joomla* existen diferentes extensiones tales como:

Abbrev: mediante este componente se especifica el término y la descripción de la palabra y luego mediante un enlace se mostrarán los mismos en forma de columnas: de un lado los términos y del otro la definición, no los agrupa por letra ni organiza alfabéticamente, si no en el orden en que fueron

creados. Este componente puede servir pues es muy sencillo, lo que se tendría que mejorar la forma en que se le muestra al usuario. Contiene además dos *plugin* que aumentan sus funcionalidades, dentro de estas la búsqueda dentro del glosario.

Glossary: este componente permite la creación de varios glosarios de términos, insertar los términos en el glosario al que corresponda acompañado por su descripción y otros parámetros como creador y fecha. Coloca automáticamente un vínculo en el menú principal que lleva a la búsqueda de los términos en el glosario. En la administración del componente se pueden especificar la cantidad de términos que se mostrarán en cada página cuando se realicen las búsquedas, en donde se mostrará el glosario en el sitio y demás aspectos de configuración.

Galerías de Imágenes

✓ Acceder a la galería

Para montar una Galería de Imágenes en el *CMS Joomla* existen extensiones que lo facilitan, estas difieren acorde a las funcionalidades que permiten, cargar y borrar diapositivas; añadir una lista de imágenes a la galería; descripción de la imágenes; borrar la imagen que se selecciona; publicar, cambiar el orden y eliminar categorías. Dentro de estas se encuentran:

Boncko PicLens Image Gallery: *plugin* que permite mostrar una galería de imágenes dentro de un artículo, donde se escribe `{boncko}/fotos/{/boncko}` (siendo fotos el nombre de la carpeta que contiene las imágenes) y se mostrará la galería. Dentro de la edición del *plugin*, se especifican entre otros parámetros el tamaño de la *thumbnails*³⁷ que se usa como portada de la galería y el directorio principal donde se encuentran las imágenes. Una vez mostrada es posible poder verla en pantalla completa, reproducirla y detenerla.

Con este *plugin* se pueden tener las imágenes agrupadas por categorías, en el directorio principal se ubican en carpetas separadas. La organización por categorías facilita la navegación y que el acceso a las imágenes se haga de manera rápida y sencilla. En un mismo artículo se puede llamar a diferentes categorías de la galería.

³⁷ **Thumbnails**, Se refiere a pre visualizaciones de menor tamaño de una imagen original lo cual se utiliza mucho para galerías.

Power Slide: módulo al cual se le especifica el directorio dentro del gestor de multimedia del *Joomla* donde se encuentran las imágenes a mostrar, la posición en que se mostrará y demás parámetros. Se visualiza mediante una lista de *thumbnails* en la parte superior que aparece oculta inicialmente y se puede desplegar para escoger la imagen que se desea mostrar.

Para el montaje de una galería de imágenes para un curso solo es necesario ajustarse a los requerimientos de la misma, existen algunas galerías que por ofrecer una mejor interfaz y mayores funcionalidades llegan a sobrecargar el sitio y mostrar opciones que no son necesarias para el curso.

Galerías de Videos

- ✓ **Acceder a la Galería de Videos**
- ✓ **Permitir la reproducción del video, su pausa, detención y control de su barra de estado y volumen**

Cuando se va a montar una galería de videos hay que tener en cuenta la instalación de componentes y módulos que facilitan su creación en el *CMS Joomla*. Entre los que existen para ello se mencionan los siguientes:

Dailymotion Video Zap: módulo que ofrece una nueva manera de controlar reproductores de vídeo en sitios Web. Permite seleccionar la lista de video que se desea reproducir o introducir una *URL*, se especifica la posición y el tamaño en que se mostrarán. Los videos son presentados en forma de tabla definiendo antes las columnas y filas que contendrá.

VJdeo: componente que permite agrupar los videos por categorías y subcategorías, añadir una imagen en cada una de ellas y su descripción. Para subir los videos que se ubicarán en cada categoría se debe hacer mediante un cliente *FTP*³⁸ o un componente de Joomla que se encargue de la gestión de archivos como el *Explorer*.

Simple Video Flash Player: módulo diseñado específicamente para *Joomla* 1.5.x, brinda la posibilidad de visualizar el video en el tamaño que se desee, lo cual permite su colocación en cualquier parte del sitio. Se puede reproducir un solo video o especificar una lista de reproducción.

³⁸ **FTP (File Transfer Protocol)**, en español **Protocolo de Transferencia de Archivos**, es un protocolo de red para la transferencia de archivos entre sistemas conectados a una red TCP, basado en la arquitectura cliente-servidor.

Este módulo también permite que los videos se puedan reproducir, pausar, detener y además, controlar su barra de estado y volumen, debido a que tiene incluido un reproductor de video.

Al igual que en las Galerías de Imágenes, en el caso de los videos se deben limitar también las funcionalidades de las extensiones a las que en realidad hagan falta, lo ideal sería que existiera un componente que gestionara ambas medias; esto, al disponer del código de extensiones que lo hacen por separado hace que sea más sencillo el poder desarrollarlo.

Audio

✓ Reproducir música de fondo, poder detenerla en cualquier momento

Para permitir la reproducción de audio en el *CMS Joomla 1.x* existen componentes y módulos que se especializan en ello, entre estos:

PremiumBeat: módulo que permite la reproducción de música, pero presenta un inconveniente y es que para su configuración es necesario editar un *XML* donde se especifica la ruta del archivo musical.

UnMP3: compuesto por un módulo y un componente mediante el cual se pueden cargar fácilmente los archivos *MP3* al sitio, es altamente configurable e intuitivo al permitir editar la lista de reproducción, personalizar el reproductor especificando el color de fondo, bordes, transparencia y entre otros aspectos el volumen inicial.

Si estos reproductores no cubren todas las perspectivas se pueden modificar al disponer del código fuente o programar una clase mediante la cual se permita la reproducción y detención de temas musicales, que estos se reproduzcan continuamente y demás funcionalidades que se necesiten, esto puede ser logrado gracias a la librería *SoundManager2*³⁹ que contiene 2 archivos: un *API* en *JavaScript* y una pequeña película en *Flash 8* que provee de las capacidades de audio.

Al combinar los dos archivos que contiene la librería *SoundManager2*, se pueden manejar los sonidos directamente con *JavaScript*, sin necesidad de editar ninguna película *Flash* (aunque se

³⁹ **Sound Manager 2**, visitar el sitio web Artistechmedia.com para más información.

incluye el código *ActionScript* por si se desea cambiar). Esta librería posee métodos como *soundManager.createSound(sonidoID, URL)* para cargar el sonido, *soundManager.play(sonidoID)* para reproducirlo, *soundManager.stop(soundID)* que detiene el audio especificado, *soundManager.onload = function() { (...) }* que se necesita para que *soundManager* esté listo para usarse, entre otras que se usan dependiendo de las necesidades de cada cual.

