



**Universidad de Cienfuegos**

**Facultad de Ingeniería**

**Trabajo de Diploma**

**Título:**

*“Módulo para la generación automática de exámenes.”*

**Autor:** Sayli Adira Chaviano Martínez

**Tutor:** Dr. Ernesto Roberto Fuentes Gari.

**Cienfuegos, Cuba.**

**Junio, 2015**

## *DECLARACIÓN DE AUTORÍA*

Yo, Sayli Adira Chaviano Martínez declaro que soy la única autora del trabajo de diploma titulado: “Módulo para la generación automática de exámenes” y autorizo a la Facultad de Ingeniería en la Universidad de Cienfuegos, para que hagan el uso que estimen pertinente del presente trabajo de diploma.

Para que así conste firmo la presente a los 10 días del mes de junio del 2015.

---

Sayli Adira Chaviano Martínez

Autor

---

Ernesto Roberto Fuentes Gari

Tutor

## *Resumen*

En este proyecto de fin de carrera se da una solución al problema de la generación automática de exámenes, integrando componentes propios del proceso docente-educativo cubano como son los niveles de asimilación. Para ello se ha realizado un módulo para la plataforma de enseñanza virtual Moodle, encargado de facilitar esta labor a los profesores.

Al hilo de esta aplicación, se ha recopilado y sistematizado el material de ayuda para desarrolladores disponible en la red. Asimismo, se ha redactado una guía que ayudará a los principiantes en la tarea de modificar módulos existentes o añadir nuevas funcionalidades a Moodle.

El presente trabajo lleva como título: "Módulo para la generación automática de exámenes".

En el informe se exponen los conceptos asociados al problema planteado en la investigación y se hace referencia a las metodologías, herramientas y tecnologías usadas. A través del documento se describen los elementos que conforman el análisis y el diseño, siguiendo lo establecido por la metodología de RUP y utilizando el Lenguaje Unificado de Modelado UML.

Palabras claves: Moodle, niveles de asimilación.

## *Abstract*

In this final year project a solution to the problem of automatic generation of tests is given by integrating own components of the Cuban educational process as are the levels of assimilation. For this we have made a module for Moodle e-learning platform, responsible for facilitating the work of teachers.

In line with this application, it has collected and systematized material help developers available online. It has also prepared a guide to help beginners in the task of modifying existing modules or add new functionality to Moodle.

This paper is entitled: module for automatic generation of tests.

In the report the concepts related to the problem posed in the research and reference methodologies, tools and technologies used is exposed. Through the document the elements of analysis and design, following the provisions of the RUP methodology and using the Unified Modeling Language UML are described.

Keywords: Moodle, levels of assimilation.

## *Índice*

Introducción.....	1
Capítulo 1. Fundamentos Teóricos. ....	6
1.1    Introducción.....	6
1.2    La integración de plataformas de enseñanza virtual en la docencia universitaria cubana, como una necesidad actual. ....	6
1.3    Sistemas de gestión de aprendizaje. ....	9
1.3.1    Sistemas de Administración de Aprendizaje o LMS .....	9
1.3.2    Sistemas de Administración de Contenidos de Aprendizaje o LCMS. 10	
1.4    Moodle .....	11
1.4.1    Ficheros principales en Moodle .....	17
1.4.2    Librerías más importantes .....	18
1.5    Flujo actual de los procesos que originan la situación problemática.....	18
1.6    Sistemas automatizados existentes vinculados al campo de acción ..	20
1.7    Metodologías y tecnologías actuales a considerar.....	22
1.7.1    Metodologías actuales a considerar Rational Unified Process ...	22
1.7.2    Tecnologías actuales a considerar. ....	24
1.8    Concepción arquitectónica.....	25
1.9    Análisis crítico de las fuentes y bibliografías utilizadas .....	27
1.10    Conclusiones.....	28
Capítulo 2. Modelo del negocio. Modelo del Sistema y Captura de Requisitos. 29	
2.1    Introducción.....	29
2.2    Modelo del negocio actual.....	29
2.3    Reglas del negocio a considerar .....	30
2.4    Modelo de casos de uso del negocio .....	30
2.4.1    Actores y trabajadores del negocio.....	31

2.4.2	Diagrama de casos de uso del negocio.....	32
2.4.3	Descripción de los casos de uso del negocio .....	32
2.4.4	Diagrama de actividades de los casos de uso del negocio. ....	34
2.5	Modelo de objetos.....	35
2.6	Captura de requisitos y Modelo del Sistema.....	35
2.6.1	Descripción del sistema propuesto.....	35
2.6.2	Actores del sistema a automatizar.....	36
2.6.3	Casos de uso del sistema.....	37
2.6.4	Diagrama de casos de uso del sistema.....	37
2.6.5	Descripción de casos de uso del sistema.....	37
2.6.6	Definición de los requisitos funcionales .....	38
2.6.7	Definición de los requisitos no funcionales .....	38
2.7	Conclusiones.....	41
Capítulo 3. Construcción de la solución propuesta. Estudio de factibilidad.....		42
3.1	Introducción .....	42
3.2	Diagrama de clases del diseño .....	42
3.3	Diseño de la Base de Datos.....	44
3.3.1	Modelo lógico de datos .....	44
3.3.2	Modelo físico de datos .....	45
3.4	Modelo de despliegue.....	45
3.5	Principios del diseño del sistema.....	46
3.5.1	Estándares en la Interfaz de la aplicación. ....	46
3.5.2	Tratamiento de excepciones.....	47
3.5.3	Concepción del sistema de protección y seguridad.....	48
3.5.4	Estándares de codificación. Restricciones en el diseño y la implementación.....	49
3.6	Estudio de factibilidad.....	50

3.6.1 Estimación por casos de uso.....	50
3.6.2 Obtención de los Puntos de Casos de Uso Ajustados (PCUA) .....	52
3.6.3 Cálculo del Esfuerzo de desarrollo (E).....	54
3.6.4 Cálculo de costos.....	56
3.6.5 Análisis de los costos y beneficios.....	56
3.9 Diseño de pruebas funcionales.....	57
3.9.1 Caso de prueba para el Caso de Uso Gestionar Niveles de asimilación.....	58
3.9.2 Caso de prueba para el Caso de Uso Generar Examen.....	58
3.9.3 Caso de prueba para el Caso de Uso Exportar examen.....	60
3.10 Conclusiones del capítulo.....	61
Conclusiones.....	62
Recomendaciones.....	64
Referencias bibliográficas .....	65
Bibliografía .....	67
Glosario de término .....	71
Anexos .....	72

## *Índice de Ilustraciones*

Ilustración 1. Diagrama de casos de uso del negocio .....	32
Ilustración 2. Diagrama de actividades de caso de uso del negocio "Realizar examen" .....	34
Ilustración 3. Diagrama de clases del Modelo de Objetos.....	35
Ilustración 4. Diagrama de casos de uso del sistema .....	37
Ilustración 5. Diagrama de Clases Web "Generar Examen " .....	43
Ilustración 6. Diagrama de Clases Web "Gestionar Niveles de asimilación por pregunta" .....	43
Ilustración 7. Diagrama de Clases Web "Exportar examen" .....	44
Ilustración 8. Modelo de despliegue .....	45
Ilustración 9. Vista de asignar nivel de asimilación a pregunta .....	58
Ilustración 10. Vista e generar examen. ....	59
Ilustración 11. Vista de exportar examen a PDF .....	60
Ilustración 12. Prototipo de Interfaz de Asignar nivel de asimilación a pregunta .....	102
Ilustración 13. Prototipo de Interfaz de Generar examen .....	102
Ilustración 14. Prototipo de Interfaz de Exportar examen a PDF .....	103
Ilustración 15. Modelo lógico de datos. ....	103
Ilustración 16. Modelo físico de datos. ....	104

## *Índice de Tablas*

Tabla 1: Ficheros principales en Moodle.....	17
Tabla 2: Librerías principales de Moodle.....	18
Tabla 3: Actores del negocio.....	31
Tabla 4: Trabajadores del negocio.....	31
Tabla 5: Descripción del caso de uso "Realizar examen" .....	32
Tabla 6. Actores del sistema.....	36
Tabla 7. Casos de uso del sistema .....	37
Tabla 8. Descripción de casos de uso del sistema.....	37
Tabla 9. Criterios factor de peso de los actores sin ajustar.....	50
Tabla 10. Clasificación de los Actores del sistema.....	51
Tabla 11. Criterios factor de peso de los casos de uso sin ajustar.....	51
Tabla 12. Clasificación de los casos de uso del sistema.....	51
Tabla 13. Descripción y Peso de los TCF .....	53
Tabla 14. Descripción y Peso de los FA.....	54
Tabla 15. Estimación del tiempo de desarrollo por etapas.....	56

## *Introducción*

En la sociedad del conocimiento del siglo XXI, el uso de herramientas tecnológicas para procesar y transmitir información ha trascendido a todos los niveles de la sociedad, impactando con mayor fuerza en los ámbitos de la educación, la universidad y, por ende, del docente universitario, pues es en este nivel donde la comunicación tanto remota como presencial, se apoya en estos medios. Y es tan así que, un número cada vez mayor de universidades en todo el mundo están implementando nuevos modelos educativos, en los que cobran mayor significación las plataformas de enseñanza e- learning.[1]

Consideradas como Sistemas de Gestión de Aprendizaje (SGA) o Learning Management System (LMS), las plataformas de enseñanza son un software que automatiza la administración de acciones de formación. Estos sistemas registran usuarios, organizan diferentes cursos en un catálogo, almacenan datos sobre usuarios y proveen de informes para la gestión. Ofrecen los mecanismos necesarios para que el formador pueda crear y distribuir contenidos, monitorizar la participación de los alumnos inscritos, así como evaluar su desempeño.[2]

La Universidad de Cienfuegos enfrenta el reto de incorporar plenamente las tecnologías de información y comunicaciones (TICs) en sus procesos formativos.

Es por ello que se asume la adopción de la plataforma de e-learning (aprendizaje en línea) Moodle, como vía para el establecimiento de una nueva forma de educación: la formación basada en la red.

El Departamento de Matemáticas de la facultad de Ingeniería Informática de la Universidad de Cienfuegos viene realizando un esfuerzo para generalizar el uso de las TICs a nivel docente. Estas actuaciones se han centrado en la utilización de Moodle como herramienta para la gestión de la docencia. No obstante, teniendo en cuenta que los estudios son presenciales, no se

considera Moodle como un medio básico de comunicación con los estudiantes, pero sí una buena herramienta para el auto aprendizaje y la autoevaluación.

Aun así, la utilización de Moodle permite organizar y programar el trabajo docente de una forma muy ágil, clara y estructurada.

Durante la puesta en marcha de la plataforma Moodle, se ha echado en falta en la herramienta una serie de funcionalidades que serían de utilidad, una de ellas es el citado módulo para generar automáticamente exámenes. Hasta la fecha, esta tarea no es soportada por otra aplicación que responda al objetivo educativo de la institución. Aunque debemos precisar que en nuestra Universidad, lo más cercano a la generación automática de exámenes se logra a partir del uso de un módulo existente en la plataforma Moodle, en el cual el docente puede confeccionar un examen, creando preguntas o agregándolas desde una base de datos. En tal sentido, la plataforma presenta limitaciones evidentes para el docente cubano, ya que no incluye elementos determinantes en el proceso docente-educativo como son la integración de niveles de asimilación por objetivos y preguntas a evaluar en un examen. Situación que está muy lejana a la realidad del modelo educativo que persigue la universidad cubana, donde lo principal no es sólo que el estudiante se apropie del conocimiento, sino que sea capaz sobre la base de él, razonar, investigar y llegar a conclusiones fundamentadas de carácter científico.

El modelo educativo cubano siempre ha estado en busca de la excelencia y la calidad; de ahí su reconocimiento a nivel mundial. Para ello la excelencia debe estar presente en toda la gestión del proceso, tanto en el diseño, como en la ejecución y en la evaluación del mismo. Razón por la cual se definen elementos dentro de la evaluación como categoría didáctica que determinan la calidad del proceso, ejemplo: los niveles de asimilación.

Cuando se planteó desarrollar este módulo, un rápido vistazo a la documentación para desarrolladores de Moodle puso de manifiesto que la información fundamental se encuentra dispersa.

Lo antes explicado recoge las necesidades a las que responde este proyecto. Por un lado, se estudiará con detenimiento la arquitectura de Moodle, así como la documentación y las librerías de código que provee al desarrollador. Además se profundizará en particularidades del proceso docente educativo cubano. Se realizará una guía o manual para el desarrollador que ayude en la tarea de modificar o desarrollar nuevas extensiones para añadir funcionalidades a la plataforma. Se desarrollará un módulo para generar automáticamente exámenes, que permitirá a los profesores generar exámenes de manera automática, seleccionando los objetivos a evaluar y asignando para cada pregunta el nivel de asimilación. Para ello será necesario categorizar los objetivos. El profesor podrá imprimir y exportar el examen a PDF, según desee.

Actualmente la plataforma virtual Moodle cuenta con una media de 100 visitas diarias, generadas por los más de 1123 usuarios registrados en la plataforma y matriculados en diversos cursos correspondientes a asignaturas que se cursan en la UCF. Estas cifras nos permiten imaginar la importancia que tendrá el módulo desarrollado y los beneficios que reportará a todos los usuarios del Sistema de Gestión de aprendizaje Moodle.

Teniendo en cuenta la **situación problemática** expuesta anteriormente se define como **problema** a resolver de esta investigación: la carencia de un módulo en el Moodle que facilite la generación de exámenes de manera automática, evitando la duplicidad de los mismos y que considere los niveles de asimilación.

Por lo anteriormente expuesto se define como **objeto de estudio**, la evaluación como categoría didáctica dentro del proceso docente educativo. Del cual se deriva como **campo de acción** la generación de exámenes haciendo uso de Moodle.

Se define como **idea a defender**: con el desarrollo de nuevas funcionalidades para Moodle que permitan generar exámenes de manera automática, a partir de la selección de objetivos a evaluar y los niveles de asimilación asociados, se facilitará el trabajo de los docentes con respecto a la evaluación.

Después de analizada la problemática, se define como **objetivo general** de este trabajo, desarrollar un módulo para Moodle que permita la generación automática de exámenes teniendo en cuenta aspectos como los niveles de asimilación.

Como **objetivos específicos** se definen:

- Profundizar en particularidades del proceso docente educativo cubano.
- Revisar la arquitectura de Moodle, así como la documentación y las librerías de código que provee al desarrollador.
- Analizar, diseñar e implementar un módulo para generar automáticamente exámenes.
- Validar el sistema propuesto.

Para cumplir los objetivos fue preciso cumplir las siguientes **tareas**:

- Análisis de la metodología a seguir para confeccionar exámenes siguiendo lo establecido por el modelo educativo cubano.
- Estudio de la metodología a seguir para integrar funcionalidades a la plataforma Moodle.
- Investigación sobre los sistemas existentes vinculados con el objeto de estudio.
- Selección de las metodologías y tecnologías a emplear que den solución al problema.
- Análisis, diseño e implementación de un módulo para la generación automática de exámenes.
- Documentación de la información referente al análisis, diseño e implementación del módulo.

El **Aporte Práctico** de este trabajo lo constituye el módulo implementado para Moodle, el cual extiende las funcionalidades de la plataforma relacionadas con los cuestionarios al permitir la generación automática de exámenes, a partir de preguntas anteriormente elaboradas; clasificadas por objetivos y niveles de

asimilación, simplificando notablemente el proceso de confección de exámenes.

En el desarrollo de este trabajo se ha utilizado el Proceso Unificado de Desarrollo (RUP) y el Lenguaje Unificado de Modelado (UML).

El presente trabajo está estructurado en 3 capítulos de la siguiente forma:

**Capítulo 1:** Fundamentación Teórica. Descripción de los principales conceptos que son necesarios para comprender el objeto de estudio. Análisis del objeto de estudio y los sistemas existentes vinculados a él. Así como, de las metodologías y tecnologías actuales a considerar para el desarrollo del módulo propuesto.

**Capítulo 2:** Modelo del Negocio. Captura de Requisitos y Modelo del Sistema. Descripción de los procesos, actores, trabajadores y casos de uso del negocio. Diagrama de casos de uso del negocio y actividades. Diagrama de clases del modelo de objetos del negocio. Descripción general del sistema que se propone. Definición de los requisitos funcionales y no funcionales, actores y casos de uso del sistema. Diagrama de casos de uso del sistema.

**Capítulo 3:** Construcción de la solución propuesta. Descripción del diseño a través del diagrama de clases, el de clases persistentes y el modelo físico de datos. Se definen además el diagrama de implementación y los principios de diseño seguidos en la aplicación. Estudio de factibilidad.

## *Capítulo 1. Fundamentos Teóricos.*

### **1.1 Introducción**

En este capítulo se analiza el flujo de los principales procesos que se desarrollan en el campo de acción tanto como las causas que originan la situación problemática. Además se puntualizan los conceptos necesarios para comprender el dominio del problema y el objeto de estudio.

También se realiza un análisis crítico y comparativo con otros softwares existentes en el país y en el extranjero. Así como las tendencias, metodologías y tecnologías actuales, sobre las cuales se fundamenta la propuesta de automatización.