Portabilidad

✓ La aplicación debe funcionar tanto desde el CD como del Disco Duro (HDD)

Una de las funcionalidades más importante de las multimedias es su portabilidad, ahora bien para que la aplicación Web elaborada con el CMS *Joomla* sea portable y se pueda acceder a ella desde un CD, existen algunos programas libres para *Windows*⁴⁰:

DWebPro (*Dynamic Web Server*) que no es más que un servidor Web dinámico gratuito, totalmente independiente, que se puede ejecutar por sí mismo y que está diseñado para poder utilizarse en distribuciones CD/DVD de sólo lectura. Incluye soporte para CGI⁴¹, ISAPI⁴², MySQL, software externo y para servidores SMTP⁴³.

Con este servidor Web, el usuario puede crear presentaciones dinámicas en CD y sitios Web que no se limiten simplemente a mostrar páginas, sino que además incluyan cualquier elemento soportado por los lenguajes *scripts*. Algunas opciones extra son la interfaz personalizable, el ícono accesible desde la bandeja del sistema y un compilador para codificar *scripts PHP*.

Server2go, servidor Web que puede ser ejecutado sin necesidad de ser instalado, lo que significa que, las aplicaciones Web basadas en este servidor se pueden utilizar directamente desde un CD,

⁴⁰ **Windows**, es un sistema operativo desarrollado y comercializado por *Microsoft*, el cual presenta un modelo de desarrollo basado en *software* no libre.

⁴¹ **CGI**, (*Common Gateway Interface*), en español Interfaz de entrada común, es una importante tecnología de la *World Wide Web* que permite a un cliente solicitar datos de un programa ejecutado en un servidor web.

⁴² **ISAPI**, (*Internet Server Application Programming Interface*), permite que los programadores puedan desarrollar aplicaciones basadas en web que se procesen mucho más rápidamente que los programas CGI.

⁴³ **SMTP**, (*Simple Mail Transfer Protocol*), en español Protocolo Simple de Transferencia de Correo, es un protocolo de la capa de aplicación.

una memoria *USB* o desde cualquier carpeta en un disco duro. Permite la creación de un sitio Web o aplicación *PHP* en un *CD*.

No existe ningún programa para que se puedan ejecutar sitios Web desde un *CD* en distribuciones de *Software Libre*, es por ello que en la UCI, el proyecto GALM se encuentra en estos momentos desarrollando un **Instalador Libre para Software Educativo**, mediante el cual esto puede ser posible. Esta aplicación permitirá instalar colecciones educativas desarrolladas en la Web de diferentes formas:

- *Servidor*, en caso de que se desee que una máquina contenga el sitio y otras accedan a esta.
- *Cliente*, para las *PCs* que acceden al servidor, donde se especifica la dirección de este y se crea un ícono de acceso directo a la misma.
- *Local*, si se desea que la *PC* funcione como servidor y cliente a la vez.
- Y una última que permita que el sitio corra desde el mismo *CD*.

Apariencia

✓ **Mostrar y saltar presentación del curso**

La presentación del curso es generalmente una pequeña animación, los tipos más usados en entornos web son: *.swf*, *.mov*, *.wmv* y *.mpeg*; dentro de estos resalta el *.swf* al poder mostrar la misma información gráfica con calidad de una forma más compacta y ser compatible con el *Flash Player* presentado en todos los navegadores. Cuando se encuentra en reproducción, los eventos que se ejecuten sobre el navegador son capturados por el reproductor de la animación y no llegan a la aplicación Web, lo que influye a la hora de escoger *.swf* para representar los cursos, pues este formato tiene mayor facilidad de comunicación con *JavaScript* mediante el cual se puede manejar este evento y otros, como por ejemplo poder detenerlo en algún momento.

Para la elaboración de las animaciones *.swf* se usan herramientas propietarias, para resolver este problema lo primero sería el cómo desarrollar estas pequeñas animaciones *Flash* con *Software Libre* y existen diversas formas, una de las más complejas es mediante el uso de las herramientas libres *SWFMILL* que genera estructuras *.swf*, partiendo de un *XML*, elementos externos y *MTASC* que es un compilador de *Software Libre* para *ActionScript 2.0*, el cual inyecta el código necesario para formar el archivo *.swf* al crear una clase nueva y compilar el archivo *XML* creado con *Swfmill*; esta solución requiere de un gran trabajo y conocimiento del lenguaje *ActionScript*.

Existen algunas herramientas gratuitas para desarrollar animaciones *Flash*: *SALASAGA* que proporciona herramientas para la creación de video tutoriales y/o presentaciones interactivas, genera archivos en formato *Flash* y se pueden incluir imágenes o realizar capturas desde la misma interfaz del programa, además, se puede añadir movimiento de ratón, texto y resaltado; lo que la hace más que una herramienta para la creación de tutoriales interactivos.

Otra herramienta que se recomienda para hacer animaciones de introducción a los cursos es *KTOON* y esto se debe a que fue ideada para el desarrollo de animación en 2D, engloba los procesos de creación y diseño de las animaciones y la mezcla de los mismos, además de incorporar elementos tan necesarios como el soporte de capas o una línea temporal, por ahora sólo está disponible para sistemas *Unix*⁴⁴, pero se espera que sea portado a otros sistemas operativos también.

Luego de tener confeccionada la animación se debe pasar a incluirla en la Web y para esto es muy usada la librería *SWFObject*, la cual permite incluir archivos *.swf* en páginas *HTML* de manera limpia, estándar y compatible para todos los navegadores. El primer paso es descargar la librería, la cual es gratis y copiar el archivo dentro del sitio Web. Luego en la página principal se inserta la siguiente línea antes del cierre del *tag HEAD*:

```
<script type="text/javascript" src="swfobject.js"></script>
```

Lo próximo es insertar un *DIV* el cual contendrá el *.swf*. Este *DIV* debe tener un *ID* el cual lo identificará para mas adelante asociarlo al *JavaScript* que insertará el *.swf*. En el siguiente ejemplo el *ID* es *contenidoflash*:

```
<div id="contenidoflash">  
    // Mensaje si no existe el plugin de flash.  
</div>
```

Finalmente la inserción del *flash* se realiza con el siguiente código:

```
<script type="text/javascript">  
var so = new SWFObject("anim.swf", "animacion", "500", "100", "6", "#003399");
```

⁴⁴ **Unix**, es un sistema operativo portable, multitarea y multiusuario.

```
so.write("contenidoflash");  
</script>
```

En donde se tiene:

"**anim.swf**": ruta del *.swf* a insertar.