### **1.2 La integración de plataformas de enseñanza virtual en la docencia universitaria cubana, como una necesidad actual.**

En la esfera educativa con el transcurrir de los años, han surgido muchas herramientas para proporcionar ayuda al profesor en su trabajo. Esto ha sido posible a través del uso e incorporación de las nuevas tecnologías en la educación. Con el pasar del tiempo el hombre ha desarrollado medios didácticos que permiten una mejor vinculación entre los profesores, los estudiantes y la tecnología mejorando considerablemente el proceso de enseñanza aprendizaje. Las aplicaciones a la educación de estas tecnologías, en los últimos años, han abarcado una amplia gama de propuestas en lo referente al uso de plataformas e- learning en la docencia.[3]

Una aproximación al estudio evolutivo de este desarrollo se pudiera establecer en tres etapas.

La primera etapa caracterizada por la creación y gestión de información en línea (texto, imágenes, gráficos y sonidos). Así como, la ausencia de herramientas elaboradas de colaboración (foros, chats, diarios, control de usuarios), ni apoyo en tiempo real.

En una segunda etapa la aplicación de los Sistemas de Gestión de Contenidos (CMS) y luego los Sistemas de Gestión de Aprendizaje (LMS), los cuales se

caracterizan por disponer de herramientas que permiten la distribución de cursos, recursos, noticias, y contenidos.

En una tercera etapa los Sistemas de Gestión de Contenidos de Aprendizaje LCMS, plataformas que integran las funcionalidades de los LMS y CMS y que incorporan la gestión de contenidos para personalizar los recursos de cada estudiante.[4]

Cuando se habla de tecnología se refiere a toda la infraestructura que apoya el sistema e-learning. Esto incluye tanto el equipamiento informático (hardware y software) como a los equipos humanos encargados de su diseño, implementación y mantenimiento.

La sociedad actual, sustentada por el uso generalizado de las potentes y versátiles tecnologías de la información y las comunicaciones (TIC), ha experimentado significativos cambios que alcanzan todos los ámbitos de la actividad humana. En particular, en la esfera de la educación sus efectos se manifiestan de manera muy especial, pues si bien el desarrollo de las computadoras había permitido avances en su aplicación en la educación, fue hasta que se reunieron los avances de las computadoras con los avances de las telecomunicaciones, cuando las aplicaciones a la educación se multiplicaron y expandieron de manera considerable.[5]

En este sentido han surgido nuevos escenarios educativos, en los cuales las TIC brindan un acceso amplio y generalizado a los servicios de educación superior y formación continua, a la vez que favorecen el autoaprendizaje, la autonomía, la interacción, la retroalimentación y la flexibilidad, transformando la enseñanza- aprendizaje en un proceso mucho más eficiente y eficaz.

Esta nueva concepción de espacios educativos, constituye una modalidad de la educación a distancia que se ha denominado “entorno virtual de enseñanza aprendizaje”, los cuales surgen como una respuesta a la necesidad de lograr la formación, capacitación y actualización permanentes en calidad y cantidad del capital humano, sin desligarlo de su propio ambiente laboral y familiar.[6]

Todo este fenómeno trae consigo la necesidad de que la universidad cubana adopte nuevas estrategias que permitan aprovechar al máximo no sólo lo que ofrecen las TIC, sino también que el aprendizaje pueda venir de la mano de experiencias de educación a distancia. En este sentido, se ha elaborado un plan estratégico para la incorporación de las mismas en los distintos ámbitos de la docencia, la investigación y la gestión, de manera que se pueda alcanzar objetivos mucho más ambiciosos. La percepción es situar el uso de las TIC como una gran oportunidad y el uso de la virtualidad como un elemento diferencial de valor añadido, factor clave para la innovación y la mejora de la calidad educativa en sus instituciones.

Situación que conlleva a una serie de cambios y transformaciones que influyen en todas las esferas y abren espacios de búsquedas científicas dentro de los elementos que tienen que ver con las nuevas realidades en la educación, la cual se caracteriza, entre otras cosas, por la introducción de plataformas educativas y de teleformación.

Es por ello, que hoy podemos decir con toda responsabilidad que las universidades cubanas, en su proceso educativo han integrado dichas plataformas y que sin lugar a dudas, la experiencia ha sido de beneplácito tanto para docentes como para estudiantes. No obstante, no podemos implementar esta experiencia en toda su magnitud, debido a las condiciones de limitaciones reales que tiene el país, hablamos de disposición de Internet y condiciones óptimas de hardware.

En este contexto, podemos afirmar que la introducción de TICs en la educación superior genera oportunidades de desarrollo en los siguientes ámbitos:

- ✓ Modelo pedagógico: Se pasa de un paradigma centrado en la enseñanza a un paradigma centrado en el aprendizaje, con énfasis en los estilos y ritmos del proceso cognitivo y en la construcción colaborativa de los conocimientos.
- ✓ Oferta educativa que traspasa fronteras: Se posibilita la entrega del servicio educativo fuera del territorio nacional de las instituciones.

- ✓ Gestión institucional: Se agiliza la toma de decisiones y su comunicación y/o socialización con la comunidad educativa.
- ✓ Estrategias de investigación: Se favorecen las relaciones en tiempo real entre investigadores y/o grupos de investigadores, sin restricciones de distancia.[7]

### 1.3 Sistemas de gestión de aprendizaje.

Tanto los LMS (Learning Management System) como los LCMS (Learning Content Management System), que veremos en los apartados siguientes, se pueden generalizar como sistemas de gestión de aprendizaje, ya que el primero gestiona la parte administrativa de los cursos así como el seguimiento de actividades y avance del alumno. El segundo gestiona el desarrollo de contenidos, su acceso y almacenamiento. En el entorno educativo, los más comunes son los LMS ya que la complejidad de los LCMS los ha llevado a un desarrollo más lento. Debemos tener en cuenta que una determinada herramienta (LMS o LCMS) no está directamente ligada a un paradigma educativo concreto, sino que cualquier sistema de gestión de aprendizaje puede adaptarse a las necesidades educativas concretas de cada institución.[8]

#### 1.3.1 Sistemas de Administración de Aprendizaje o LMS

Entre las herramientas más utilizadas para los ambientes o sistemas e-learning están los Sistemas de Administración de Aprendizaje o LMS, también ampliamente conocidos como plataformas de aprendizaje. Un LMS es un software basado en un servidor Web que provee módulos para los procesos administrativos y de seguimiento que se requieren para un sistema de enseñanza-aprendizaje, simplificando el control de estas tareas. Se distinguen como grupos funcionales de los sistemas de aprendizaje: gestión de cursos, gestión de clases, herramientas de comunicación, herramientas para los estudiantes, gestión del contenido, herramientas de evaluación y gestión de la institución educativa.

En los LMS los módulos con funciones administrativas permiten, por ejemplo, configurar cursos, matricular alumnos, registrar profesores, asignar cursos a un alumno, llevar informes de progreso y calificaciones. Los LMS también facilitan el aprendizaje distribuido y colaborativo a partir de actividades y contenidos preelaborados, de forma síncrona o asíncrona, utilizando los servicios de comunicación de Internet como el correo, los foros, las videoconferencias y el chat.

El alumno interactúa con la plataforma a través de una interfaz web que le permite seguir las lecciones del curso, realizar las actividades programadas, comunicarse con el profesor y con otros alumnos, así como dar seguimiento a su propio progreso con datos estadísticos y calificaciones.

Algunos LMS se han desarrollado con base a ciertos estándares y están preparados para comunicarse unos con otros y para compartir contenidos. Como contenidos se consideran aquellos de formato digital, como son los diferentes tipos de documentos electrónicos, textos, imágenes, multimedia, animaciones, vídeos, simulaciones, etcétera.[4]

### **1.3.2 Sistemas de Administración de Contenidos de Aprendizaje o LCMS.**

Los Sistemas de Administración de Contenidos de Aprendizaje o LCMS (Learning Content Management Systems) tienen su origen en los CMS (Content Management System) cuyo objetivo es simplificar la creación y la administración de los contenidos en línea, y han sido utilizados principalmente en publicaciones periódicas (artículos, informes, fotografías, etcétera). En la mayoría de los casos lo que hacen los CMS es separar los contenidos de su presentación o estilo en pantalla y también facilitar un mecanismo de trabajo para la gestión de una publicación web. Los LCMS siguen el concepto básico de los CMS, que es la administración de contenidos, pero enfocados al ámbito educativo, administrando y concentrando únicamente recursos educativos y no todo tipo de información.

En esencia, se define LCMS como un sistema basado en Web que es utilizado para crear, aprobar, publicar, administrar y almacenar recursos educativos (como los objetos de aprendizaje) y cursos en línea.[4] Los principales usuarios son los diseñadores de cursos que utilizan los contenidos para construir dichos cursos, los profesores que utilizan los contenidos para complementar su material de clase e incluso los alumnos en algún momento pueden acceder a la herramienta para desarrollar sus tareas o completar sus conocimientos.

Los contenidos usualmente se almacenan como objetos descritos e identificables de forma única. En un LCMS se tienen contenedores o repositorios para almacenar los recursos, que pueden ser utilizados de manera independiente o directamente asociados a la creación de cursos dentro del mismo sistema. Es decir, el repositorio puede estar disponible para que los profesores armen los cursos, pero también pueden estar abiertos para que cualquier usuario recupere recursos no vinculados a ningún curso en particular, que les pueden ser de utilidad para reforzar lo aprendido sobre algún tema. El proceso de trabajo dentro de un LCMS requiere de control en cada fase del contenido, esto conlleva un proceso editorial para controlar la calidad de los contenidos creados y para permitir y organizar su publicación.

El mercado de los LCMS es mucho más reducido que el de los LMS, entre los productos más comunes se encuentra ATutor, distribuido como software libre y compatible con estándares e-learning y b-learning para la exportación e importación de contenidos con otras plataformas y con Repositorios de Objetos de Aprendizaje.[9]

#### **1.4 Moodle**

Dentro de los Sistemas de Gestión de Aprendizaje de código abierto cabe destacar a Moodle (Module Object-Oriented Dynamic Learning Environment), que cuenta en la actualidad con 44,921 sitios y más de 30 millones de usuarios en 208 países del mundo y se ha puesto, en los últimos años, a la cabeza del mercado de aprendizaje a distancia. [2]

Moodle es un paquete de software libre para la creación de cursos y sitios Web basados en Internet. Consiste en un proyecto diseñado con la filosofía de la educación social constructivista.

De acuerdo con Stallman (2004), con software libre nos referimos a la libertad de los usuarios para ejecutar, copiar, distribuir, estudiar, cambiar y mejorar el software. Nos referimos especialmente a cuatro clases de libertad para los usuarios de software: [10]

- ✓ Libertad 0: la libertad para ejecutar el programa sea cual sea nuestro propósito.
- ✓ Libertad 1: la libertad para estudiar el funcionamiento del programa y adaptarlo a tus necesidades –el acceso al código fuente es condición indispensable para esto.
- ✓ Libertad 2: la libertad para redistribuir copias y ayudar así a tu vecino.
- ✓ Libertad 3: la libertad para mejorar el programa y luego publicarlo para el bien de toda la comunidad –el acceso al código fuente es condición indispensable para esto.

Software libre es cualquier programa cuyos usuarios gocen de estas libertades.

El software libre aplicado a contextos educativos, ofrece posibilidades que pueden favorecer el proceso de enseñanza-aprendizaje en función de los destinatarios, de sus necesidades, su nivel de formación ya que puede ser modificado y adaptado en función de nuestros intereses y de los objetivos que persigamos.[11]

Moodle se encuadra dentro de la categoría de los LMS, y, puede ser utilizado tanto para desarrollar cursos completamente on-line siguiendo la metodología del e- learning, como para complementar la educación presencial haciendo uso tan solo de las herramientas que el profesor considere oportunas para facilitar sus clases.

La palabra Moodle es el acrónimo de Modular Object-Oriented Dynamic Learning Environment (Ambiente de Aprendizaje Modular Orientado a Objetos Dinámicos). Se distribuye gratuitamente bajo licencia GPL de GNU.

Entre las características más importantes de la plataforma Moodle a nivel técnico destacan las siguientes:

1. Su diseño es modular, permitiendo gran flexibilidad para agregar y suprimir funcionalidades en muchos niveles.
2. Se ejecuta sin necesidad de cambios en el sistema operativo bajo Unix, Linux, Windows, Mac OS X, Netware y todos aquellos sistemas operativos que permitan PHP.
3. Soporta las principales marcas de manejadores de bases de datos.
4. Su actualización desde una versión anterior a la siguiente es un proceso muy sencillo. Dispone de un sistema interno capaz de reparar y actualizar sus bases de datos cada cierto tiempo.[11]

A nivel pedagógico Moodle ofrece funcionalidades bastante atractivas para los docentes, como las siguientes:

1. Promueve una pedagogía constructivista social. Dado el carácter colaborativo de las herramientas utilizadas en él y la filosofía de trabajo en la que se sustenta.
2. Es adecuado para la enseñanza únicamente a través de la Red como para complementar la enseñanza presencial.
3. Cuenta con una interfaz atractiva, de tecnología sencilla, ligera eficiente y compatible.
4. Permite el acceso de invitados a los cursos.
5. Los cursos son clasificados en categorías, lo que facilita su búsqueda.

6. Es fácil de instalar, por lo que no precisa un nivel avanzado de conocimientos informáticos para proceder a su implementación.

7. Un sitio Moodle puede albergar miles de cursos y estos pueden ser clasificados en distintas categorías.

Moodle se rige por una filosofía basada en la teoría constructivista, sustentada en la idea de interacción con los demás y a través de la conexión de conocimientos nuevos con aprendizaje previamente adquirido. Conectando la estrategia didáctica del profesorado con las ideas previas del alumnado, construyendo de manera sólida los conceptos, interconectando los unos con los otros en forma de red de conocimiento.[6]

Posee una interfaz intuitiva que facilita a los diseñadores de cursos la construcción de foros, chats, lista de correo, glosarios, bibliotecas, autoevaluaciones y demás funcionalidades. Los recursos disponibles en Moodle se pueden clasificar en tres categorías: Recursos Transmisoros, Recursos Interactivos y Recursos Colaborativos. A estas tres categorías también podemos añadir las herramientas de comunicación.

- ✓ Recursos Transmisoros: Se refiere a todos los módulos, recursos, actividades en Moodle que tienen como función principal la de transmitir información. El profesor se establece prácticamente en emisor único. Los alumnos, en receptores. Lo más usual es que estos contenidos estén constituidos por algún tipo de texto más o menos estructurado con apoyo de imágenes, esquemas. Vienen a ser lo que en la enseñanza tradicional llamamos libros de texto, apuntes, manuales didácticos, presentaciones multimedia, etc. En Moodle este tipo de contenidos se generan fundamentalmente a partir de lo que el propio programa llama recursos. El elemento recurso no es más que un enlace a cualesquiera de los recursos materiales que puedan representarse por un archivo electrónico. Documentos de texto, presentaciones con diapositivas, archivos de imagen, programas de ordenador ejecutables, archivos de CAD, archivos de audio y vídeo, etc. pueden constituir un recurso. Los

recursos transmitivos disponibles en Moodle son: Página de texto, Página Web, Enlace a archivo o Web, Directorio, Etiqueta y Libro.

- ✓ Recursos interactivos: Estos recursos se centran más en el alumno, quien tiene determinado control de navegación sobre los contenidos. Cuanto menos lineales sean los contenidos y la propia navegación, mayor interactividad habrá. Se pone el peso, por lo tanto, en definir el sistema por el cual el que aprende accede a la información que se le quiere transmitir. En Moodle los Recursos interactivos disponibles son: Lecciones, Cuestionarios, SCORM, Glosarios y Tareas.
- ✓ Recursos colaborativos: Estas herramientas pueden introducir en nuestras escuelas la posibilidad de disponer de recursos altamente orientados a la interacción y el intercambio de ideas y materiales tanto entre el profesor y los alumnos como de los alumnos entre sí [12]. Moodle incluye las siguientes herramientas colaborativas: Foros, Talleres y Wikis.
- ✓ Herramientas de comunicación: Uno de los propósitos principales de la plataforma Moodle consiste en facilitar y enriquecer la interacción entre todos los miembros de la comunidad (estudiantes entre sí, profesores entre sí y entre alumnos y profesores). Por ello las herramientas de comunicación son básicas. Para Gómez (2006) no resulta conveniente integrarlas dentro de los recursos transmitivos, colaborativos o interactivos, más bien considera las herramientas de comunicación un requisito para que las otras tres funcionen adecuadamente. Las herramientas de comunicación disponibles en Moodle son: Correo electrónico, Chats, Mensajes, Consultas y Encuestas.

Utilizando Moodle como Entorno Virtual de Aprendizaje obtendremos un sistema flexible donde, además de aprender los alumnos pueden compartir experiencias de aprendizaje y conocimientos con otras comunidades virtuales, compuestas por otros usuarios de la plataforma en todo el mundo.

Una de las características principales de Moodle es su modularidad. Cada uno de los recursos que componen la plataforma se define como una pieza de software independiente que puede ser modificada o eliminada según las necesidades de la institución de enseñanza que la adopte. De la misma forma también es posible instalar nuevos módulos que añadan funcionalidad al sistema, bien descargándolos desde la página oficial de módulos no estándar de Moodle o bien desarrollándolos a medida. El proceso de desarrollo de un módulo no es complejo, pero requiere conocer un poco más a fondo la arquitectura y el funcionamiento interno del sistema. Remito entonces al Anexo número 1 (**A1**) en el que se describen los principales elementos de la Arquitectura del Moodle. Teniendo en cuenta lo dispersa que está la información en la red y lo difícil que es acceder a ella, pues esta constancia resulta de vital importancia a futuros desarrolladores en la tarea de agregar nuevas funcionalidades al Moodle.

Quizás lo más complicado en el desarrollo de módulos de actividad para Moodle sean los primeros pasos, debido a la escasa documentación existente, especialmente en español. El Anexo número 2 (**A2**) constituye un manual mediante el cual se pretende simplificar al desarrollador la tarea de programar un nuevo módulo que añada funcionalidades a la plataforma, o de modificar uno ya existente para adaptarlo a unas necesidades concretas. En él se ha recopilado, organizado y sistematizado la información disponible en Internet, especialmente en la página para desarrolladores de la Web oficial de Moodle.

### 1.4.1 Ficheros principales en Moodle

**Tabla 1: Ficheros principales en Moodle.**

Nombre de los ficheros principales	Descripción
Access.php	Define las capacidades de los usuarios en función a los roles para este módulo.
Install.xml	Este fichero XML define las tablas de la base de datos que tendrá nuestro módulo
Upgrade.php	Este fichero mantiene un registro de los cambios que es necesario realizar entre distintas versiones del módulo.
Icong.gif	Este archivo es el icono que representa a un módulo.
Index.php	El fichero lista todas las instancias de una actividad que están presentes en un determinado curso.
Lib.php	El fichero es la biblioteca de funciones del módulo a desarrollar, en ella se implementará toda (o casi toda) la funcionalidad del módulo
Mod_form.php	Mediante el fichero se configurará cada instancia del módulo que se añada a un curso.
Versión.php	Este archivo es importante a la hora de modificar las tablas de la base de datos.
View.php	En él se define el contenido que se mostrará a cada usuario de la actividad mediante el uso de capacidades o permisos.

### 1.4.2 Librerías más importantes

Algunas de estas librerías fueron desarrolladas por el propio equipo de Moodle, otras por terceros y, hoy en día, se distribuyen con Moodle gracias a sus licencias de código abierto.[13]

No existe un único documento que abarque toda la API de Moodle, pero un recurso que puede resultar de gran ayuda es la referencia del código generada a partir de los comentarios del propio código fuente por medio de PHPXref, que se puede encontrar en <http://xref.moodle.org/nav.html?index.html>.

**Tabla 2: Librerías principales de Moodle**

Nombre de la Librería	Descripción
Accesslib.php	Esta librería describe funciones para el soporte de capacidades.
Datalib.php	En esta librería se encuentran las funciones para el manejo de la información de la base de datos
Ddllib.php	En esta librería se pueden encontrar las funciones para el manejo de la estructura de la base de datos.
Dmllib.php	En esta librería se encuentran las funciones necesarias para obtener, borrar, actualizar, contar e insertar registros en la base de datos.

### 1.5 Flujo actual de los procesos que originan la situación problemática.

En la UCF se imparten un grupo significativo de asignaturas de diferentes disciplinas en la modalidad de estudio presencial. Cada materia es impartida por uno o más profesores, siendo ellos los máximos responsables de que el proceso de confección de exámenes se realice con la calidad que amerita. El

jefe de la disciplina o jefe de departamento, son en principio, los que avalarán dicho examen y luego entonces se aplica a los estudiantes.

El proceso de confección de exámenes se inicia cuando el profesor o los profesores de la asignatura consultan el Plan de estudio de la asignatura para fijar los objetivos a evaluar. Revisan la bibliografía disponible, libros de texto y exámenes anteriores, de donde conforman las preguntas y definen el nivel de asimilación que comprobarán para cada una de ellas. Luego, confeccionan el temario con su clave de calificación y son entregados al Jefe de Departamento o Jefe de Disciplina con el fin de aprobar el examen. Una vez aprobado en una fecha previamente fijada los estudiantes realizan los exámenes, al finalizar se recogen y son revisados por los profesores. Finalmente, el profesor comunica la nota al estudiante.

La complejidad de confeccionar un examen, no sólo radica en el hecho de ser completamente manual, sino que además es un proceso que incluye una adecuada preparación metodológica del docente, que puede incluso catalogarse como una actividad extra ya que deben revisar una extensa bibliografía para elaborar preguntas que respondan a un grupo de requerimientos como son: responder a los objetivos y niveles de complejidad a evaluar, que no se repitan entre exámenes, así como evitar la repetición de exámenes de un curso escolar a otro, entre otras cuestiones. Razón que le exige estar elaborando constantemente diversos temarios de exámenes. Independientemente de que los profesores reciben un incentivo económico, el proceso de confección de exámenes no deja de ser de mucha responsabilidad y rigurosidad. Por lo que se convierte en un proceso engorroso y lento.

Como ya se ha comentado en apartados anteriores, la Universidad de Cienfuegos utiliza activamente la plataforma de enseñanza virtual Moodle como complemento on-line a su actividad docente presencial. Su utilización permite organizar y programar el trabajo docente de una forma muy ágil, clara y estructurada haciendo uso de sus recursos estándar, así como de otros módulos no estándar.

En ese sentido, en el Departamento de Matemáticas ha surgido la idea de desarrollar un módulo para Moodle que facilite esta labor a los docentes integrando cualidades del proceso docente- educativo cubano. De manera que se simplifique e integre los elementos a tener en cuenta para desarrollar esta tarea, tales como disponibilidad de las preguntas, clasificación de las mismas por objetivos y niveles de asimilación; así como la entrega de los mismos por parte de la aplicación en un formato imprimible. Logrando estar en sintonía el objetivo educativo de la plataforma con el objetivo del educativo de la institución.

El nivel de asimilación significa el nivel de dominio que deberá tener el estudiante del contenido. Este nivel se puede clasificar en **reproductivo**, **productivo** o **creativo**. El reproductivo es aquel nivel de asimilación que exige que el estudiante sea capaz de repetir el contenido que se le ha informado, ya sea éste en forma declarativa o resolviendo problemas iguales o muy similares a los ya resueltos. El productivo es aquel nivel de asimilación que exige que el estudiante sea capaz de aplicar, en situaciones nuevas para el alumno, los contenidos. La enseñanza problémica, heurística, investigativa, es consecuencia de tener objetivos a un nivel productivo. El nivel más alto de lo productivo es lo creativo. En este nivel creativo el estudiante tiene que hacer aportes cualitativamente novedosos para él, utilizando para ello, la lógica de la investigación científica.[14]

Explicado los niveles de asimilación, se evidencia la importancia que cobran como componente determinante dentro del proceso docente educativo cubano.

## 1.6 Sistemas automatizados existentes vinculados al campo de acción

El tema de la generación automatizada de exámenes ha sido objeto de estudio por muchos desarrolladores de software a nivel mundial, alcanzando resultados de importancia en el tema. Existen varios sistemas encargados de llevar a cabo

esta tarea, que de manera manual resulta lenta y dificultosa. A continuación se describen algunos de estos sistemas:

Tester v2.0: Permite generar exámenes y usarlos para evaluar los conocimientos de los alumnos. El mismo soporta preguntas con múltiple elección, y también permite realizar preguntas en las que se deban completar espacios en blanco mostrando dónde se han cometido errores. Este sistema evalúa y puntúa los exámenes y permite al profesor evaluar las respuestas cuya calificación no es posible realizar automáticamente.[15]

ExamView: Permite crear cualquier tipo de examen o guía de estudio, le provee la flexibilidad de crear exámenes online (Internet y red LAN) o en papel en tan solo minutos. Permite además la actualización del banco de preguntas existente. El sistema resulta ideal para las necesidades de capacitación empresarial y cursos de capacitación. Además permite incorporar elementos de multimedia en sus preguntas de manera sencilla tales como gráficos, cuadros, fotos, ecuaciones matemáticas, sonidos y extractos de películas digitales.[16]

Exámenes: Este sistema permita la generación automática de exámenes partiendo de preguntas almacenadas en una base de datos clasificadas por temas. Los exámenes generados pueden ser almacenados en la base de datos del sistema para su posterior consulta. El software permite que la confección de los exámenes se efectúe de manera automática y manual. [17]

AnyTest 2.50: Este creador y organizador de exámenes resulta muy útil y práctico, ya que permite la aplicación de las evaluaciones por la red o de manera individual en el ordenador. El mismo dispone de opciones para la creación de base de datos personalizadas para el almacenamiento de los datos de las evaluaciones.[18]

TestIt 3.0 Build 110: Esta aplicación permite generar exámenes de diverso tipo y guardar la información en una base de datos muy sencilla de administrar. Se pueden imprimir las pruebas y crearlas en formato HTML.[19]

Genex 1.0: El software permite que la confección de exámenes se efectúe de manera automática. Almacenando preguntas y exámenes en una base de datos.[20]

Los softwares mencionados tienen una aplicación práctica y muestran características funcionales similares a la propuesta de este trabajo, por ejemplo el almacenamiento de preguntas en bases de datos para la posterior utilización de las mismas y brindan opciones para la aplicación de los exámenes tanto Online como en papel. De una manera u otra, no se ajustan totalmente a las necesidades actuales de la Universidad de Cienfuegos, además es necesario considerar que para la utilización de todas las funciones que brindan estos sistemas se debe efectuar el pago de la licencia de los mismos pues las versiones gratis no presentan la funcionalidad completa.

## 1.7 Metodologías y tecnologías actuales a considerar

En el presente epígrafe se realiza un estudio de las principales tendencias, metodologías y tecnologías actuales a tener en cuenta para el desarrollo de la propuesta.

### 1.7.1 Metodologías actuales a considerar Rational Unified Process.

#### **RUP (Rational Unified Process)**

El Proceso Unificado Racional o RUP (Rational Unified Process), es un proceso de desarrollo de software y junto con el Lenguaje Unificado de Modelado UML, constituye una metodología estándar muy utilizada para el análisis, implementación y documentación de sistemas orientados a objetos. Tiene las siguientes características:

- Forma disciplinada de asignar tareas y responsabilidades (quién hace qué, cuándo y cómo).
- Pretende implementar las mejores prácticas en ingeniería de Software.
- Desarrollo iterativo.

- Administración de requisitos.
- Uso de arquitectura basada en componentes.
- Control de cambios.
- Modelado visual del software.
- Verificación de la calidad del software.

RUP es un producto de Rational (IBM). Se caracteriza por ser iterativo e incremental, estar centrado en la arquitectura y guiado por los casos de uso. Incluye artefactos (que son los productos tangibles del proceso como por ejemplo, el modelo de casos de uso, el código fuente, etc.) y roles (papel que desempeña una persona en un determinado momento, una persona puede desempeñar distintos roles a lo largo del proceso).[21]

Divide el proceso de desarrollo en ciclos, teniendo un producto final al final de cada ciclo, cada ciclo se divide en fases que finalizan con un hito donde se debe tomar una decisión importante:

- Inicio: se hace un plan de fases, se identifican los principales casos de uso y se identifican los riesgos
- Elaboración: se hace un plan de proyecto, se completan los casos de uso y se eliminan los riesgos
- Construcción: se concentra en la elaboración de un producto totalmente operativo y eficiente y el manual de usuario
- Transición: se implementa el producto en el cliente y se entrena a los usuarios. Como consecuencia de esto suelen surgir nuevos requerimientos a ser analizados.[21]

### **Lenguaje Unificado de Modelado (UML)**

Lenguaje Unificado de Modelado (UML, por sus siglas en inglés, Unified Modelling Language) es el lenguaje de modelado de sistemas de software más

conocido en la actualidad; está apoyado en gran manera por el OMG (Object Management Group). Es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema de software. El UML ofrece un estándar para escribir un "plano" del sistema, incluyendo aspectos conceptuales tales como procesos de negocios y funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y componentes de software reutilizables. El UML cuenta con varios tipos de modelos, los cuales muestran diferentes aspectos de las entidades representadas.[21]

### 1.7.2 Tecnologías actuales a considerar.

En este apartado se introducirán algunos de los conceptos más importantes relacionados con la arquitectura de Moodle; cómo está estructurado y cómo funciona. Para ello, en primer lugar repasaremos las características del entorno técnico en el que se asienta Moodle que será el que marque la del módulo, deteniéndonos especialmente en los requisitos de base de datos, lenguaje de programación y servidor Web.

#### **Tecnologías de programación del lado del servidor: PHP (Hypertext Preprocessor).**

Moodle está escrito en PHP al igual que nuestro módulo. Como PHP es un lenguaje interpretado, el código de Moodle se almacena como ficheros de código fuente en el servidor web. Cuando un fichero en particular es solicitado por el servidor, el intérprete PHP parsea el código al instante y la salida resultante se envía a través del servidor web. La ,M' de Moodle significa ,Modular', y su estructura de directorios refleja esta modularidad, ya que cada carpeta del nivel superior representa un componente de Moodle. Algunos de los principales componentes soportan módulos de expansión o *plugins*. Cada módulo de expansión tiene su propia carpeta dentro del directorio del componente. En algunos casos, los módulos de actividad también pueden tener soporte para expansiones adicionales. Desde el punto de vista del usuario final, los módulos se instalan copiando su

carpeta dentro del directorio apropiado en el servidor. Moodle detecta el nuevo módulo la siguiente vez que un administrador accede al sistema, localiza el código SQL del módulo, lo ejecuta y finalmente muestra por pantalla los resultados.[22]

### **Servidor Web utilizado: Apache**

Moodle normalmente trabajará con cualquier servidor web que soporte la versión apropiada de PHP. En la práctica, el servidor web más usado es Apache y ha sido también nuestra elección, teniendo en cuenta que es un servidor HTTP de código abierto para varias plataformas que presenta mensajes de error altamente configurables, Base de Datos de Autenticación y negociado de contenidos. Además está disponible para la mayoría de sistemas operativos. Aunque Internet Information Services (IIS) es otro servidor web muy popular para alojar Moodle.

### **Gestor de Base de Datos utilizado: MySQL**

Moodle proporciona soporte para varias bases de datos, entre las que se incluyen: MySQL, PostgreSQL, Microsoft SQL y Oracle. En la práctica la mayoría de los sistemas están desplegados utilizando MySQL, por lo que se presta más atención a las instalaciones de MySQL. Razón por la cual ha sido escogida para el módulo.

MySQL es un sistema de gestión de base de datos relacional lo que permite velocidad y flexibilidad, ya que los datos se archivan en tablas separadas, en lugar de ser colocados en un único archivo, y son conectados por relaciones definidas que hacen posible combinar datos de diferentes tablas sobre pedido.[13]

### **1.8 Concepción arquitectónica.**

Podemos considerar a Moodle como una arquitectura de tres capas. La ventaja principal de esta implementación física es que el desarrollo se puede llevar a cabo en varios niveles y, en caso de que sobrevenga algún cambio, sólo se ataca al nivel requerido sin tener que revisar entre código mezclado. Además,

permite distribuir el trabajo de creación de una aplicación por niveles; de este modo, cada grupo de trabajo está totalmente abstraído del resto de niveles, de forma que basta con conocer la API que existe entre niveles. La API de Moodle no es más que un conjunto de clases y librerías de funciones construidas sobre funciones definidas en las librerías de PHP4. Detalles que enmarcan también la arquitectura específica de nuestro sistema.

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

- Presentación

Como su nombre indica, se limita a la navegabilidad y a gestionar todos aquellos aspectos relacionados con la lógica de presentación de la aplicación, como comprobación de datos de entrada, formatos de salida, internacionalización de la aplicación, etc. Para ello se utiliza fundamentalmente la librería `weplib.php` y `textlib.php` donde se encuentran las funciones para la creación de elementos HTML y salida web en general.

- Negocio o dominio

El resultado del análisis funcional de la aplicación viene a ser la identificación del conjunto de reglas de negocio que abstraen el problema real a tratar. Estas son las que realmente suponen el motor del sistema, dado que se basan en el funcionamiento del modelo real. Esta capa se comunica con la capa de presentación, para recibir las solicitudes y presentar los resultados, y con la capa de datos, para solicitar al gestor de base de datos el almacenamiento o la recuperación de los mismos. A este nivel se usaron librerías como `lib.php`, `datalib.php` y `accesslib.php`.

- Acceso a datos

Esta capa es la encargada de persistir las entidades que se manejan en el negocio, el acceso a los datos almacenados, la actualización, etc., aunque puede ofrecer servicios relacionados con la persistencia o recuperación de información más complejos. A este nivel se usaron las siguientes librerías `dmllib.php` y `ddl.php`; en las cuales se encuentran las funciones necesarias para obtener, borrar, actualizar, contar e insertar registros en la base de datos.

Todas las librerías mencionadas integran funciones y ficheros que conforman la API interna de Moodle. Algunos de los ficheros usados en la implementación del módulo se describen en el **A2** en la sección dedicada a las librerías.

Las capas pueden residir en un mismo equipo. Lo más usual es que haya una multitud de computadoras donde resida la capa de presentación, es decir los clientes de la arquitectura cliente/servidor, y una computadora central que ejerce la función de servidor en dicha arquitectura. Las capas de negocio y de datos pueden residir en un mismo servidor y si el crecimiento de las necesidades lo aconseja se pueden separar en dos o más servidores. Así mismo, si el tamaño o la complejidad de la base de datos aumentaran, se puede separar en varios servidores de datos, los cuales recibirán las peticiones del servidor donde reside la capa de negocio. [23]

### 1.9 Análisis crítico de las fuentes y bibliografías utilizadas

La bibliografía utilizada para realización de este documento fue de gran importancia e impacto, debido a que brinda exhaustiva documentación sobre las tecnologías que aborda e información del funcionamiento interno del Moodle. Su creación es principalmente de los últimos 5 años y tomada en buena parte de los propios autores o personalidades conocidas de años de experiencia en el uso de la tecnología, por lo que se puede afirmar que este trabajo cuenta con bibliografía actualizada y de calidad.