"**animación**": identificador del *.swf* a insertar, esto se usará para procesarlo con *JavaScript* en caso se necesite.

"**500**": ancho de la película a insertar.

"**100**": alto de la película a insertar.

"**6**": versión del *Flash Player* necesario para reproducir la película.

"**#003399**": es el color de fondo de la película insertada.

En caso de que se necesiten añadir parámetros adicionales a la página, esto se hace con el uso del comando **addParam**. Entre los parámetros se mencionan:

- **useExpressInstall**: Si desea permitir a los usuarios actualizar se utiliza la característica *ExpressInstall*, asignándole *'true'* a este valor.
- **Quality**: La calidad de reproducción de la película *flash*. Si no indica este valor, la calidad por defecto es alta: *"high"*.
- **xiRedirectUrl**: Si desea redirigir usuarios que completen la instalación *expres* (*ExpressInstall*), puede especificar una *URL*⁴⁵ alternativa aquí.
- **detectKey**: Este es el nombre de la variable de *URL* que el *script SWFObject* buscará para sobrepasar la detección. Por defecto es *'detectflash'*. Ejemplo: Para sobrepasar la detección *Flash* y simplemente escribir la película *Flash* directamente en la página, se podría añadir *?detectflash=false* a la *URL* del documento que contenga la película *Flash*.

✓ **Permitir cerrar la aplicación en cualquier momento**

El que se pueda cerrar la aplicación al dar *click* en un botón ubicado en ella suele ser muy sencillo. En una clase principal que sería donde se ubiquen los objetos y funciones de la página inicial, se puede implementar un método donde se llame a la función *window.close()*, esta cerrará el navegador

⁴⁵ **URL**, (*Uniform Resource Locator*), en español, localizador uniforme de recurso. Es una secuencia de caracteres, de acuerdo a un formato estándar, que se usa para nombrar recursos, como documentos e imágenes en Internet, por su localización.

Web, aunque hay algunos que vienen por defecto para que no permita esto como en el caso del *Mozilla*⁴⁶, pero ya sería propio de cada navegador en específico, de todas formas se pueden implementar variantes como el uso del objeto *history*, el cual contiene la información acerca de las *URL* que ha visitado con su navegador en la actual sesión, esto permite el poder moverse hacia atrás o adelante y con esto regresar a una página abierta anteriormente o se podría cargar una página en blanco.

2.5. ESTRUCTURA DE MÓDULOS

El diseño de los cursos a distancia requiere trazar estrategias acerca de los pasos y las directrices a seguir, para ello se aboga por la creación de un entorno donde el estudio se desarrolle de manera amigable y casi inconsciente. Lo elemental no es el uso de las TIC en estos, sino cómo se produce el proceso de aprendizaje. La disposición de los contenidos no es suficiente, se debe ofrecer una secuencia de actividades que planteen qué hacer con dichos contenidos.

Para facilitar el trabajo en paralelo el producto se encontrará dividido en módulos, capaces de responder a las necesidades de los cursos de segundo perfil que ayudan a la correcta elaboración de *software* de este tipo, estos son: **Contenido**, **Galería de Imágenes**, **Galería de Videos** y **Cuestionario**.

2.6. ORGANIZACIÓN ARQUITECTÓNICA

Con la propuesta de una organización arquitectónica de la forma en que se deba implementar se desea lograr una estandarización de todos los productos que se quieran desarrollar más adelante. La arquitectura de la aplicación deberá estar basada en el patrón de diseño *MVC*, al ser la usada en el *CMS Joomla* sobre el cual se trabajará.

La estructura de paquetes de la aplicación responderá en todo momento a la necesidad de crear una división que permita el trabajo en paralelo. El objetivo fundamental de la estructuración elegida es disminuir el tiempo de desarrollo así como incrementar la facilidad para mantener y modificar la aplicación. En la Figura 5 se muestra la forma en que se organizan los archivos referentes al

⁴⁶ **Mozilla Firefox**, es un navegador de Internet multiplataforma con código fuente es software libre.

directorio principal de *Joomla* respetando el patrón *MVC*, establecido en la arquitectura base del *CMS* sobre el cual se desarrollará la aplicación.

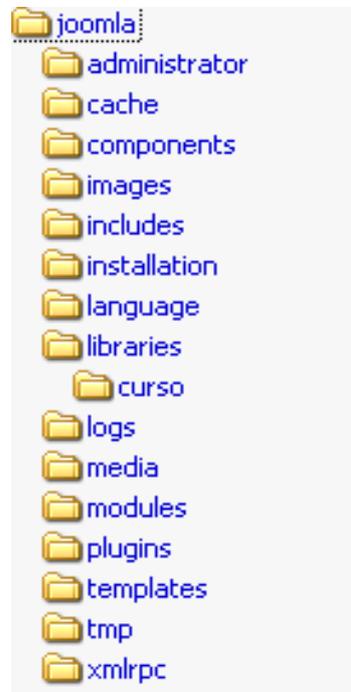


Figura 5. Estructura del directorio principal de Joomla.

El directorio *libraries* en la versión 1.x de *Joomla* es la columna vertebral y en él residen casi todas las clases siguiendo una jerarquía lógica, contiene todas las librerías usadas así como los objetos de acceso a datos. Si se le hace una modificación a algún elemento del *Joomla* que no sea la plantilla, dentro de este directorio se guardarán todas estas funcionalidades dentro de una carpeta con el nombre del curso.

En el caso de la plantilla del curso mediante la cual se diseña la interfaz del sitio, es decir su aspecto visual, tendrá una estructura como la mostrada en la Figura 6.

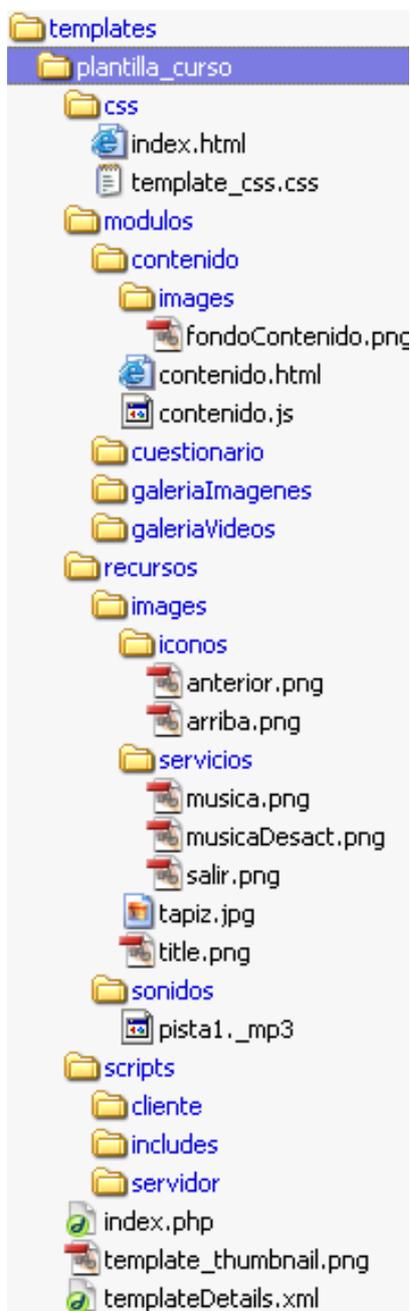


Figura 6. Estructura de la plantilla.