### 1.10 Conclusiones

La confección de exámenes dirigidos a las materias que se imparten en la UCF, tanto como la revisión de la bibliografía y los materiales existentes para la realización de esta labor, resulta actualmente un proceso engorroso, además de traer consigo una carga de trabajo extra para los profesores responsables de esta labor. Hoy en día la Universidad “Carlos Rafael Rodríguez” no cuenta con un sistema automatizado que sea capaz de simplificar todo este proceso, a partir de uso de la plataforma de gestión de docencia Moodle.

Después de ser estudiado los procesos que inciden antes y durante la elaboración de los exámenes, la necesidad de crear un sistema que permita un manejo más fácil sobre la información vinculada a esta labor se hace evidente.

Para el desarrollo del sistema propuesto se utilizó RUP como metodología con UML. Las tecnologías seleccionadas están determinadas por el marco tecnológico de Moodle, una instalación de PHP en funcionamiento que esté soportado a partir de Moodle utilizado .Servidor Web Apache y una base de datos MySQL que esté completamente soportada y recomendada para su uso.

## *Capítulo 2. Modelo del negocio. Modelo del Sistema y Captura de Requisitos.*

### **2.1 Introducción**

Es preciso antes de comenzar a desarrollar una propuesta de solución, comprender el negocio y los principales procesos que tienen lugar en él. Para ello, haremos uso del modelado del negocio, basado en dos tipos de modelos UML: modelo de casos de uso y modelo de objetos.

En el presente capítulo se realiza un estudio de los principales procesos del negocio, se identifican los actores y trabajadores que intervienen en ellos, los casos de uso y objetos. Se describen las reglas que caracterizan el negocio y la solución propuesta utilizando algunos de los artefactos que propone la Metodología RUP como son: diagrama de caso de uso del negocio y diagrama de clases de objeto.

Se realiza una descripción general del sistema que se propone y se definen los requisitos funcionales y no funcionales, actores y casos de uso del sistema. Así como el diagrama de casos de uso del sistema.

### **2.2 Modelo del negocio actual**

El modelo del negocio es una técnica para comprender los procesos de negocio de la organización. Como objetivo identificar los casos de uso y las entidades de negocio relevantes que el software debe soportar, de forma que podríamos modelar sólo lo necesario para comprender el contexto. El modelado del negocio está soportado por dos tipos de modelos de UML: modelos de casos de uso y modelos de objetos.[24]

El proceso del negocio estudiado en el presente trabajo es la confección de exámenes para las asignaturas que se imparten en la Universidad de Cienfuegos “Carlos Rafael Rodríguez”, el cual se desarrolla como sigue:

El proceso de confección de exámenes se inicia cuando el profesor o los profesores de la asignatura consultan el Plan de estudio de la asignatura para

fijar los objetivos a evaluar. Revisan la bibliografía disponible, libros de texto y exámenes anteriores, de donde conforman las preguntas y definen el nivel de asimilación que comprobarán para cada una de ellas. Luego, confeccionan el temario con su clave de calificación y son entregados al Jefe de Departamento o Jefe de Disciplina con el fin de aprobar el examen. Una vez aprobado en una fecha previamente fijada los estudiantes realizan los exámenes, al finalizar se recogen y son revisados por los profesores. Finalmente, el profesor comunica la nota al estudiante.

### **2.3 Reglas del negocio a considerar**

Los profesores de las asignaturas son plantillas de la Universidad de Cienfuegos, por tanto tienen responsabilidades con el proceso docente educativo.

Cada profesor que imparte su asignatura es el máximo responsable de confeccionar los exámenes de su materia así como su clave de calificación.

Un examen se confecciona con preguntas que se clasifican por objetivos a evaluar y niveles de asimilación.

Los objetivos a evaluar en los exámenes están previamente estipulados en el plan de la asignatura.

Los profesores entregan los exámenes en formato digital al Jefe de la disciplina o Jefe de departamento.

Los profesores y el Jefe de la disciplina o Jefe de Departamento son los únicos que conocen el contenido de los exámenes hasta el momento de su aplicación.

Luego de aplicados los exámenes el profesor o el colectivo de profesores que imparten la asignatura analizan sus soluciones y claves de calificación.

### **2.4 Modelo de casos de uso del negocio**

Un modelo de casos de uso del negocio describe los procesos de negocio de una empresa en términos de casos de uso del negocio y actores del negocio

que se corresponden con los procesos del negocio y los clientes, respectivamente.[21]

#### 2.4.1 Actores y trabajadores del negocio.

Es considerado actor del negocio a cualquier individuo, grupo, entidad, organización, máquina o sistema de información externos; que interactúa con el negocio para beneficiarse de sus resultados.[21]

**Tabla 3: Actores del negocio**

Actor	Descripción
Estudiante	Estudia en la UCF y realiza los exámenes de las asignaturas que recibe.

Un trabajador es una abstracción de una persona (o grupo de personas), una máquina o un sistema automatizado; que actúa en el negocio realizando una o varias actividades, interactuando con otros trabajadores y manipulando entidades.[21]

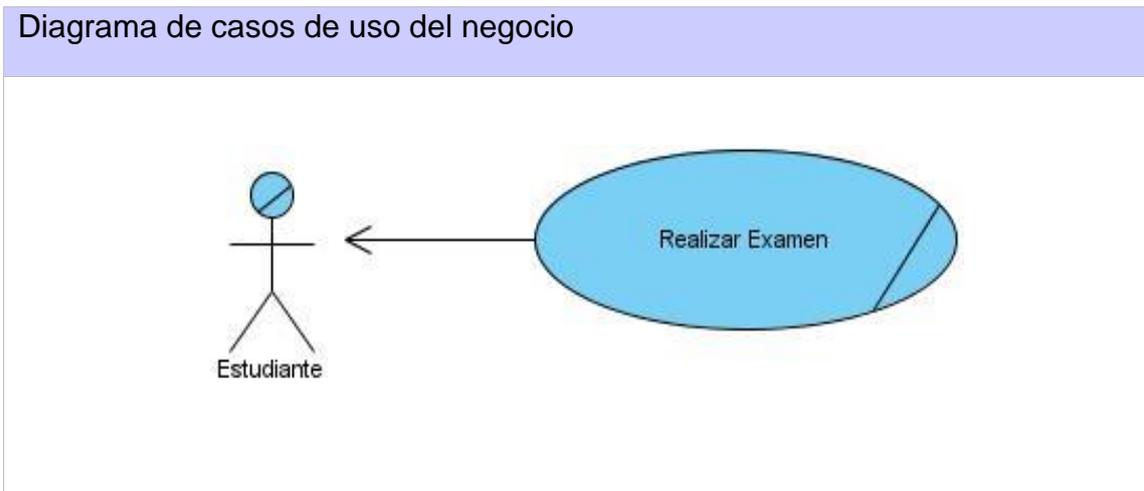
**Tabla 4: Trabajadores del negocio**

Trabajadores	Descripción
Profesor de la UCF	Es el que selecciona en el Plan de Estudio de la asignatura los objetivos que desea evaluar en el examen, luego confecciona preguntas que responden a un objetivo y a un nivel de asimilación. Así como, su clave de calificación.
Jefe de Departamento	Es el encargado de revisar el examen previamente confeccionado por el profesor, da su aprobación o no y luego ya una vez aprobado, lo archiva.

### 2.4.2 Diagrama de casos de uso del negocio

El modelo de casos de uso del negocio se describe mediante diagramas de casos de uso. Presenta un sistema (en este caso, el negocio) desde la perspectiva de su uso, y esquematiza cómo proporciona valor a sus usuarios.

En el siguiente diagrama se muestran los procesos del negocio como casos de uso y sus relaciones con los actores del negocio.



**Ilustración 1. Diagrama de casos de uso del negocio**

### 2.4.3 Descripción de los casos de uso del negocio

**Tabla 5: Descripción del caso de uso "Realizar examen"**

<b>Nombre del caso de uso del negocio:</b>	Realizar examen.
<b>Actores del negocio:</b>	Estudiante
<b>Propósito:</b>	Confeccionar examen de una asignatura que se imparte en la UCF para los estudiantes que la cursan.
<b>Resumen:</b>	
El caso de uso inicia cuando el profesor o los profesores de la asignatura consultan el Plan de estudio de la asignatura para fijar los objetivos a evaluar. Revisan la bibliografía disponible, libros de texto y exámenes anteriores, de	

donde conforman las preguntas y definen el nivel de asimilación que comprobarán para cada una de ellas. Luego, confeccionan el temario con su clave de calificación y son entregados al Jefe de Departamento o Jefe de Disciplina con el fin de aprobar el examen. En caso de ser aprobado, los estudiantes se presentan al examen el día fijado. Una vez aplicado el profesor recoge los exámenes, los califica y le comunica al estudiante la nota. Finalizando el caso de uso. Por otro lado, si no es aprobado el examen, el profesor confecciona otro temario y se reanuda el proceso de aprobación, aplicación y calificación del examen.

<b>Casos de uso asociados:</b>	
<b>Curso normal de los eventos</b>	
<b>Acción del actor</b>	<b>Respuesta del negocio</b>
	<p>El profesor o colectivo de profesores de la asignatura consulta el plan de estudio de la asignatura y fija los objetivos a evaluar.</p> <p>El profesor o colectivo de profesores de la asignatura consulta bibliografía para confeccionar examen.</p> <p>El profesor o colectivo de profesores de la asignatura elabora el examen y su clave de calificación; luego lo entrega en formato digital al Jefe de Departamento o Jefe de disciplina.</p> <p>El Jefe de Departamento o Jefe de Disciplina aprueba el examen.</p> <p>El profesor o colectivo de profesores de la asignatura aplica los exámenes en una fecha previamente fijada.</p>
El estudiante se presenta y realiza el examen.	<p>El profesor o colectivo de profesores de la asignatura recogen los exámenes y los califica en virtud de la clave.</p> <p>El profesor o colectivo de profesores de la asignatura le comunica a los estudiantes la nota alcanzada.</p>
<b>Cursos alternos:</b>	

Si el jefe de departamento o jefe de disciplina determina no aprobar el temario presentado por el profesor de la asignatura, entonces confecciona otro temario y comienza nuevamente el proceso de aprobación, aplicación y calificación del examen.	
<b>Prioridad:</b>	Crítica
<b>Mejoras:</b>	El proceso de confección de exámenes se realizará de manera más ágil y segura, ya que será de forma automatizada.

#### 2.4.4 Diagrama de actividades de los casos de uso del negocio.

Los diagramas de actividades son grafos que describen con más detalles la secuencia de las actividades y los estados transitados por las entidades.

#### Caso de uso “Realizar examen”

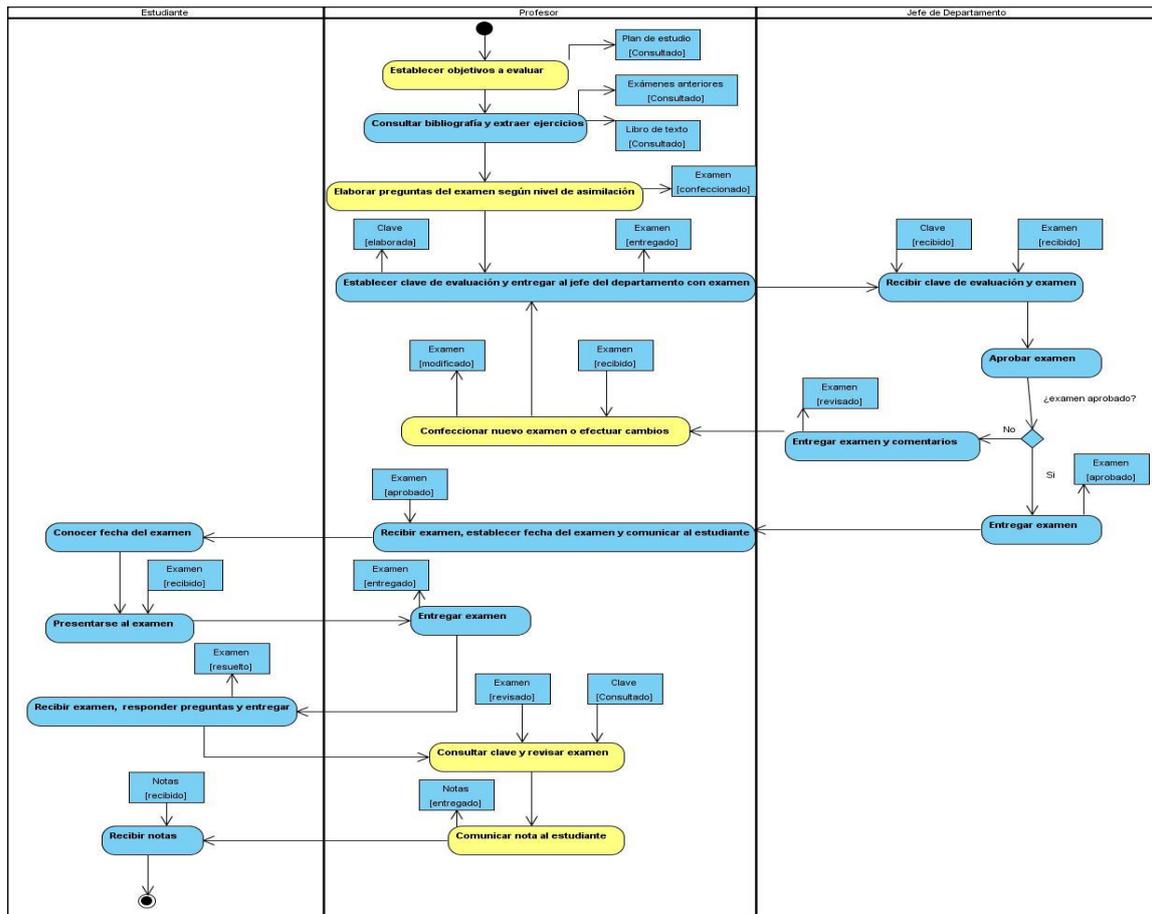


Ilustración 2. Diagrama de actividades de caso de uso del negocio “Realizar examen”

## 2.5 Modelo de objetos

Un modelo de objetos del negocio es un modelo interno a un negocio. Describe cómo cada caso de uso de negocio es llevado a cabo por parte de un conjunto de trabajadores que utilizan un conjunto de entidades del negocio. Una entidad del negocio representa algo, como una factura, que los trabajadores toman, inspeccionan, manipulan, producen o utilizan en un caso e uso del negocio.[21]

Diagrama de clases del Modelo de Objetos "Realizar examen"

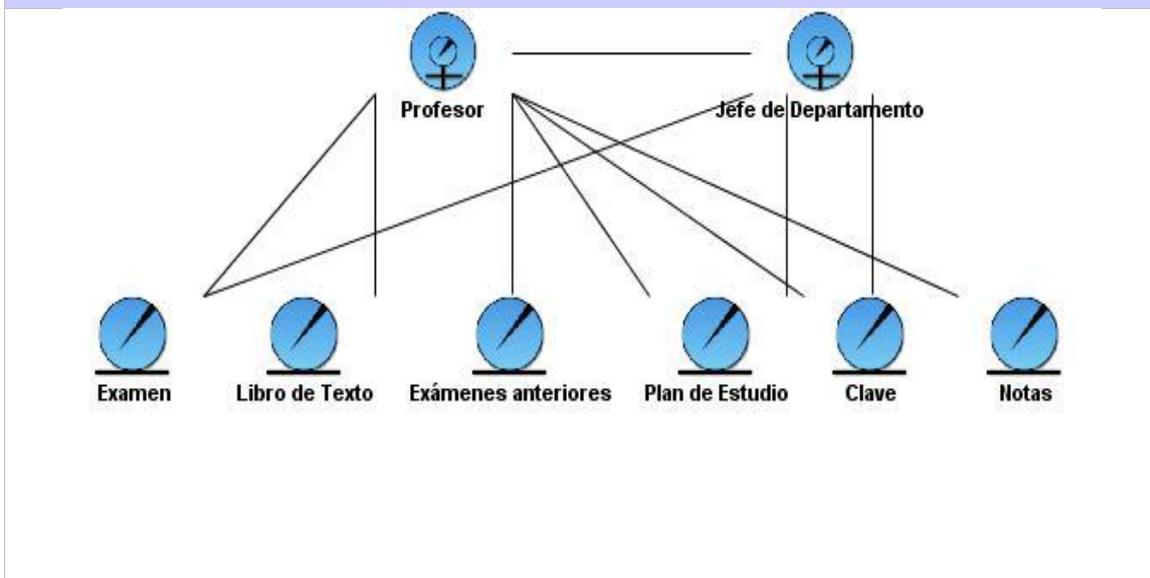


Ilustración 3. Diagrama de clases del Modelo de Objetos

## 2.6 Captura de requisitos y Modelo del Sistema.

### 2.6.1 Descripción del sistema propuesto.

En el presente epígrafe se realiza una descripción textual de la concepción general del funcionamiento del sistema y se definen los requisitos funcionales y no funcionales del mismo.

#### Concepción general del sistema

El resultado que se persigue alcanzar con esta investigación es la obtención de un módulo implementado para Moodle dirigido a la generación automática de exámenes para las asignaturas impartidas en la UCF. El mismo debe presentar al usuario una interfaz compatible con la plataforma que permita la gestión de

toda la información almacenada. Para ello el profesor debe categorizar los objetivos a evaluar usando la opción que Moodle brinda para crear categorías. Luego crea las preguntas mediante la interfaz de Moodle donde le asigna un nivel de asimilación a cada una y la categoría a la que pertenece. Una vez realizada estas acciones, el profesor podrá generar un examen seleccionando las preguntas previamente creadas y almacenadas en la BD, filtrándolas por categorías y niveles de asimilación. Los exámenes generados en el curso no se repiten en un período que selecciona el profesor, a lo que hemos llamado tiempo de caducidad del examen o quiz, término usado en el Moodle para llamar a los cuestionarios. Además, por decisión del profesor podrán aplicarse a los estudiantes de manera online y ser revisados automáticamente, o de lo contrario, podrán imprimirse y aplicarlo a los estudiantes de la forma tradicional.

### 2.6.2 Actores del sistema a automatizar

Un actor no es más que un conjunto de roles que los usuarios de casos de uso desempeñan cuando interactúan con estos casos de uso. Los actores representan terceros fuera del sistema que colaboran con el mismo. Una vez que se han identificado los actores del sistema, tenemos identificado el entorno externo del sistema.

En este caso, se puede distinguir un tipo de usuario que interactúa con las nuevas funcionalidades añadidas al Moodle: el profesor de un curso.

Descripción de Actores del sistema:

**Tabla 6. Actores del sistema**

Actor del sistema	Descripción
Profesor de un curso	<p>El profesor puede realizar todas las funciones del módulo: gestionar nivel de asimilación a pregunta, generar examen y exportar examen.</p> <p>Requisitos funcionales asociados: 1, 2, 3, 4, 5, 6.</p>

### 2.6.3 Casos de uso del sistema

Tabla 7. Casos de uso del sistema

Nombre del caso de uso	Actores del Sistema	Requerimientos al que responde
Gestionar Niveles de Asimilación por Preguntas	Profesor del Curso	R1,R2,R3,R4
Generar Examen	Profesor del Curso	R5,R5.1,R5.2,R5.3
Exportar Examen a PDF	Profesor del Curso	R6

### 2.6.4 Diagrama de casos de uso del sistema

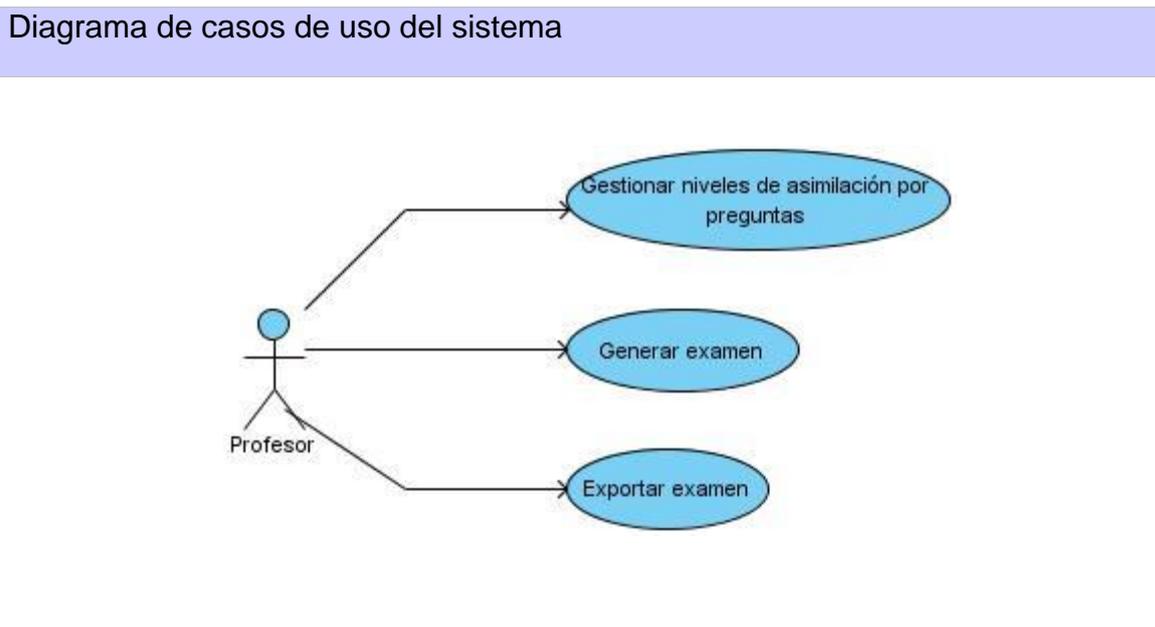


Ilustración 4. Diagrama de casos de uso del sistema

### 2.6.5 Descripción de casos de uso del sistema

Tabla 8. Descripción de casos de uso del sistema

Nombre del caso de uso	Descripción de caso de uso	Prototipo de caso de uso
Gestionar Niveles de Asimilación por Preguntas	Ver <b>A3</b>	Ver <b>A6</b>
Generar Examen	Ver <b>A4</b>	Ver <b>A7</b>

Exportar Examen a PDF	Ver <b>A5</b>	Ver <b>A8</b>
-----------------------	---------------	---------------

### 2.6.6 Definición de los requisitos funcionales

Un requisito funcional especifica una acción que debe ser capaz de realizar el sistema, sin considerar restricciones físicas; requisito que especifica comportamiento de entrada/salida de un sistema.[21]

En el sistema se definen los siguientes requerimientos funcionales:

1. Asignación de nivel de asimilación a pregunta por categoría.
2. Insertar nivel de asimilación por pregunta.
3. Eliminar nivel de asimilación por pregunta.
4. Modificar nivel de asimilación por pregunta.
5. Generar Examen.
  - 5.1. Calcular si el examen es posible generarlo sin que se repita.
  - 5.2. Autoagregar preguntas aleatoriamente.
  - 5.3 Imprimir Examen.
6. Exportar Examen.

### 2.6.7 Definición de los requisitos no funcionales

Un requisito no funcional especifica propiedades del sistema, como restricciones del entorno o de implementación, rendimiento, dependencias de la plataforma; especifica restricciones físicas sobre un requisito funcional.[21]

Para el sistema propuesto se han definido los siguientes requisitos no funcionales:

#### **Requerimientos de apariencia o interfaz externa**

El módulo mostrará una interfaz adaptada al estilo global de Moodle: atractiva, de tecnología sencilla, ligera, eficiente e intuitiva. De manera que facilite a los diseñadores de cursos la construcción de foros, chats, lista de correo, glosarios, bibliotecas, autoevaluaciones y demás funcionalidades.

### **Requerimientos de Usabilidad**

- Facilidad de del manejo de la plataforma Moodle; razón por la cual recomiendo planificar a los docentes cursos de superación de este tema.
- Consistencia en la interfaz de usuario.

### **Requerimientos de Rendimiento**

- Velocidad de procesamiento o cálculo ante las solicitudes del usuario.
- Eficiencia al permitir el acceso a datos de manera simultánea por usuarios.
- Tiempo de respuesta mínimo.

### **Requerimientos de Soporte**

El sistema estará implementado de forma tal que posibilitará su perfeccionamiento y la incorporación de nuevas opciones y funcionalidades en un futuro.

### **Requerimientos de confiabilidad**

Sólo el profesor del curso y el administrador tendrán acceso a modificar la información, garantizándose la seguridad e integridad de los datos almacenados y de esa forma la confiabilidad de la información de los cursos.

### **Requerimientos de Portabilidad**

El módulo se integrará a la plataforma virtual Moodle; la cual generalmente se instala en alguno de los tres sistemas operativos principales: Windows, Mac OS X o Linux (o sistemas operativos basados en Unix).

### **Requerimientos Político-Culturales**

Este sistema no brindará prioridades o limitantes a ningún tipo de persona, independientemente del nivel cultural, social o étnico que tenga.

## **Requerimientos Legales**

La herramienta propuesta responderá a los intereses de la Universidad de Cienfuegos y a la Constitución de la República de Cuba. Este producto no podrá ser comercializado pues, la aplicación fue diseñada con fines específicos de la universidad.

## **Requerimientos de Hardware**

Los requisitos mínimos de hardware deseables, tanto para el equipo como para la impresora, son los siguientes:

Equipo de desarrollo:

- ✓ Procesador Intel Pentium 4, 2 GHz.
- ✓ 1 Gb Memoria RAM
- ✓ 40 Gb disco duro
- ✓ Conexión a internet
- ✓ UPS o fuente de corriente ininterrumpida.

Impresora láser a color:

- ✓ Velocidad de impresión en color (normal, A4): 30 ppm
- ✓ Calidad de impresión en color (óptima): Hasta 1200 x 600 ppp
- ✓ Memoria de serie: 256 Mb

## **Requerimientos de Software**

La aplicación debe poderse ejecutar en entornos Windows y/o Linux (Multiplataforma). La PC donde se encuentre instalada la aplicación debe estar conectada a la red de datos del centro educativo.

Para Moodle versión 2.8, se necesita una instalación de PHP 5.5.12, Servidor Wep Apache 2.4.9 y MySQL 5.4.9 o PostgreSQL 9.3

## Requerimientos de Seguridad

- **Confidencialidad:** La información manejada por el sistema está protegida de acceso no autorizado y divulgación.
- **Integridad:** La información manejada por el sistema será objeto de cuidadosa protección contra la corrupción y estados inconsistentes.
- **Disponibilidad:** Los datos estarán listos para ser usados por cualquier usuario.

## Ayuda y documentación en línea

Teniendo en cuenta que las nuevas funcionalidades implementadas, han sido diseñadas respetando la colección de ideas que ofrece Moodle para adaptar interfaces específicas de nuevos módulos al estilo global, se añadió un botón de Ayuda en las interfaces modificadas de Moodle donde explica al usuario cómo usarlas.

### 2.7 Conclusiones

En este capítulo se describió el proceso de confección de exámenes para las asignaturas impartidas en la Universidad de Cienfuegos. Esta descripción fue realizada mediante el modelo del negocio, para lo cual se elaboraron los modelos de casos de uso y objetos del negocio, identificando: actores, trabajadores, casos de uso y objetos, así como la relación entre ellos. La modelación del negocio propició una mejor comprensión del problema. Además se describió la concepción general del sistema propuesto y se definieron los requisitos funcionales y no funcionales del mismo. También, se identificaron, los tipos de usuarios que tendrá la aplicación y el comportamiento de cada uno de ellos; obteniéndose el modelo de casos de uso del sistema, utilizando la Metodología RUP.

## *Capítulo 3. Construcción de la solución propuesta. Estudio de factibilidad.*

### **3.1 Introducción**

En el presente capítulo se realiza una descripción de la construcción de la solución propuesta. En esta descripción se ha utilizado el diagrama de clases del diseño, el diagrama de clases persistentes y el diagrama del modelo físico de datos, artefactos que propone la Metodología de RUP. Para describir los elementos fundamentales de la implementación se muestra el diagrama de implementación, artefacto que también es propuesto por la metodología utilizada. Además se describen los principios de diseño utilizados y las consideraciones de codificación que se tuvieron en cuenta en la implementación de este sistema. Se aborda el tema relativo al estudio de la factibilidad del producto. Determinar si el desarrollo de un software será factible o no es algo que se debe hacer desde que comienza el proceso de la elaboración del mismo. La estimación de valores de tiempo y esfuerzos asociados a la realización del proyecto constituyen los elementos básicos para llevar a cabo el análisis de la factibilidad, utilizaremos para ello la estimación del esfuerzo basada en el Análisis de Puntos de Casos de Uso.

### **3.2 Diagrama de clases del diseño**

El diagrama de clases presenta las clases del sistema con sus relaciones estructurales y de herencia. En el caso de las aplicaciones Web, el diagrama de clases representa las colaboraciones que ocurren entre las páginas, donde cada página lógica puede ser representada como una clase. Al tratar de utilizar el diagrama de clases tradicional para modelar aplicaciones Web surgen varios problemas, por lo cual los especialistas del Rational plantearon la creación de una extensión al modelo de análisis y diseño que permitiera representar el nivel de abstracción adecuado y la relación con los restantes artefactos de UML. [21]

El Diagrama de clases Web fue definido a partir de los casos de uso del sistema y se muestra en la figura siguiente:

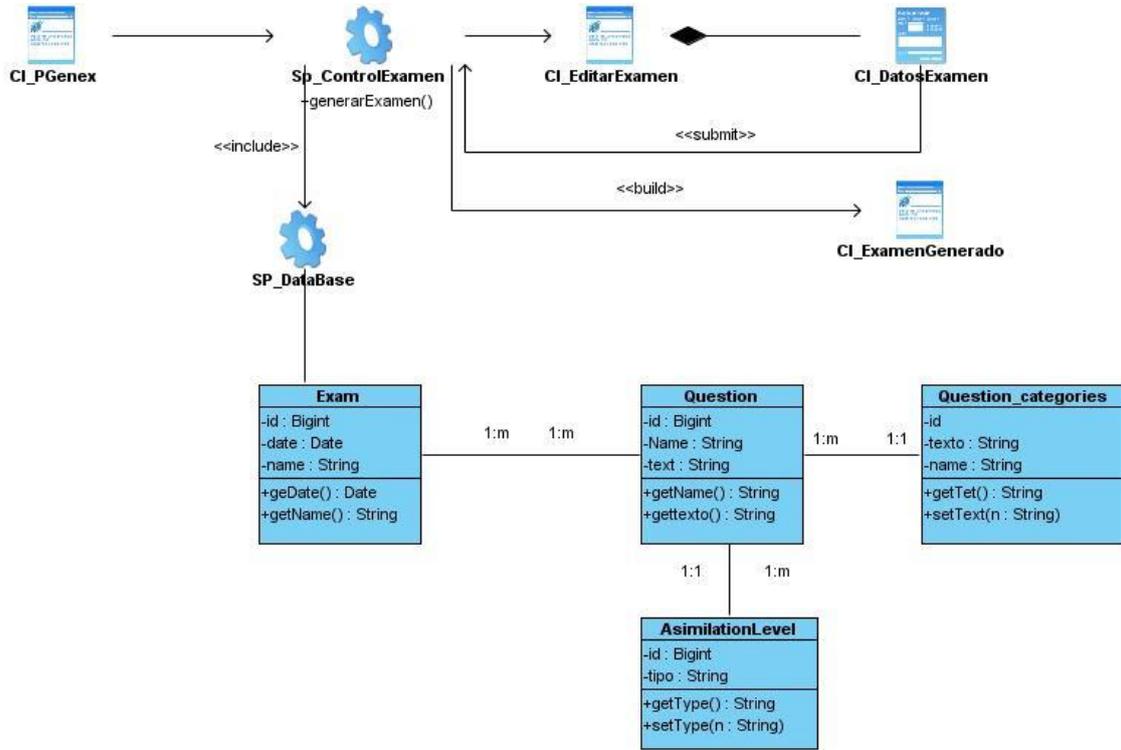


Ilustración 5. Diagrama de Clases Web "Generar Examen"

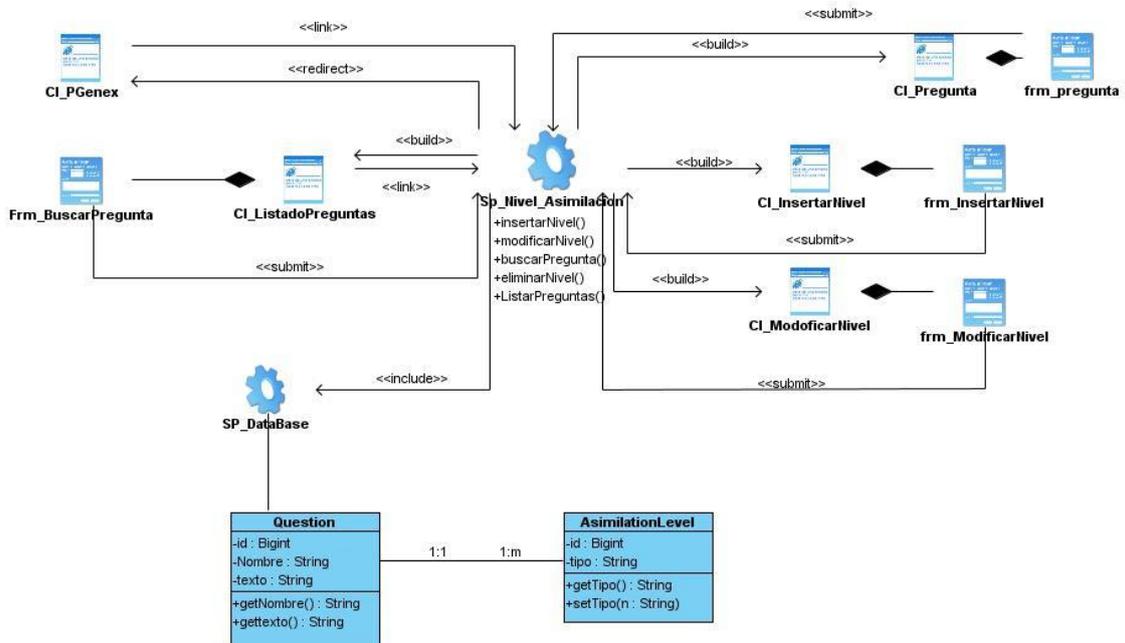


Ilustración 6. Diagrama de Clases Web "Gestionar Niveles de asimilación por pregunta"