Al principio del capítulo se especificaron los archivos que componen una plantilla en *Joomla*, a la hora de hacer la propia del curso es necesario que además de estos se agreguen unas carpetas para la correcta organización del trabajo y poder manejar, modificar y estandarizar los nuevos archivos que sean necesarios crear, para satisfacer las funcionalidades descritas anteriormente y otras con respecto al diseño que se quieran agregar.

css: carpeta que contendrá las Hojas de Estilos en Cascada (CSS), las cuales contienen el código que define el estilo visual del sitio Web, ejemplo: tamaños de letra, colores. Pueden existir tantos archivos CSS como se desee.

módulos: dentro de esta carpeta se situarán las funcionalidades que usa la plantilla específicamente para cada módulo y las imágenes respectivas a ellos.

recursos: este directorio contendrá las imágenes, sonidos y videos que se cargarán en la plantilla:

- **images:** se agruparán dependiendo de su funcionalidad.
- **sonidos:** en caso de que se manejen los sonidos de la música de fondo directamente con *JavaScript*, estos estarán ubicados dentro de la carpeta sonidos. Si se usa para ello el componente *UnMP3* este los ubicará automáticamente dentro del directorio *administrator >> components >> com_unmp3 >> songs*.

scripts: este directorio contiene tres carpetas:

- **cliente,** dentro de la cual se ubicarán las librerías que se ejecutan en el cliente, este es el caso de la librería *Dojo* utilizada para facilitar el desarrollo de la aplicación Web, la *SWFObject* para el montaje de la presentación del curso y la *SoundManager 2* si se emplea para el manejo de la música de fondo; estas librerías se pondrán cada una en una carpeta nombrada como ella. Además las clases *JavaScript*, que permiten la navegación entre los módulos, cambiar de secciones de código en la página activa posibilitando que se carguen o descarguen secciones en esta y restantes funcionalidades creadas que se ubicarán juntas en una carpeta con el nombre del curso.
- **includes:** se ubicarán las secciones de páginas que contienen aspectos generales de la plantilla, utilizadas para lograr la navegación global en el entorno.
- **servidor:** que contendrá los archivos *PHP*, clases y librerías que se tengan que ejecutar del lado del servidor, estas últimas son las que gestionan datos de la base de datos generada en *Joomla*.

2.7. CONCLUSIONES PARCIALES

En este capítulo se llevó a cabo el estudio de las funcionalidades del *Moodle* y las antiguas Multimedias con el fin de confeccionar las que responderán al entorno interactivo de aprendizaje que se propone, las cuales fueron agrupadas para una mejor comprensión. Luego se seleccionaron las herramientas *CMS Joomla* y *LMS Moodle* para el desarrollo del mismo. Después de su estudio se

describieron las funcionalidades basadas en el empleo de estas herramientas, con el fin de que sirva como guía para los desarrolladores y que los mismos conozcan los elementos que las componen y extensiones que se emplean para cubrir las funcionalidades.

Finalmente se planteó la estructura de los módulos que regirá el paquete y una organización arquitectónica de la forma en que se debe implementar el producto, con el fin de lograr una estandarización de todos los que se quieran desarrollar más adelante.

Capítulo 3

Análisis de los Resultados

En el presente capítulo se analizan los resultados de la investigación realizada; mediante el método de Expertos, el cuál se basa en la evaluación cuantitativa de criterios previamente definidos por parte de expertos en el tema. Seguidamente se describe la forma en que se aplica este método y los elementos necesarios para el mismo, posteriormente se presentan los resultados obtenidos de la evaluación.

3.1. MÉTODO DE EXPERTOS

El método de Expertos se basa en la consulta a personas que tienen grandes conocimientos sobre el entorno en el que la organización desarrolla su labor. Estas personas exponen sus ideas y finalmente se redacta un informe en el que se indican cuáles son, en su opinión, las posibles alternativas que se tendrán en el futuro (20).

3.2. APLICACIÓN DEL MÉTODO DE EXPERTOS

Para llevar a cabo Método de Expertos se efectúan un conjunto de pasos:

Paso 1: Elaborar los criterios de evaluación en función de las características concretas de la propuesta y organizarlos por grupos, quedando de la siguiente manera:

Grupo No. 1: Criterios de mérito científico

1. Valor científico de la investigación.
2. Calidad de la investigación.
3. Novedad científica de la investigación.

Grupo No. 2: Criterios de implantación

4. Necesidad del empleo de la propuesta.
5. Posibilidad de aplicación de la propuesta.
6. Integración de la propuesta al proceso de enseñanza-aprendizaje en la UCI.

Grupo No.3: Criterios de flexibilidad

1. Adaptabilidad al proceso de enseñanza-aprendizaje de la UCI.
2. Empleo de las herramientas necesarias para la elaboración de las funcionalidades.
3. Velocidad de desarrollo de la propuesta.

Grupo No.4: Criterios de impacto

1. Repercusión en el proceso de enseñanza-aprendizaje de la UCI.
2. Organización en el proceso de documentación de la investigación.
3. Impacto en el área para la cual está destinada la investigación.

CAPÍTULO III ANÁLISIS DE LOS RESULTADOS

Paso 2: Asignar a cada grupo de criterios un peso relativo en dependencia del porcentaje que representa cada uno del total y los intereses a evaluar. El peso asignado a cada grupo es el siguiente para sumar un total de 100:

Grupo No. 1.....25

Grupo No. 2.....25

Grupo No.3.....25

Grupo No.4.....25

Paso 3: Organizar un comité de expertos donde se convoquen como mínimo 7 expertos. Estos fueron seleccionados teniendo en cuenta su especialidad, grado científico y currículum.

Paso 4: Entregar a cada experto la propuesta para que estudien el tema a evaluar, un cuestionario relacionado con la propuesta (Ver Anexo 1) y 2 modelos: en el primero deben valorar el peso relativo de cada criterio (Ver Anexo 2) y otro para realizar una evaluación cuantitativa de los criterios en una escala del 1 al 5 (Ver Anexo 3), luego deben registrar su apreciación cualitativa de la propuesta con una clasificación final que puede ser: excelente, bueno, aceptable, cuestionable y malo; además pueden registrar sus consideraciones respecto a la propuesta.

Paso 5: Determinar por cada criterio el peso promedio, a partir de los pesos dados por los expertos (Ver Anexo 4). El peso promedio de cada criterio es el siguiente:

G	Expertos/Criterios	E1	E2	E3	E4	E5	E6	E7	Exp. P
25	C ₁	10	8	8	8	7	8	8	8.14
	C ₂	10	10	9	9	8	9	9	9.14
	C ₃	5	7	8	8	10	8	8	7.71
25	C ₄	5	5	9	7	10	10	10	8.00
	C ₅	10	10	9	9	5	8	8	8.43
	C ₆	10	10	7	9	10	7	7	8.57
25	C ₇	8	6	9	9	7	8	9	8.00
	C ₈	9	10	8	9	9	9	9	9.00
	C ₉	8	9	8	7	9	8	7	8.00

25	C₁₀	9	10	8	8	10	10	9	9.14
	C₁₁	8	8	9	8	5	8	8	7.71
	C₁₂	8	7	8	9	10	7	8	8.14
T		100	100	100	100	100	100	100	100

Tabla 5. Peso promedio/Criterio.

Paso 6: Verificar la consistencia en el trabajo de los expertos, utilizando el coeficiente de concordancia de *Kendall* y el estadígrafo *Chi* cuadrado (X^2) (Ver Anexo 5). En caso de no existir concordancia en el trabajo de los expertos, este debe repetirse.

Dado C el número total de criterios a evaluarse y E el número de expertos involucrados en la evaluación, se realiza el siguiente procedimiento para determinar la consistencia en el trabajo de los expertos:

- ✓ Calcular para cada criterio: ΣE , que representa la sumatoria del peso dado por los expertos.
- ✓ Determinar el valor de Exp. P: puntuación promedio de cada criterio.
- ✓ Se calcula el peso medio de cada criterio $M\Sigma E = \Sigma E / C_n$.
- ✓ Hallar el valor de ΔC , que representa la diferencia entre ΣE y $M\Sigma E$.
- ✓ Determinar la desviación de la media, que posteriormente se eleva al cuadrado para obtener la dispersión S , dada por la expresión: $S = \Sigma (\Sigma E - \Sigma \Sigma E / C)^2$.
- ✓ Conociendo la dispersión se puede calcular el coeficiente de concordancia de Kendall W , dado por la expresión: $W = S / E^2 (C^3 - C) / 12$.
- ✓ Calcular el Chi cuadrado real a partir del valor del coeficiente de Kendall teniendo en cuenta la siguiente expresión: $X^2 = E (C-1) W$.

A continuación se muestran los datos obtenidos luego de realizar los pasos anteriores.

Expertos/Criterios	E ₁	E ₂	E ₃	E ₄	E ₅	E ₆	E ₇	ΣE	Exp. P	ΔC	ΔC^2
C₁	10	8	8	8	7	8	8	57	8.14	1.33	1.77
C₂	10	10	9	9	8	9	9	64	9.14	5.67	32.15
C₃	5	7	8	8	10	8	8	54	7.71	4.33	18.75
C₄	5	5	9	7	10	10	10	56	8.00	2.33	5.45

CAPÍTULO III ANÁLISIS DE LOS RESULTADOS

C₅	10	10	9	9	5	8	8	59	8.43	0.67	0.45
C₆	10	10	7	9	10	7	7	60	8.57	1.67	2.79
C₇	8	6	9	9	7	8	9	56	8.00	2.33	5.43
C₈	9	10	8	9	9	9	9	63	9.00	4.67	21.81
C₉	8	9	8	7	9	8	7	56	8.00	2.33	5.43
C₁₀	9	10	8	8	10	10	9	64	9.14	5.67	32.15
C₁₁	8	8	9	8	5	8	8	54	7.71	4.33	18.75
C₁₂	8	7	8	9	10	7	8	57	8.14	1.33	1.77
DC	100	100	100	100	100	100	100	700	100	36.66	146.7
MΣE	58.33										
W	0.021										
X²_{real}	1.617										

Tabla 6. Tabla para el cálculo de concordancia de Kendall y el estadígrafo Chi cuadrado.

Posteriormente, se compara el X^2 real, con el valor del dato estadístico, siendo $\alpha=0.5$ y $C = 12$ y debe cumplirse que $X^2 < X^2(\alpha, c-1)$ para que el trabajo realizado por los expertos sea valorado de consistente. $X^2(\alpha, c-1) = X^2(0,5, 11) = 10,3410$.

1.617 < 10,341, quedando demostrada la consistencia en el trabajo realizado por los expertos.

Paso 7: Calcular $P \times c$ donde "P" es el peso relativo de cada criterio y c es la calificación promedio dada por los expertos (Ver Anexo 6).

P: peso promedio de cada criterio.

$$P = (\text{Exp. P})/100$$

c: calificación promedio de cada criterio concebida por los expertos.

Criterios	Calificación (c)					P	P x c
	1	2	3	4	5		
C₁				x		0.81	3.24
C₂			x			0.91	2.73
C₃			x			0.77	2.31
C₄					x	0.80	4.00
C₅				x		0.84	3.36

C₆				x		0.86	3.44
C₇			x			0.80	2.40
C₈				x		0.90	3.60
C₉			x			0.80	2.40
C₁₀				x		0.91	3.64
C₁₁				x		0.77	3.08
C₁₂			x			0.81	2.43

Tabla 7. Cálculo de P*c.

Paso 8: Se calcula el índice de aceptación de la propuesta.

Se calcula a partir de la siguiente fórmula teniendo que $\sum (P * c) = 36,63$

$$IA = \sum (P * c) / 5.$$

Al despejar los valores, obtener el índice de aceptación:

$$IA = 7,326$$

Paso 9: Se determina la probabilidad de éxito de la propuesta.

Se determina a partir de los rangos predefinidos del índice de aceptación:

IA > 0,7.....Existe alta probabilidad de éxito.

0,7 > IA > 0,5.....Existe probabilidad media de éxito.

0,5 > IA > 0,3.....Existe baja probabilidad de éxito.

0,3 > IA.....No existe probabilidad ninguna de éxito.

Dado el resultado de IA es igual a **7,326** entonces se puede concluir que la probabilidad de éxito es: ALTA.

Además, es establecida por los expertos una valoración final de: Aceptable.

3.3. CONCLUSIONES PARCIALES

Con el empleo del Método de Expertos se llega a la conclusión de que la investigación realizada puede ser aplicada, puesto que se declara aceptable según las valoraciones de los expertos, que además confirmaron la probabilidad de éxito de la misma.