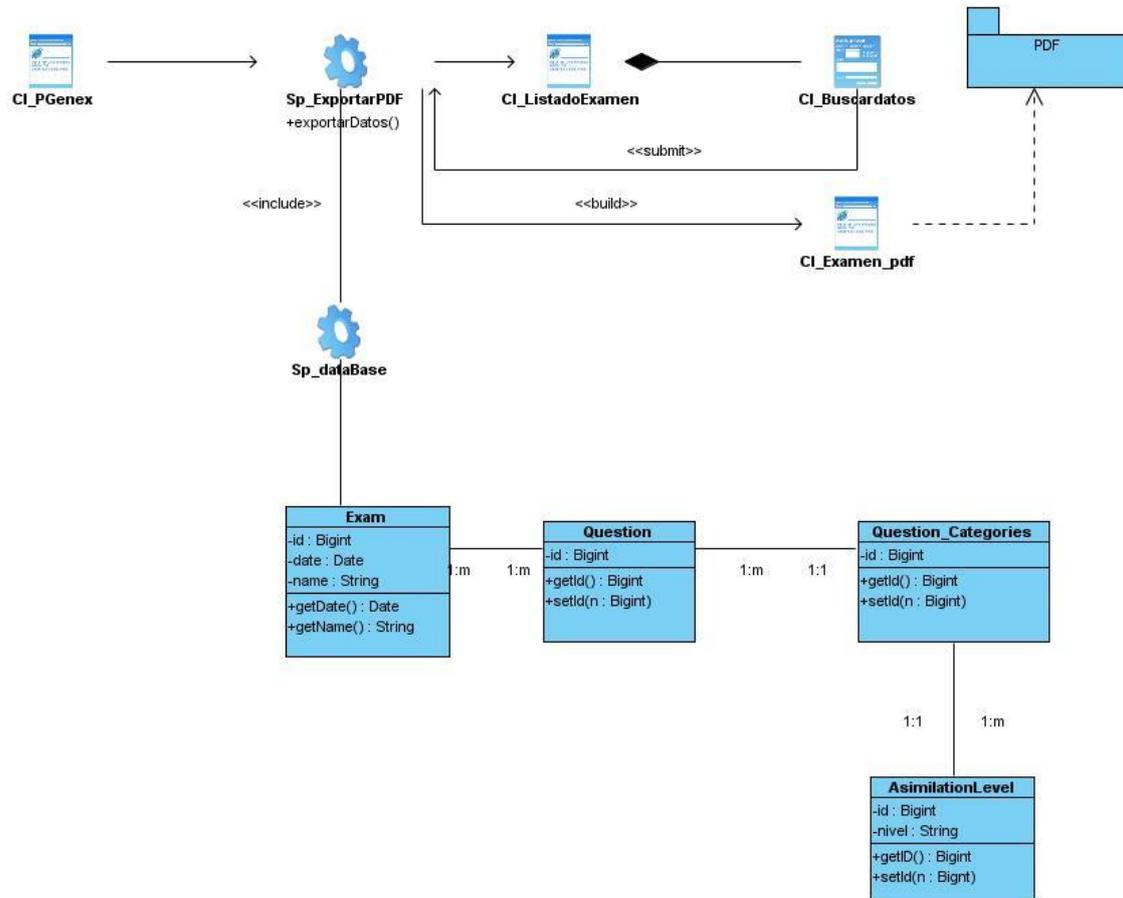


Ilustración 7. Diagrama de Clases Web "Exportar examen"

### 3.3 Diseño de la Base de Datos.

El diseño de la base de datos se divide en dos etapas fundamentales, para de esta forma brindar una mejor comprensión del sistema propuesto.

#### 3.3.1 Modelo lógico de datos

El diagrama del modelo lógico de datos o diagrama de clases persistentes, muestra las clases capaces de mantener su valor en el espacio y en el tiempo.[21] Ver Anexo 9 (A9).

### 3.3.2 Modelo físico de datos

Cuando se define correctamente el modelo lógico, se hace mucho menos engorroso llegar al modelo físico o modelo de datos como también se le denomina en la metodología RUP. “El modelo de datos representa la estructura o descripción física de las tablas de la base de datos y es obtenido a partir del diagrama de clases persistentes”. [21] Ver Anexo 10 (A10).

### 3.4 Modelo de despliegue.

El modelo de despliegue captura la configuración de los elementos de procesamiento, y las conexiones entre estos elementos en el sistema. Consiste en uno o más nodos (elementos de procesamiento con al menos un procesador, memoria, y posiblemente otros dispositivos), dispositivos (nodos estereotipados con una capacidad de procesamiento en el nivel modelado de abstracción), y conectores, entre nodos, y entre nodos y dispositivos. Mapea procesos dentro de estos elementos de procesamiento, permitiendo la distribución del comportamiento a través de los nodos que son representados. [21]

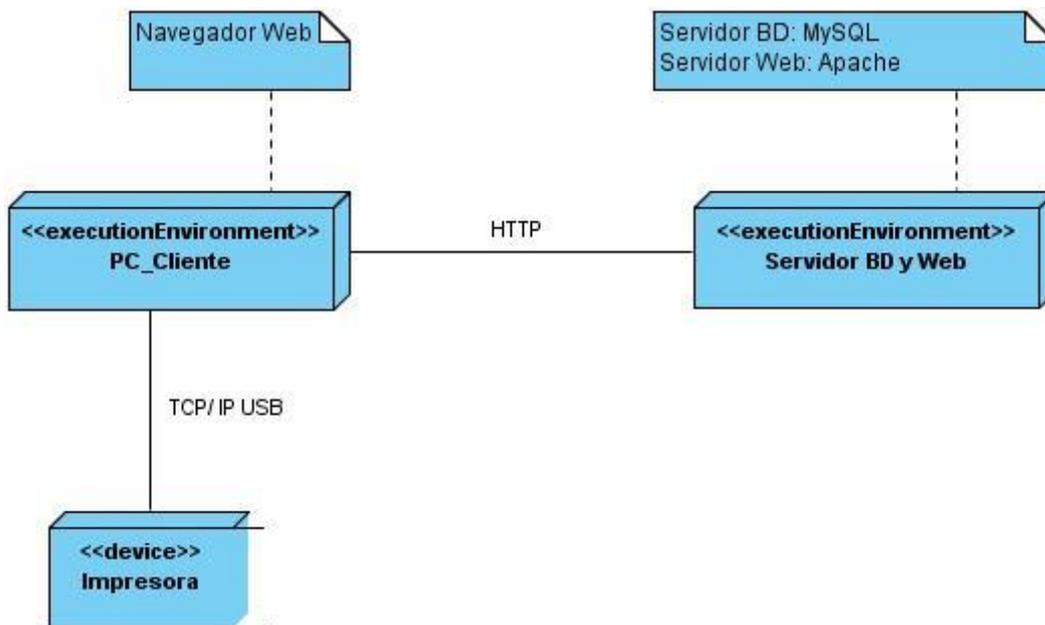


Ilustración 8. Modelo de despliegue

### 3.5 Principios del diseño del sistema.

Para desarrollar un sistema de software con calidad se debe prestar cuidadosa atención al diseño, que tiene como objetivo modelar el sistema y encontrar su forma para que soporte todos los requisitos, tanto funcionales como no funcionales. Además permite al equipo de desarrollo de software evaluar la calidad del software antes de implementarlo.

#### 3.5.1 Estándares en la Interfaz de la aplicación.

La interfaz gráfica es la portada del sistema al cliente y ha de tener gran consistencia, es decir mantener su coherencia de principio a fin. En ese sentido Moodle ofrece una colección de ideas para adaptar la interfaz de nuestro módulo, al estilo global de Moodle de la mejor manera posible. Es por ello se han de mantener las siguientes reglas:

- **Simplicidad:** Debemos usar el mínimo de interfaz necesario para obtener la funcionalidad que necesitamos.
- **Páginas estándar:** Tanto en módulos de actividad como en bloques, existen una serie de ficheros que se deben crear obligatoriamente. Ver **A1**.
- **Debe haber un script por función o página principal.**
- **Plantilla de página:**
  - Imprimiremos las cabeceras con `print_heading()`, usaremos el CSS para IDs y Clases.
  - Imprimiremos las cajas alrededor del texto usando `print_simple_box()`.
- **Plantilla de formulario:**
  - Mostraremos las opciones más importantes en la parte superior.
  - Cada entrada debe tener una etiqueta y, si es necesario, un archivo de ayuda.
  - Si hay más de 10 opciones, las desglosaremos en los parámetros necesarios y opcionales, extra o avanzados.
- **Manejo de tablas:**
  - Usaremos la función `print_table()` cuando sea posible.

➤ **Herramientas de navegación estándar:**

- Todas las páginas deberían llamar a *print\_header()*, y suministrar una ruta de navegación estándar que apareciera allí. Donde sea posible, debería verse como: *COURSE >> INDEX >>INSTANCE >> SUBPAGES...*
- Las páginas incluidas en módulos de actividad deberían llamar a *navmenu()* para generar el menú de navegación apropiado.

➤ **Direcciones URL:**

- Las direcciones URL deben ser lo más cortas posible.
- No debemos usar subrayado en nombres de parámetros o nombres de archivos.
- Nunca debemos usar dos palabras cuando una sea suficiente.

➤ **Botones o enlaces:** El *Web Accelerator* de Google proporciona algunas sugerencias:

- Las acciones que puedan modificar el estado de Moodle (archivos de datos, base de datos, información de sesión) deben ser llevadas a cabo por medio de botones.
- Como mínimo, tales acciones, que son implementadas como enlaces, deben remitir a una página de confirmación que sí utilice botones.

➤ **Enlaces a archivos de ayuda:**

- Los botones de ayuda deben estar a la derecha del objeto (excepcionalmente pueden estar a la izquierda, si el objeto está alineado a la derecha)

### 3.5.2 Tratamiento de excepciones.

Las situaciones que pueden provocar fallos en la ejecución normal de un programa se denominan excepciones. El módulo propuesto presenta una interfaz diseñada, implementada y dirigida a evitar tales situaciones y errores.

En ese sentido se ha prestado especial atención a mantener un nivel de validación que restrinja la introducción de información errónea al mismo y

aclare al usuario el tipo de información que debe manipular; para ello es preciso evitar errores en la edición de la BD. Es recomendable utilizar el editor XMLDB, que podemos encontrar en la sección de administración del sitio, en el submenú *Miscelánea*. Usando este editor se evitarán errores ya que no permite guardar la tabla con errores y solamente nos permite dar valores válidos a las propiedades. Controla además, con el uso de las variables de sesión que brinda el lenguaje PHP, el acceso a páginas restringidas. Todo ello a través, de una serie de mensajes de error con textos sencillos de fácil comprensión para los usuarios.

### 3.5.3 Concepción del sistema de protección y seguridad.

La concepción de protección y seguridad del módulo implementado, está dada por la propia política y reglas que rigen el funcionamiento, la seguridad y la protección de la información en Moodle.

El primer elemento es la autenticación previa de los usuarios que ingresan al sistema, con el objetivo de controlar los niveles de acceso a la información. A las nuevas funcionalidades sólo tendrá acceso el profesor del curso una vez logueado correctamente. El profesor de curso tiene como rol estándar asignado el de **profesor**.

Moodle cuenta con un potente sistema de permisos y control de acceso, cuyo núcleo son las cuentas de usuario. Una cuenta de usuario proporciona acceso individual al sistema mediante un nombre de usuario y una contraseña. Estas cuentas se crean utilizando los plugins de autenticación. Las contraseñas en Moodle se codifican con una función hash MD5 para hacerlas ilegibles y difíciles de averiguar.

Para acceder al sistema, un usuario introduce su usuario y contraseña. Si son correctos se le permite el acceso al sitio. El sistema de sesiones de Moodle utiliza funciones de cookies de PHP para introducir cookies en la sesión actual y ayudar a identificar al usuario durante toda su visita al sistema.

Otro elemento a considerar es garantizar la seguridad de los datos almacenados. Para Moodle cada módulo individual es responsable de realizar

la copia de seguridad de su código, sin tener que modificar el código del núcleo. Para ello se utiliza el fichero *cron.php*, que se encuentra en la carpeta */admin* de Moodle, el mismo se ejecuta como un proceso batch para realizar tareas de mantenimiento del sistema como la realización de las copias de seguridad de los cursos.

Es también requisito importante garantizar la integridad de los datos que se almacenen. La información almacenada deberá ser consistente y se utilizarán validaciones que limiten la entrada de datos incorrectos.

#### **3.5.4 Estándares de codificación. Restricciones en el diseño y la implementación.**

El código fuente de la aplicación deberá respetar los estándares de Moodle.

Se comentan las principales reglas de estilo que se deben cumplir

- El sangrado del texto debe ser siempre de 4 espacios.
- Los nombres de las variables tienen que ser siempre fáciles de leer, procurando que sean palabras en minúsculas con significado en inglés.
- Las constantes tienen que definirse siempre en mayúsculas, y empezar por el nombre del módulo al que pertenecen.
- Los nombres de las funciones tienen que ser palabras sencillas en minúsculas y en inglés. Además de empezar con el nombre del módulo al que pertenecen.
- Los bloques de código siempre deben estar encerrados por llaves.
- Las cadenas tienen que ser definidas utilizando comillas simples siempre que sea posible.
- Los comentarios deben ser añadidos de forma que resulten prácticos, para explicar el flujo del código y el propósito de las funciones y variables.

### 3.6 Estudio de factibilidad.

En este apartado, se muestra todo el análisis realizado en relación al cálculo de la factibilidad del proyecto, utilizando para ello la estimación del esfuerzo basada en el Análisis de Puntos de Casos de Uso. Se decidió utilizar este método ya que es muy útil en proyectos pequeños, con pocos casos de uso del sistema y es recomendable para su uso que el sistema no cuente con más de 50 casos de uso.

La estimación de Puntos de Casos de Uso es un método creado por Gustav Karner que se utiliza para estimar el tiempo de creación de un proyecto mediante la asignación de "pesos" a una cantidad determinada de factores que lo afectan, para luego obtener el tiempo total estimado para la realización del proyecto a partir de esos factores. Estos valores no son absolutos, sino que pueden variar de acuerdo a las características de la organización y del proyecto. Se realiza también una valoración de los resultados del proyecto, teniendo en cuenta una valoración económica y la opinión de los usuarios.

#### 3.6.1 Estimación por casos de uso.

##### Obtención de los Puntos de Casos de Uso sin ajustar (PCU)

###### a. Cálculo del Factor de Peso de los Actores (FPA)

**Tabla 9. Criterios factor de peso de los actores sin ajustar**

Tipo de Actor	Descripción	Factor de Peso
Simple	Otro sistema que interactúa con el sistema a desarrollar mediante una interfaz de programación.	1
Medio	Otro sistema que interactúa con el sistema a desarrollar mediante un protocolo o una interfaz basada en texto	2
Complejo	Una persona que interactúa con el sistema mediante una interfaz gráfica	3

**Tabla 10. Clasificación de los Actores del sistema**

Actores	Tipo de Actor	Factor de Peso
Profesor del curso	Complejo	3
Total		3

Como se describe en la tabla anterior existen en el sistema a desarrollar, 1 actor de tipo complejo: Profesor del curso, pues es una persona que interactúa con el sistema mediante una interfaz gráfica. Multiplicando la cantidad de actores de cada tipo por el peso correspondiente se obtiene que:

$$FPA = 1 * 3$$

$$FPA = 3$$

**b. Cálculo del Factor de Peso de los Casos de Uso (FPCU)**

**Tabla 11. Criterios factor de peso de los casos de uso sin ajustar**

Tipo de Caso de Uso	Descripción	Factor de Peso
Simple	El caso de uso contiene de 1 a 3 transacciones	5
Medio	El caso de uso contiene de 4 a 7 transacciones	10
Complejo	El caso de uso contiene más de 8 transacciones	15

**Tabla 12. Clasificación de los casos de uso del sistema**

Caso de Uso	Clasificación	Factor de Peso
Asignación de Nivel de asimilación a pregunta	Complejo	15
Generar Examen	Medio	10
Exportar Examen	Simple	5

Como puede verse en la tabla de clasificación anterior el sistema está conformado por 3 casos de uso, de ellos 1 complejo, 1 medio y 1 simple.

De ahí que el factor de peso de los Casos de Uso sin ajustar puede calcularse como:

$$FPCU = 1 * 15 + 1 * 10 + 1 * 5$$

$$FPCU = 30$$

Como ya se dispone de los valores de factor de peso de actores y casos de uso sin ajustar es posible obtener el valor de los puntos de caso de uso sin ajustar es:

$$PCU = FPA + FPCU$$

$$PCU = 3 + 30$$

$$PCU = 33$$

### 3.6.2 Obtención de los Puntos de Casos de Uso Ajustados (PCUA)

Una vez que se tienen los Puntos de Casos de Uso sin ajustar, se debe ajustar éste valor mediante la siguiente ecuación:

$$PCUA = PCU * FCT * FA$$

Donde:

**PCUA:** Puntos de Casos de Uso Ajustados

**PCU:** Puntos de Casos de Uso sin ajustar

**FCT:** Factor de Complejidad Técnica

**FA:** Factor de Ambiente

Es necesario calcular los valores de FCT y FA.