CONCLUSIONES GENERALES

Al culminar con la investigación se llegaron a las siguientes Conclusiones Generales:

- ✓ Mediante la realización de un estudio de las multimedias de apoyo a los cursos de perfil realizadas hasta el momento en la facultad 8 y del EVA disponible en la UCI, se pudo confeccionar una guía con las principales funcionalidades que presentan cada uno, permitiendo así poder abarcar todos los requisitos que deberá cumplir el nuevo entorno interactivo a desarrollar.
- ✓ Con la elaboración de los nuevos requisitos para el montaje correcto de los cursos en la Web fue posible hacer un estudio detallado de los elementos y herramientas que le darían solución a los mismos, para luego hacer énfasis en las funcionalidades que pudiesen presentar problemas técnicos en el momento ser llevadas a la Web.
- ✓ Al definir la estructura estándar de todos los cursos, así como la organización arquitectónica y los módulos que lo compondrán, es posible que cualquier desarrollador pueda crear nuevos cursos mediante el estudio de dicha estructura y adaptarse a la misma, para una mejor organización de su trabajo y una futura comprensión por demás desarrolladores.
- ✓ La valoración emitida por los expertos permitió demostrar que la investigación realizada puede ser aplicada en el desarrollo materiales de apoyo a los cursos optativos para la facultad 8.

RECOMENDACIONES

Tomando como base la investigación realizada y la experiencia acumulada durante la realización de este trabajo, se proponen las siguientes recomendaciones en aras de lograr buenos resultados en la confección de un entorno de aprendizaje interactivo en la Web:

- ✓ Ampliar las funcionalidades del entorno de acuerdo a las necesidades que requiera el curso que se desea montar.
- ✓ Mantener actualizada la presente investigación de acuerdo a las últimas versiones de las herramientas y librerías que se abordan en él.
- ✓ Elaborar una guía mas detallada del proceso de confección del componente para la administración de los cuestionarios mediante el uso de funcionalidades del *Moodle*.
- ✓ Enriquecer el trabajo con la documentación generada a partir del desarrollo de un curso de segundo perfil basado en la propuesta que se expone en el trabajo.

TRABAJOS CITADOS

1. *Preparando para un nuevo modo de conocer*. **Bartolomé, A.** s.l. : EDUTEC, 1996, Revista Electrónica de Tecnología Educativa, pág. No. 4.
2. **Martínez, F.** *Perspectivas de las nuevas tecnologías en la educación*. Madrid : F.J. Tejedor y A. G. VAlcárcel, 1996
3. **Rodríguez, D F.** Monografías. [En línea] [Citado el: 1 de febrero de 2009.] <http://www.monografias.com/trabajos10/mmedia/mmedia.shtml?monosearch#quees> .
4. Informatical. [En línea] [Citado el: 1 de febrero de 2009.] <http://www.psicopedagogia.com>.
5. **Sánchez, Jaime.** *Construyendo y Aprendiendo con el Computador*. 1999.
6. El *Software* Educativo. **Marquès, Perez.** Barcelona : s.n., 2001.
7. **Fabio Hernando González Reyes.** Mailxmail. *Diseño de Software Educativo*. [En línea] 2 de febrero de 2006. [Citado el: 3 de febrero de 2009.] <http://www.mailxmail.com/curso/informatica/disenosoftware/capitulo9.htm>.
8. **Joan Valentina Sancler.** Blog de Tópicos Especiales I. [En línea] 28 de 4 de 2007. [Citado el: 4 de febrero de 2009.] <http://valentinasancler.blogia.com>.
9. Slideshare. [En línea] 2008. [Citado el: 4 de febrero de 2009.] <http://www.slideshare.net.9>.
10. **Usef Hassan Montero.** NSU. [En línea] 1 de diciembre de 2002. [Citado el: 10 de febrero de 2009.] <http://www.nosolousabilidad.com/articulos/hipermedia.htm>. ISSN 1886-8592.
11. Todo Programas. *Alice 2.0.7*. [En línea] 21 de enero de 2009. [Citado el: 15 de febrero de 2009.] <http://www.todoprogramas.com/macintosh/alice>.

12. Barrapunto. [En línea] 20 de mayo de 2007. [Citado el: 16 de febrero de 2009.]
<http://formacion.barrapunto.com/article.pl?sid=07/05/20/108251>.
13. Nodo Creativo. [En línea] 29 de marzo de 2008. [Citado el: 19 de marzo de 2009.]
http://nodocreativo.blogspot.com/2008_03_23_archive.html.
14. Mykme. [En línea] 01 de 04 de 2007. [Citado el: 25 de marzo de 2009.]
<http://mykme.com/blog/2007/04/01/que-es-un-lms-una-definicion/>.
15. Mosaic. [En línea] [Citado el: 25 de marzo de 2009.] <http://mosaic.uoc.edu/articulos/cms1204.html>.
16. **Steve White, Andy Wallace.** *Manual del Usuario Joomla! 1.0.x.* 2006.
17. **Kennard, James.** *Mastering Joomla! 1.5 Extension and Framework Development.* s.l. : Packt Publishing Ltd., 2007. ISBN 978-1-84719-282-0.
18. Centro de ayuda Joomla! Spanish. [En línea] [Citado el: 13 de abril de 2009.]
<http://ayuda.joomlaspanish.org/content/view/387/82/>.
19. RibosoMatic. [En línea] 17 de noviembre de 2007. [Citado el: 20 de abril de 2009.]
<http://www.ribosomatic.com/articulos/top-librerias-ajax-dhtml-y-javascript/>.
20. El método Delphi. [En línea] [Citado el: 19 de mayo de 2009.]
<http://www.gtlic.ssr.upm.es/encuestas/delphi.htm>.

BIBLIOGRAFÍA

1. **Boton "cerrar ventana" window.close.** [En línea] 2005.[Citado el: 23 de abril de 2009.]
<http://www.efectosjavascript.com/boton-cerrar-ventana.html>.
2. **Creación de Contenidos y Menú.** [En línea] 2005.[Citado el: 23 de abril de 2009.]
<http://www.joomlaos.net/creacion-de-contenidos-y-menu.php>.
3. **Curso Rápido Moodle.** [En línea] [Citado el: 24 de abril de 2009.]
<http://teleformacion.uci.cu/course/view.php?id=248>.
4. **Documentación de Moodle en español.** [En línea] [Citado el: 8 de mayo de 2009.]
<http://docs.moodle.org/es>.
5. **Documentación de Moodle en inglés.** [En línea] [Citado el: 8 de mayo de 2009.]
<http://docs.moodle.org/en>.
6. **El método Delphi.** [En línea] [Citado el: 8 de mayo de 2009.]
<http://www.gtlic.ssr.upm.es/encuestas/delphi.htm>.
7. **Flash libre, fácil.** [En línea] [Citado el: 23 de abril de 2009.]
http://zarate.tv/articulos/flash_libre_facil/.
8. **Html-VideoCodes.** [En línea] [Citado el: 23 de abril de 2009.]
<http://www.tizag.com/htmlT/htmlvideocodes.php>.
9. **Ktoon.** [En línea] [Citado el: 23 de abril de 2009.] <http://ktoon.uptodown.com/ubuntu/>.
10. **Luis CT's. Estructura de archivos de DATABASE.** [En línea] [Citado el: 21 de mayo de 2009.]
<http://blogs.dfwikilabs.org/luisct/2008/02/13/estructura-de-archivos-de-database/>.
11. **Mercè Gisbert Cervera, Antoni Bellver Torlà, Robert Rallo Moya, Jordi Adell Segura.** Entornos Virtuales de Enseñanza - Aprendizaje. [En línea] [Citado el: 19 de marzo de 2009.]
<http://www.ucm.es/info/multidoc/multidoc/revista/cuad6-7/evea.htm>.

12. **Mykme.** [En línea] 01 de 04 de 2007. [Citado el: 19 de 03 de 2009.]
<http://mykme.com/blog/2007/04/01/que-es-un-lms-una-definicion/>.
13. **Mosaic.** [En línea] [Citado el: 19 de 03 de 2009.] <http://mosaic.uoc.edu/articulos/cms1204.html>.
14. **Nodo creativo.** *Manual de Adobe Flex: Introducción*. [En línea] 29 de marzo de 2008. [Citado el: 8 de mayo de 2009.] <http://nodocreativo.blogspot.com/2008/03/adobe-flex-cierre->.
15. **Noticias Moodle.** *Las ventajas de la Web 2.0 en la educación de hoy en día.* [En línea] 3/12/2008.[Citado el: 19 de marzo de 2009.]
<http://www.noticiasmoodle.com/2008/12/las-ventajas-de-la- Web -20-en-la-educacion-de-hoy-en-dia/>.
16. **Round the bend wizards.** *Using SWFObject.* [En línea] [Citado el: 6 de mayo de 2009.]
<http://www.rtbwizards.com/helpcenter/swfobjectscript.htm>.
17. **Sanchez, Marko.** ¿Se puede producir multimedia con *software* libre? [En línea] 19/7/2007.[Citado el: 18 de marzo de 2009.]
<http://markosanchez.com/%C2%BFse-puede-producir-multimedia-con-software-libre/23/>.
18. **Sigüenza, Juan Alberto.** *Diseño de materiales docentes multimedia en entornos virtuales de enseñanza-aprendizaje.* [En línea].
<http://www.ucm.es/info/multidoc/multidoc/revista/num8/siguenza.html>.
19. **Soundmanager Sitio Web Oficial.** [En línea] [Citado el: 23 de abril de 2009.]
<http://www.schillmania.com/projects/soundmanager/>.
20. **Soundmanager2 Basic Examples.** [En línea] [Citado el: 23 de abril de 2009.]
<http://artistechmedia.com/sm2/soundmanagerv290a-20081028/demo/api/>.
21. **Tareas. Experiencia realizada utilizando los Talleres y.** Diseño de Objetos de Aprendizaje con Moodle. *Experiencia realizada utilizando los Talleres y Tareas.* [En línea] [Citado el: 17 de marzo de 2009.] <http://www.sociedadtecnologia.org/sonia/files/124/377/Salamanca+2008.pdf>.

22. The art of Joomla. *How Joomla! 1.5 Files are Organised.* [En línea] [Citado el: 5 de mayo de 2009.] <http://www.theartofjoomla.com/magazine/article/16-how-joomla>.

23. Tutorial en español de Joomla. [En línea] 2006. [Citado el: 23 de abril de 2009.] <http://www.astrolabio.us/Tutorial%20de%20manejo%20del%20administrador%20de%20contenido.pdf>.

24. unijimpe. *Como usar SWFObject.* [En línea] [Citado el: 6 de mayo de 2009.] <http://blog.unijimpe.net/insertar-flash-en-tu-wordpress/>.

ANEXOS

Anexo 1: Cuestionario a Miembros del Panel de Experto.

Nombre del Encuestado: _____

Ocupación: _____

1. ¿Considera usted que la alternativa de la construcción del entorno de aprendizaje interactivo que sirva de apoyo a los cursos de segundo perfil en la universidad mediante el empleo de la Web sea ventajosa?

Si ___ No ___ No Sé ___

¿Por Qué?

2. ¿Considera usted que las funcionalidades desarrolladas para la creación del entorno de aprendizaje interactivo cubren todas las necesidades?

Si ___ No ___ No Sé ___

¿Por Qué?

3. ¿Cree usted que la solución propuesta contiene los elementos y herramientas necesarias para su desarrollo?

Si ___ No ___ No se ___

¿Por qué?

Anexo 2: Modelo para informar el peso de los criterios.

Nombre y Apellidos del evaluador _____

1. Usted debe otorgarle a cada criterio un peso en dependencia del peso total dado al grupo al que pertenece:

- ✓ Grupo No.1.....25
- ✓ Grupo No.2.....25
- ✓ Grupo no.3.....25
- ✓ Grupo No.4.....25

Para que el peso total asignado sea igual a 100.

Grupo No. 1: Criterios de mérito científico.

1. Valor científico de la investigación.
Peso___
2. Calidad de la investigación.
Peso___
3. Novedad científica de la investigación.
Peso___

Grupo No. 2: Criterios de implantación.

1. Necesidad del empleo de la propuesta.
Peso___
2. Posibilidad de aplicación de la propuesta.
Peso___
3. Integración de la propuesta al proceso de enseñanza-aprendizaje en la UCI.
Peso___

Grupo No.3: Criterios de flexibilidad.

1. Adaptabilidad al proceso de enseñanza-aprendizaje de la UCI.
Peso___
2. Empleo de las herramientas necesarias para la elaboración de las funcionalidades.
Peso___
3. Velocidad de desarrollo de la propuesta.
Peso___

Grupo No.4: Criterios de impacto.

1. Repercusión en el proceso de enseñanza-aprendizaje de la UCI.
Peso___
2. Organización en el proceso de documentación de la investigación.
Peso___
3. Impacto en el área para la cual está destinada la investigación.
Peso___

Anexo 3: Modelo para calificar los criterios.

Nombre y Apellidos del evaluador _____

1. Evaluación de los criterios en una escala del 1 al 5:

Grupo No. 1: Criterios de mérito científico.

1. Valor científico de la investigación.

Peso____

2. Calidad de la investigación.

Peso____

3. Novedad científica de la investigación.

Peso____

Grupo No. 2: Criterios de implantación.