#### a. Cálculo del Factor de Complejidad Técnica (FCT)

Este coeficiente se calcula mediante la cuantificación de un conjunto de factores que determinan la complejidad técnica del sistema. Cada uno de los factores se cuantifica con un valor de 0 a 5, donde 0 significa un aporte irrelevante y 5 un aporte muy importante. En la siguiente tabla se muestra el significado y el peso de cada uno de éstos factores:

**Tabla 13. Descripción y Peso de los TCF**

Factor	Descripción	Factor de Peso
T1	Sistema distribuido	2
T2	Objetivos de performance o tiempo de respuesta	1
T3	Eficiencia del usuario final	1
T4	Procesamiento interno complejo	1
T5	El código debe ser reutilizable	1
T6	Facilidad de instalación	3
T7	Facilidad de uso	5
T8	Portabilidad	4
T9	Facilidad de cambio	2
T10	Concurrencia	2
T11	Incluye objetivos especiales de seguridad	1
T12	Provee acceso directo a terceras partes	1
T13	Se requieren facilidades especiales de entrenamiento a usuarios	1

El Factor de complejidad técnica se calcula mediante la siguiente ecuación:

$$FCT = 0.6 + 0.01 * \Sigma (\text{Peso}i * \text{Valor asignado}i)$$

$$FCT = 0.6 + 0.01 * (2 * 3 + 1 * 3 + 1 * 2 + 1 * 3 + 1 * 5 + 0.5 * 5 + 0.5 * 5 + 2 * 5 + 1 * 4 + 1 * 1 + 1 * 5 + 1 * 1 + 1 * 1)$$

$$FCT = 0.6 + 0.01 * 46.00$$

$$\mathbf{FCT = 1.060}$$

**b. Cálculo del Factor de Ambiente (FA)**

**Tabla 14. Descripción y Peso de los FA**

Factor	Descripción	Factor de Peso
E1	Familiaridad con el modelo de proyecto utilizado	1.5
E2	Experiencia en la aplicación	0.5
E3	Experiencia en orientación a objetos	1
E4	Capacidad del analista líder	0.5
E5	Motivación	1
E6	Estabilidad de los requerimientos	2
E7	Personal a tiempo compartido	-1
E8	Dificultad del lenguaje de programación	-1

El Factor de ambiente se calcula mediante la siguiente ecuación:

$$FA = 1.4 - 0.03 * \sum (\text{Peso}_i * \text{Valor asignado}_i)$$

$$FA = 1.4 - 0.03 * (1.5 * 5 + 0.5 * 2 + 1 * 2 + 0.5 * 4 + 1 * 5 + 2 * 5 - 1 * 4 - 1 * 3)$$

$$FA = 1.4 - 0.03 * 20.5$$

**FA = 0.785**

Con el cálculo de estos valores, es posible sustituir en la ecuación inicial y obtener el valor de los puntos de caso de uso ajustado.

$$PCUA = PCU * FCT * FA \quad PCUA = 33 * 1.060 * 0.785$$

**PCUA = 27.4593**

**3.6.3 Cálculo del Esfuerzo de desarrollo (E)**

Originalmente sugirió que cada Punto de Casos de Uso requiere 20 horas-hombre. Posteriormente, surgieron otros refinamientos que proponen una granularidad algo más fina, según el siguiente criterio:

- Se contabilizan cuántos factores de los que afectan al Factor de ambiente están por debajo del valor medio (3), para los factores E1 a E6.

- Se contabilizan cuántos factores de los que afectan al Factor de ambiente están por encima del valor medio (3), para los factores E7 y E8.
- Si el total es 2 o menos, se utiliza el factor de conversión 20 horas-hombre/Punto de Casos de Uso, es decir, un Punto de Caso de Uso toma 20 horas-hombre.
- Si el total es 3 o 4, se utiliza el factor de conversión 28 horas-hombre/Punto de Casos de Uso, es decir, un Punto de Caso de Uso toma 28 horas-hombre.
- Si el total es mayor o igual que 5, se recomienda efectuar cambios en el proyecto, ya que se considera que el riesgo de fracaso del mismo es demasiado alto.

En este proyecto si se analizan los valores tabulados anteriormente, es posible percatarse que el total es 3 o 4, por lo que se utiliza el factor de conversión 28 horas-hombre/Punto de Casos de Uso.

Por lo que el esfuerzo en horas-hombre viene dado por:

$$E = \text{PCUA} \times \text{FC}$$

Donde:

**E:** Esfuerzo estimado en horas-hombre

**PCUA:** Puntos de Casos de Uso Ajustados

**FC:** Factor de Conversión.

De esta forma se obtiene que el esfuerzo necesario para desarrollar los casos de uso del sistema es igual a:

$$E = \text{PCUA} \times \text{FC}$$

$$E = 27.4593 * 28$$

$$E = 768.8604 \text{ Horas-Hombre}$$

Duración:

Trabajando los 25 días al mes y 10 horas al día como promedio, se obtiene:

$$\begin{aligned} \text{Duración (días)} &= \text{Total de horas /hombre entre 10 horas al día} = 768.8604 /10 \\ &= 76.88 \text{ días} \end{aligned}$$

$$\text{Duración (meses)} = \text{Total de días /25 días por mes} = 76.88 /25$$

=3.07 meses ≈ 3 meses y 10 días.

El proyecto se realiza en 3 meses.

Se debe tener en cuenta que éste método proporciona una estimación del esfuerzo en horas-hombre contemplando sólo el desarrollo de la funcionalidad especificada en los casos de uso. Por lo que para obtener una estimación más completa de la duración total del proyecto, hay que agregar a la estimación del esfuerzo obtenida, las estimaciones de esfuerzo de las demás actividades relacionadas con el desarrollo de software. Existe un criterio que estadísticamente se considera aceptable, que distribuye el esfuerzo de las diferentes actividades dentro del desarrollo de un proyecto según la estimación que se muestra en la tabla siguiente, a la que también se le ha agregado el cálculo del valor del esfuerzo para el sistema de esta investigación:

**Tabla 15. Estimación del tiempo de desarrollo por etapas**

Actividad	Porcentaje	Valor (Horas-Hombre)
Análisis	10 %	76.88604
Diseño	20%	153.77208
Programación	40 %	307.54416
Pruebas	15 %	115.32906
Sobrecarga (otras actividades)	15 %	115.32906
<b>Total de horas</b>	100 %	768.8604

#### 3.6.4 Cálculo de costos.

Tomando como salario promedio mensual \$315.00

**Costo**= 3 meses \* \$315 mensual = **\$945.00**

#### 3.6.5 Análisis de los costos y beneficios.

El módulo implementado para Moodle, no implica costo alguno para nuestro centro de estudio o cualquier otra entidad donde se pretenda implantar, sin embargo, al desarrollarlo debemos asociarle un costo y su justificación económica dado por los beneficios tangibles e intangibles. Además el estudio de factibilidad, permitió constatar la necesidad de la elaboración de un

cronograma, que distribuido de manera eficaz, contribuya al éxito del desarrollo del proyecto. Para la realización de la aplicación no se incurrió en gastos adicionales de equipamiento, compra de algún otro sistema necesario, ni de herramientas de desarrollo y además no hubo necesidad de contratar personal calificado que realizara el trabajo imprescindible para obtener el producto final. Cuando analizamos los costos apreciamos que son bajos, lo cual, junto a los grandes beneficios de la realización y utilización del sistema que se propone, determina la factibilidad del desarrollo del mismo.

### **Beneficios tangibles e intangibles.**

Dentro de los beneficios tangibles que tiene la realización de este proyecto podemos referirnos a que mejoró el proceso de gestión de la docencia en la plataforma Moodle, así como el de generación de exámenes; ya que el mismo permite el almacenamiento y procesamiento de gran cantidad de información en poco tiempo haciendo este proceso mucho más rápido y aliviando enormemente la carga laboral de los docentes.

### **3.9 Diseño de pruebas funcionales.**

En el control de calidad de desarrollo de software, un aspecto crucial a tener en cuenta, es precisamente las pruebas y dentro de éstas, las funcionales. Mediante las cuales se hace una verificación dinámica del comportamiento de un sistema, basada en la observación de un conjunto seleccionado de ejecuciones controladas o casos de prueba. Se requiere para ello una planificación que consiste en definir los aspectos a chequear y la forma de verificar su correcto funcionamiento, punto en el cual adquieren sentido los casos de prueba.

### 3.9.1 Caso de prueba para el Caso de Uso Gestionar Niveles de asimilación.

Página de editar preguntas donde se le asigna un nivel de asimilación.  
**Agregando pregunta verdadero/falso** 

[▶ Expandir todo](#)

▼ General

---

Categoría

Nombre de la pregunta\*

Niveles de Asimilación

Enunciado de la pregunta\*



**Ilustración 9. Vista de asignar nivel de asimilación a pregunta**

Validaciones:

La Validación se hace en el Evento “OnClick” del Botón Guardar cambios, con las siguientes reglas:

- El combobox Niveles de Asimilación, es obligatoria la elección de uno de los 3 tipos que muestra. Si la validación no tuvo éxito se mostrará un “\*” al lado del campo.

En el evento del botón Guardar cambios se muestra un mensaje de éxito o de error según sea el caso en la Zona de Errores.

Zona de Errores y de Éxitos: Ubicada en la parte superior de la vista.

### 3.9.2 Caso de prueba para el Caso de Uso Generar Examen.

Página de generar examen a partir de preguntas aleatorias.

## Pregunta aleatoria desde una categoría existente

Categoría

Incluir también preguntas de subcategorías

Número de preguntas al azar

Niveles de Asimilación

## Pregunta aleatoria utilizando una categoría nueva

Nombre

Ilustración 10. Vista e generar examen.

Validaciones:

La Validación se hace en el Evento “OnClick” del Botón Agregar pregunta aleatoria y Crear categoría y agregar pregunta aleatoria, con las siguientes reglas:

- El combobox Categorías, Números de preguntas al azar, Niveles de Asimilación y Categoría padre es obligatoria la elección. En caso de no seleccionar alguna de las opciones el sistema muestra un “\*” al lado del campo.
- El campo Nombre es un String y es requerido. En caso de que el dato insertado no cumpla con la condición el sistema muestra un mensaje de error.

En el evento del botón Agregar pregunta aleatoria y Crear categoría y agregar pregunta aleatoria se muestra un mensaje de éxito o de error según sea el caso en la Zona de errores.

Zona de Errores y de Éxitos: Ubicada en la parte superior de la vista.

### 3.9.3 Caso de prueba para el Caso de Uso Exportar examen.

Página de exportar examen a PDF.

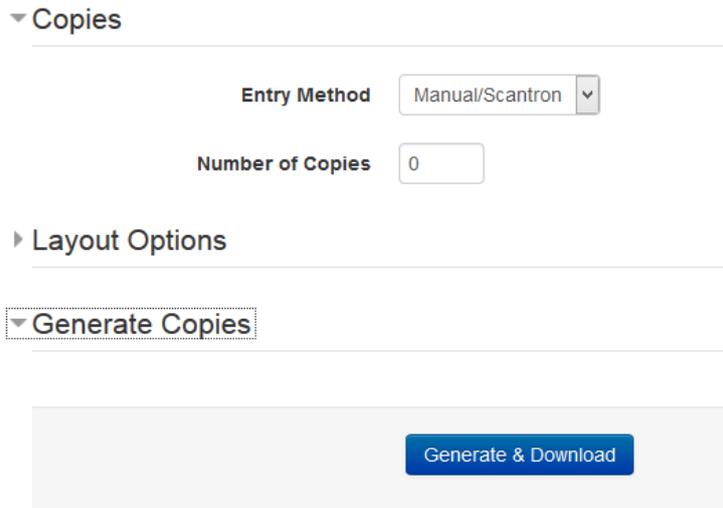


Ilustración 11. Vista de exportar examen a PDF

Validaciones:

La validación se hace en el evento "OnClick" del botón Generate & Download, con las siguientes reglas:

- El campo Número de copias es un número entero positivo. En caso de que el dato insertado no esté incluido en este conjunto numérico, el sistema mostrará al lado derecho del campo el mensaje: El número de copias debe ser un número entero positivo.
- El combobox Entrar método es obligatoria la elección. En caso de no seleccionar alguna de las opciones el sistema muestra un "\*" al lado del campo.

En el evento del botón Agregar pregunta aleatoria y Crear categoría y agregar pregunta aleatoria se muestra un mensaje de éxito o de error según sea el caso en la Zona de errores.

Zona de Errores y de Éxitos: Ubicada en la parte superior de la vista.

### 3.10 Conclusiones del capítulo.

En este capítulo se definieron las clases del diseño y se representaron en el diagrama de clases del diseño. Se describió el diseño de la base de datos a través de los diagramas de clases persistentes y del modelo físico de datos y se representaron los componentes del sistema en el diagrama de implementación. Además, se describieron los principios del diseño seguidos en el sistema propuesto, profundizando específicamente en los estándares de la interfaz y el tratamiento de excepciones. Por último se explicó el estándar de codificación seguido en la implementación. Además como método para la estimación se utilizó COCOMO II. Las estimaciones realizadas arrojaron como resultado que para la realización del proyecto el tiempo sería aproximadamente de 3 meses empleando 1 persona y esto tendría un costo aproximado de \$945.00 en moneda nacional.

Este estudio de planificación y factibilidad proporcionó valiosos argumentos que permitieron llegar a la conclusión de que la solución de software, es factible y reportará significativos beneficios a los profesores de la UCF.

## Conclusiones

Los resultados de este trabajo de fin de carrera responden a los objetivos planteados en el inicio del proyecto.

- Se logró sistematizar la documentación para desarrolladores tomando como fuente el Anexo 1 Arquitectura del Moodle y el Anexo 2 Guía para desarrolladores.
- Se desarrolló un módulo para la generación automática de exámenes que integra elementos propios del modelo docente educativo cubano.
- Se realizó el diseño de los casos de pruebas funcionales que permitió obtener una verificación del comportamiento del sistema, garantizando de esta forma el control de la calidad en el desarrollo del software.
- Se concluyó que fue factible desarrollar el sistema, utilizando para ello el método de estimación Puntos de Casos de Uso.

En cuanto a la documentación para desarrolladores, se han examinado varias fuentes disponibles en Internet y se ha realizado una recopilación de todos los elementos que el desarrollador novel debe tener en cuenta antes de comenzar su trabajo. Además se ha realizado una sencilla guía en la que se explica paso por paso cómo realizar un módulo para Moodle que disponga de las funcionalidades básicas. A partir de ahí será tarea del desarrollador profundizar lo que considere necesario para añadir todas las funcionalidades que requiera su aplicación.

El módulo para la generación automática de exámenes ofrece funcionalidades que permiten al docente asignarle niveles de asimilación a las preguntas creadas y almacenadas en la BD del Moodle. Además permite generar exámenes a partir de la selección de objetivos a evaluar y preguntas categorizadas por objetivos y niveles de asimilación; evitando la duplicidad de los mismos en un período seleccionado por el profesor.

Finalmente, se ha redactado el presente informe (memoria) que cuenta, entre otros apartados, con Anexos que incluyen una descripción detallada de la arquitectura de Moodle, un listado detallado de los recursos y librerías de

---

código disponibles para desarrolladores, y un manual que explicará los pasos a seguir a la hora de desarrollar un nuevo módulo para la plataforma.

Una vez alcanzados los objetivos que se plantearon en el anteproyecto, a nivel personal este trabajo me ha servido para consolidar mis conocimientos de programación de aplicaciones Web (en concreto en el lenguaje PHP), así como para aprender a administrar una plataforma de enseñanza virtual.

Finalmente, es importante señalar que con el desarrollo de este módulo de alguna manera, hemos logrado poner en sintonía el objetivo educativo de la plataforma con el objetivo educativo de la educación superior cubana.

## *Recomendaciones*

Por otro lado, aunque el módulo es completamente funcional, tras los ensayos realizados han salido a la luz nuevos requisitos funcionales que no es posible desarrollar dentro del entorno de este proyecto de fin de carrera debido a limitaciones temporales, pero que sería bueno tener en cuenta para futuras versiones de la aplicación.

1. Integrar otros elementos a tener en cuenta en la evaluación, como los niveles de dificultad.
2. Integrar los objetivos de la asignatura en el propio diseño del curso.
3. Crear un grupo de desarrolladores en la Universidad.
4. Mejorar el proceso de generación automática de exámenes a partir de las necesidades particulares de cada estudiante, de manera que se pueda potenciar el nivel de aprendizaje.

---

## Referencias bibliográficas

- [1] R. S. Raúl López, "Curso de Moodle. Una necesidad actual de las universidades." 2005.
- [2] Sánchez, Jaime., "Comunidades virtuales de aprendizaje: Conceptos e ideas.," 2002. [Online]. Available: <http://www.dcc.uchile.cl/~jsanchez>. (5/1/2015)
- [3] Correa Gorospe, J.M., "La integración de plataformas de e-learning en la docencia universitaria: Enseñanza, aprendizaje e investigación con Moodle en la formación inicial del profesorado, Revista Latinoamericana de Tecnología Educativa.," p. 4 (1), 37-48., 2005.
- [4] RENGARAJAN, R., "LCMS and LMS: Taking Advantage of Tight Integration.," 2001. [Online]. Available: [http://www.e-learn.cz/soubory/lcms\\_and\\_lms.pdf](http://www.e-learn.cz/soubory/lcms_and_lms.pdf). (5/1/2015)
- [5] Centre D'Educació i Noves Tecnologies, "Selección de un entorno virtual de enseñanza/aprendizaje de código fuente abierto para la Universitat Jaume I," 2004. [Online]. Available: <http://cent.uji.es>. (12/3/2015)
- [6] BALLESTER, A, "El Aprendizaje Significativo en la Práctica. Cómo hacer el Aprendizaje Significativo en el Aula. España." 2002.
- [7] Herrera Batista, Miguel A., "Las fuentes de aprendizaje en ambientes virtuales educativos. Revista Iberoamericana de Educación.," 2002.
- [8] B. MARCELO, C. M., "e-learning, Teleformación, Diseño, Desarrollo y Evaluación de la Formación a través de Internet ."
- [9] JOIN, "Evaluación de las plataformas LMS." 2005.
- [10] STALLMAN, R M., "Software Libre para una Sociedad Libre.Madrid: GNU Press.," 2004.
- [11] ROMERO, T.A., "Moodle, Unimos Mentes, Creamos Conocimiento Libre. Ponencia presentada al VI Congreso Internacional Virtual de Educación CIVE 2006, España." 2006.
- [12] GÓMEZ, J., "Moodle 1.5 Manual de Consulta. Propuesta Pedagógica. Valladolid: GNU Press." 2006.
- [13] C. de M. GATE, "Manual de estilo de código."
- [14] Dr. Cs. Carlos M. Alvarez de Zayas, *La escuela en la vida*, vol. Cap-3. .