1. Necesidad del empleo de la propuesta.

Peso____

2. Posibilidad de aplicación de la propuesta.

Peso____

3. Integración de la propuesta al proceso de enseñanza-aprendizaje en la UCI.

Peso____

Grupo No.3: Criterios de flexibilidad

1. Adaptabilidad al proceso de enseñanza-aprendizaje de la UCI.

Peso____

2. Empleo de las herramientas necesarias para la elaboración de las funcionalidades.

Peso____

3. Velocidad de desarrollo de la propuesta.

Peso____

Grupo No.4: Criterios de impacto.

1. Repercusión en el proceso de enseñanza-aprendizaje de la UCI.

Peso____

2. Organización en el proceso de documentación de la investigación.

Peso____

3. Impacto en el área para la cual está destinada la investigación.

Peso____

2. Categoría final de la estrategia:

Excelente (Alta novedad científica, con aplicabilidad y resultados relevantes) ____

Bueno (Novedad científica, resultados destacados) ____

Aceptable (Suficientemente bueno con reservas) ____

Cuestionable (No tiene relevancia científica y los resultados son malos) ____

Malo (No aplicable) ____

3. Valoración final de la estrategia

Sugerencias del evaluador para mejorarla:

Elementos críticos que deben mejorarse:

Anexo 4: Plantilla-Peso Promedio/Criterio

G	Expertos/Criterios	E1	E2	E3	E4	E5	E6	E7	Exp. P
	C1								
	C...								
	C...								
	C...								
	C...								
	C...								
	C...								
	C...								
	C...								
	C...								
	Cn								

Anexo 5: Plantilla-Consistencia en trabajo de Expertos.

Expertos/Criterios	E ₁	E ₂	E ₃	E ₄	E ₅	E ₆	E ₇	ΣE	Exp. P	ΔC	ΔC ²
C1											
C...											
C...											
C...											
C...											
C...											
C...											
Cn											
DC											
MΣE											
W											
X ² _{real}											

Anexo 6: Plantilla – Cálculo de $P * c$

Criterios	Calificación (c)					P	P * c
	1	2	3	4	5		
C1							
C...							
C...							
C...							
C...							
C...							
Cn							

GLOSARIO DE TÉRMINOS

Animación: Es una técnica para dar sensación de movimiento a imágenes o dibujos.

Aplicación: Tipo de programa informático diseñado como herramienta para permitir a un usuario realizar uno o diversos tipos de trabajo.

Aprendizaje constructivista: Donde el aprendizaje colaborativo juega un papel importante, activo y manipulable, que involucra a los estudiantes de manera que sean ellos mismos quienes interactúan y exploran los contenidos.

Arquitectura: Es un nivel de diseño que hace foco en algoritmos y estructuras de datos de la computación; el diseño y especificación de la estructura global del sistema es un nuevo tipo de problema. Indica la estructura, funcionamiento e interacción entre las partes de un *software*.

Audiovisual: Proyección de imágenes de una película o diapositivas combinada con sonidos, con fines didácticos.

Back-end: Término que se relaciona con el final de un proceso.

Banners: Formato publicitario en Internet, que se crean a partir de imágenes o de animaciones y son diseñados con la intención de atraer la atención, resaltar noticias o comunicar un mensaje deseado.

Código fuente: Conjunto de líneas de texto que son las instrucciones que debe seguir la computadora para ejecutar en un programa informático. Está escrito por un programador en algún lenguaje de programación.

Conexión: Enlace, juntura o relación entre distintos elementos.

Cuestionario: Documento formado por un conjunto de preguntas que deben estar redactadas de forma coherente y organizadas, secuenciadas y estructuradas de acuerdo con una determinada planificación, con el fin de que sus respuestas nos puedan ofrecer toda la información que se precisa.

Destreza: Habilidad y experiencia en la realización de una actividad determinada, generalmente automática o inconsciente.

Editor: Programa que sirve para crear, modificar, visualizar e imprimir normalmente texto o un conjunto de datos.

Estrategia: Técnica y conjunto de actividades destinadas a conseguir un objetivo.

Evolución: Desarrollo o transformación gradual.

Framework: En el desarrollo de *software*, es una estructura de soporte definida, mediante la cual otro proyecto de *software* puede ser organizado y desarrollado. Incluye soporte de programas, bibliotecas y un lenguaje interpretado entre otros *software* para ayudar a desarrollar y unir los diferentes componentes de un proyecto.

Front-end: De forma general, es un término que hace referencia al estado inicial de un proceso.

Herramienta: Es un objeto elaborado a fin de facilitar la realización de una tarea que requiere de una aplicación correcta.

Interacción: Se refiere a una acción recíproca entre dos o más objetos.

Interfaz: Es el medio con que el usuario puede comunicarse con una máquina, un equipo o una computadora y comprende todos los puntos de contacto entre el usuario y el equipo, normalmente suelen ser fáciles de entender y fáciles de accionar.

Librerías: Conjunto de subprogramas utilizados para desarrollar *software*. Contienen código y datos, que proporcionan servicios a programas independientes.

Llamativo: Que atrae la atención por su apariencia o aspecto externo.

Metodología: Métodos de investigación que se siguen para alcanzar una gama de objetivos en una ciencia.

Multimedia: Uso del ordenador para presentar texto, gráfica, video, animación y sonido (los 5 en una forma integrada).

Pantalla: grupo de elementos de medias visuales que están comprendidos en una vista determinada.

Plataforma: Es precisamente el principio, en el cual se constituye un hardware, sobre el cual un *software* puede ejecutarse/desarrollarse.

Portabilidad: Característica que posee un *software* para ejecutarse en diferentes plataformas, el código fuente del *software* es capaz de reutilizarse en vez de crearse un nuevo código cuando el *software* pasa de una plataforma a otra.

Producto: Cualquier bien material, servicio o idea que posee un valor para el consumidor o usuario y sea susceptible de satisfacer una necesidad.

Requerimiento: Una característica, propiedad o comportamiento que se desea para el sistema.

Satisfacer: Cumplir unas condiciones o exigencias.

Scripts: Es un guión o conjunto de instrucciones que permiten la automatización de tareas creando pequeñas utilidades.

Streaming: Término que se refiere a ver u oír un archivo directamente en una página Web sin necesidad de descargarlo antes al ordenador.

Técnica: Procedimiento o conjunto de estos, (reglas, normas o protocolos), que tienen como objetivo obtener un resultado determinado, ya sea en el campo de la ciencia, de la tecnología, del arte o en cualquier otra actividad.

Tecnología: Conjunto de saberes que permiten fabricar objetos y modificar el medio ambiente, incluyendo las plantas y animales, para satisfacer las necesidades y deseos humanos.