- [15] “Tester v2.0 Generador de test dirigido a profesores.,” 10-Feb-2015. [Online]. Available: <http://www.archivospc.com/programas/categorias/Generador%20de%20Ex%E1menes.php?> (12/3/2015)
- [16] “Generador de exámenes ExamView.,” de enero de-2015. [Online]. Available: <http://espanol.pearsonncs.com/software/examview.htm>. (16/4/2015)
- [17] “Exámenes v1.20. Herramienta que permite confeccionar, almacenar y ordenar exámenes diferentes.,” de enero de-2015. [Online]. Available: <http://www.tinitasoft.com/programaexa.htm>. (16/4/2015)
- [18] “Generador de exámenes Any Test 2.50.,” 25-Feb-2015. [Online]. Available:<http://www.archivospc.com/programas/categorias/Generador%20de%20Ex%E1menes.php?> (16/4/2015)
- [19] “TestIt 3.0 Build 110. El programa ideal para la creación de exámenes de todo tipo.,” 25-Feb-2015. [Online]. Available: <http://www.archivospc.com/programas/categorias/Generador%20de%20Ex%E1menes.php>. (16/4/2015)
- [20] Alex Guerra, “Sistema generador de exámenes para las asignaturas de las carreras de Universalización.”. UCF, 2007
- [21] G. B. Ivar Jacobson, *El proceso unificado de desarrollo de software.*, Ed. Félix Varela. La Habana, 2004.
- [22] PHPXREF: MOODLE, “Referencia del código fuente de Moodle,” 2010. [Online]. Available: <http://xref.moodle.org/nav.html?index.html>. (20/4/2015)
- [23] Daniel Fernández, “Definición de una arquitectura software para el diseño de aplicaciones web basadas en tecnología Java-J2EE.” Estudios de Doctorado Avances en Informática, Universidad de Oviedo.
- [24] LARMAN, Craig, *UML y patrones. Introducción al análisis y diseño orientado de objetos.*, Ed. Félix Varela., vol. – 2 t. La Habana, 2004.

## *Bibliografía*

- [1]BARTOLOMÉ, A., “Blended Learning, Conceptos Básicos. Píxel-Bit Revista de Medios y Educación,” pp. 23, 7–20, 2004.
- [2]Sánchez, Jaime., “Comunidades virtuales de aprendizaje: Conceptos e ideas.,” 2002. [Online]. Available: <http://www.dcc.uchile.cl/~jsanchez>. (5/1/2015)
- [3]R. S. Raúl López, “Curso de Moodle. Una necesidad actual de las universidades.” 2005.
- [4]G. P. Margelys Hernández, “Curso de superación para docentes ‘Moodle y su aplicación a distancia.’” Cienfuegos-2015.
- [5]Daniel Fernández, “Definición de una arquitectura software para el diseño de aplicaciones web basadas en tecnología Java-J2EE.” Estudios de Doctorado Avances en Informática, Universidad de Oviedo.
- [6]Letelier TORRES, Patricio, “Desarrollo de Software Orientado a Objeto usando UML.,” 24-Mar-2015. [Online]. Available: <http://www.creangel.com/uml/creditos.php>. (5/1/2015)
- [7]Ferrando Bravo G and Moreno Bonett A, “Educación continua a distancia: modelos, entornos, desarrollo y especificaciones,” 2004. [Online]. Available: <http://espacio.uned.es>. (12/3/2015)
- [8]BALLESTER, A, “El Aprendizaje Significativo en la Práctica. Cómo hacer el Aprendizaje Significativo en el Aula. España.” 2002.
- [9]Dávila Espinosa, Sergio, “El aprendizaje significativo. Esa extraña expresión. Contexto Educativo.,” 2000. [Online]. Available: <http://contexto-educativo.com.ar>. (12/3/2015)
- [10]B. MARCELO, C. M., “e-learning, Teleformación, Diseño, Desarrollo y Evaluación de la Formación a través de Internet .” .

- [11]G. B. Ivar Jacobson, El proceso unificado de desarrollo de software., Ed. Félix Varela. La Habana, 2004.
- [12]“Entorno de desarrollo de software: PHP.” [Online]. Available: <http://es.wikipedia.org/wiki/php>. (20/4/2015)
- [13]JOIN, “Evaluación de las plataformas LMS.” 2005.
- [14]“Exámenes v1.20. Herramienta que permite confeccionar, almacenar y ordenar exámenes diferentes.” de enero de-2015. [Online]. Available: <http://www.tinitasoft.com/programaexa.htm>. (20/4/2015)
- [15]“Generador de exámenes Any Test 2.50.” 25-Feb-2015. [Online]. Available: <http://www.archivospc.com/programas/categorias/Generador%20de%20Ex%E1menes.php?> (16/4/2015)
- [16]“Generador de exámenes ExamView.” de enero de-2015. [Online]. Available: <http://espanol.pearsonncs.com/software/examview.htm>. (16/4/2015)
- [17]MATOS, Rosa María, Introducción al trabajo con Base de Datos: material para uso docente, La Habana: [s.n]. 2001.
- [18]Dr. Cs. Carlos M. Alvarez de Zayas, La escuela en la vida, vol. Cap-3. .
- [19]Correa Gorospe, J.M., “La integración de plataformas de e-learning en la docencia universitaria: Enseñanza, aprendizaje e investigación con Moodle en la formación inicial del profesorado, Revista Latinoamericana de Tecnología Educativa.” p. 4 (1), 37-48., 2005.
- [20]Herrera Batista, Miguel A., “Las fuentes de aprendizaje en ambientes virtuales educativos. Revista Iberoamericana de Educación.” 2002.
- [21]C. C. Rundle M, “Las tecnologías, una historia breve. En: Tecnologías emergentes: un estudio sobre sus consecuencias éticas.París: UNESCO.” pp. p. 4–6., 2007.

- [22]RENGARAJAN, R., "LCMS and LMS: Taking Advantage of Tight Integration.," 2001. [Online]. Available: [http://www.e-learn.cz/soubory/lcms\\_and\\_lms.pdf](http://www.e-learn.cz/soubory/lcms_and_lms.pdf). (5/1/2015)
- [23]LÓPEZ GUZMÁN, C., "Los repositorios de objetos de aprendizaje como soporte para los entornos e-learning.," 2005. [Online]. Available: [http://www.biblioweb.dgsca.unam.mx/libros/repositorios/la\\_web.htm](http://www.biblioweb.dgsca.unam.mx/libros/repositorios/la_web.htm). (20/4/2015)
- [24]C. de M. GATE, "Manual de estilo de código."
- [25]C. de M. GATE, Manual de Moodle.
- [26]GÓMEZ, J., "Moodle 1.5 Manual de Consulta. Propuesta Pedagógica. Valladolid: GNU Press." 2006.
- [27]ROMERO, T.A., "Moodle, Unimos Mentes, Creamos Conocimiento Libre. Ponencia presentada al VI Congreso Internacional Virtual de Educación CIVE 2006, España." 2006.
- [28]MOODLE, "Página oficial de Moodle," 2010. [Online]. Available: <http://moodle.org/>. (24/4/2015)
- [29]PHPXREF: MOODLE, "Referencia del código fuente de Moodle," 2010. [Online]. Available: <http://xref.moodle.org/nav.html?index.html>. (24/4/2015)
- [30]Centre D'Educació i Noves Tecnologies, "Selección de un entorno virtual de enseñanza/aprendizaje de código fuente abierto para la Universitat Jaume I," 2004. [Online]. Available: <http://cent.uji.es>. (12/3/2015)
- [31] "Sistema de gestión de bases de datos Microsoft SQL Server." [Online]. Available: [http://es.wikipedia.org/wiki/SQL\\_Server](http://es.wikipedia.org/wiki/SQL_Server). (24/4/2015)
- [32] "Sistema de gestión de bases de datos MySQL.," 2015.
- [33]Alex Guerra, "Sistema generador de exámenes para las asignaturas de las carreras de Universalización.". UCF, 2007.

---

[34]STALLMAN, R M., “Software Libre para una Sociedad Libre.Madrid: GNU Press.,” 2004.

[35]M. P. José Hernández Ortega, Tendencias emergentes en educación con TICS. Asociación Espiral, Educación y Tecnología, 2012.

[36] “Tester v2.0 Generador de test dirigido a profesores.,” 10-Feb-2015. [Online]. Available:

<http://www.archivospc.com/programas/categorias/Generador%20de%20Ex%E1menes.php?> (12/3/2015)

[37] “TestIt 3.0 Build 110. El programa ideal para la creación de exámenes de todo tipo.,” 25-Feb-2015. [Online]. Available:

[http://www.archivospc.com/programas/categorias/Generador%20de%20Ex%E1menes.php.](http://www.archivospc.com/programas/categorias/Generador%20de%20Ex%E1menes.php) (20/4/2015)

[38]RODRÍGUEZ MARTÍN, F.J., “Tutorial para la creación de un Módulo en Moodle.,” 2007. [Online]. Available:

[http://moodle.org/pluginfile.php/226/mod\\_forum/attachment/335652/Tutorial\\_M\\_dulos.pdf.](http://moodle.org/pluginfile.php/226/mod_forum/attachment/335652/Tutorial_M_dulos.pdf) (20/4/2015)

[39]LARMAN, Craig, UML y patrones. Introducción al análisis y diseño orientado de objetos., Ed. Félix Varela., vol. – 2 t. La Habana, 2004.

---

*Glosario de término*

**SGA:** Sistemas de Gestión de Aprendizaje

**LMS:** Learning Management System

**CMS:** Content Management System

**LCMS:** Learning Content Management System

**TICs:** Tecnologías de la Información y las Comunicaciones

**Moodle:** Module Object-Oriented Dynamic Learning Environment(Ambiente de Aprendizaje Modular Orientado a Objetos Dinámicos)

## *Anexos*

### **A1. Arquitectura del Moodle.**

#### **Estructura de directorios de Moodle**

Moodle significa Modular Object-Oriented Dynamic Learning Environment (Entorno de enseñanza dinámico orientado a objetos modular). Desde la perspectiva del programador, la "M" de Moodle contiene un concepto muy importante, ya que Moodle, en su totalidad, está diseñado de forma modular. Esto permite al desarrollador realizar importantes modificaciones a la plataforma sin necesidad de cambiar el código fuente. La modularidad se convierte en una importante característica que permite reducir la cantidad de tiempo empleada en realizar modificaciones cuando se lanzan nuevas versiones de Moodle.

Cada sistema Moodle en funcionamiento se puede dividir en tres áreas separadas: código, datos y base de datos.

#### ***Código de Moodle***

Como PHP es un lenguaje interpretado, el código de Moodle se almacena como ficheros de código fuente en el servidor web. Cuando un fichero en particular es solicitado por el servidor, el intérprete PHP pasea el código al instante y la salida resultante se envía a través del servidor web. Como se ha mencionado antes, la "M" de Moodle significa Modular, y su estructura de directorios refleja esta modularidad, ya que cada carpeta del nivel superior representa un componente de Moodle. Algunos de los principales componentes soportan módulos de expansión o plugins. Cada módulo de expansión tiene su propia carpeta dentro del directorio del componente. En algunos casos, los módulos de actividad también pueden tener soporte para expansiones adicionales. Desde el punto de vista del usuario final, los módulos se instalan copiando su carpeta dentro del directorio apropiado en el servidor. Moodle detecta el nuevo módulo la siguiente vez que un administrador accede al

sistema, localiza el código SQL del módulo, lo ejecuta y finalmente muestra por pantalla los resultados.

Los procesos de actualización se realizan de la misma manera, haciendo un seguimiento de la versión de la base de datos y actualizando automáticamente si es necesario.

A continuación, exploraremos algunos de los directorios más importantes utilizados por los desarrolladores que desean realizar cambios a Moodle. Moodle utiliza una nomenclatura simple para sus módulos, cuyos archivos se almacenan en su propia carpeta, y el nombre de esta carpeta es el nombre que Moodle muestra en su interfaz cuando se referencia al módulo.

- **admin:** Esta carpeta almacena los ficheros PHP que controlan la interfaz de los usuarios administradores. También contiene el fichero `cron.php`, que se ejecuta como un proceso batch para realizar tareas de mantenimiento del sistema como el envío de correos y la realización de las copias de seguridad de los cursos. A menudo se utiliza el proceso cron para realizar este tipo de operaciones programadas.
- **auth:** La carpeta auth contiene todos los módulos de autenticación de Moodle. Cada módulo tendrá su propio directorio dentro de esta carpeta. Los módulos de autenticación controlan la creación de usuarios y el acceso de estos al sistema.
- **backup:** Esta carpeta contiene las utilidades de copia de seguridad del núcleo del sistema. Se trata de funciones para guardar, restaurar e importar cursos. Cada módulo individual es responsable de realizar la copia de seguridad de su código y puede hacer uso de estas funciones si lo necesita. Como cada módulo es auto-contenido, se permite a los desarrolladores añadir módulos limpiamente a Moodle sin tener que modificar el código del núcleo.
- **blocks:** Los bloques se usan para mostrar bloques de información en la columna de la izquierda o de la derecha de la página de Moodle. Son uno de

los tipos más simples de módulos que se pueden realizar, y suelen funcionar a través de las versiones de Moodle sin apenas alguna modificación.

- **course:** Este componente de Moodle tiene una importancia obvia, dado que la plataforma se organiza en cursos. Como desarrolladores, podemos desear añadir o modificar formatos de cursos e informes. Los formatos de curso personalizados se pueden usar para cambiar la disposición de los elementos en los cursos.
- **enrol:** La carpeta enrol contiene todos los módulos de matriculación de Moodle. Estos módulos controlan la creación y administración de las matriculaciones a nivel de curso.
- **files:** El componente files permite a Moodle incorporar archivos al sistema. Esto incluye subida de archivos, control de acceso, y visualización de ficheros.
- **filter:** El sistema de filtros de Moodle es una facilidad de búsqueda y reemplazo basada en expresiones de texto y expresiones regulares. Este sistema es alimentado durante la creación de la página por el contenido introducido por el usuario que se encuentra en la base de datos. Los filtros encuentran coincidencias y modifican la página antes de ser mostrada. Por ejemplo, existe un filtro que soporta auto-conversión del lenguaje de marcado TEX a gráficos de ecuaciones. El filtro de plugins multimedia encuentra referencias a tipos comunes de archivos multimedia y los envuelve en sus apropiadas etiquetas, para embeber automáticamente el contenido multimedia en la página. Aunque esta es una capacidad muy potente, debe ser desarrollada con cuidado, teniendo en mente las implicaciones de rendimiento.
- **lang:** La carpeta lang almacena las cadenas de idioma del núcleo del sistema. Esta es la base del soporte de localización e idioma de Moodle. Todas las cadenas mostradas al usuario final se asignan a través de esta utilidad. Las

cadena de idioma se almacenan en la carpeta lang de los datos de Moodle. Esta estructura permite una personalización local muy sencilla de los ficheros de idioma.

- lib: La carpeta lib almacena las librerías de funciones del núcleo del sistema. Cuando se desarrollen módulos o se personalice el sistema, se usarán clases y funciones definidas en esta carpeta.
- mod: La carpeta mod almacena los módulos de actividad como Tarea, Wiki, Lección o Foro. Las actividades educativas son el núcleo de cada curso desarrollado en Moodle. Los módulos de actividad son más difíciles de crear que los bloques y deben diseñarse de manera que instruyan al estudiante.
- my: my es un portal ligero en Moodle. Proporciona la lista de cursos en las que un usuario participa, incluyendo un resumen de las actividades próximas. El usuario puede también añadir y borrar bloques en su portal. Este componente proporciona una buena localización para mostrar información personalizada con mínimos cambios en el núcleo de Moodle.
- theme: La carpeta theme almacena todos los temas prediseñados de Moodle y cualquier tema personalizado instalado en el sistema. Los temas son una combinación de CSS, HTML y PHP. Cada tema tiene su propia carpeta. El sistema de temas es útil para definir el aspecto visual, la cabecera y el pie de la página de Moodle. Sin embargo está limitado en la cantidad de elementos de la página que puede modificar. Por ejemplo, ciertos componentes de la página de Moodle están escritos para que se muestren en una determinada forma.

### ***Base de Datos del Moodle***

La base de datos de Moodle está organizada en aproximadamente doscientas tablas relacionadas. La instalación por defecto añade el prefijo mdl\_ al nombre de cada una de las tablas. Cada componente principal del sistema normalmente tiene una o más tablas, cada una comenzando con el nombre del componente. Por ejemplo, hay dos tablas relacionadas con el componente config: mdl\_config y mdl\_config\_plugins. Como programadores, tendremos que

modificar la base de datos de forma regular. Por tanto es importante para nosotros ser capaces de tratar la base de datos completa como una entidad, copiando y moviendo instancias de una base de datos completa de Moodle para la creación de áreas de pruebas en las que desarrollar nuestro código. Normalmente haremos esto con herramientas de línea de comandos como mysqldump y MySQL.

### ***Datos del Moodle***

Los ficheros subidos por los usuarios al sistema se almacenan en la localización moodledata. En esta carpeta también se almacenan los datos de sesión de los usuarios que se han logueado en el sistema si las sesiones basadas en fichero están configuradas. En los datos de Moodle también se almacenan paquetes de idioma opcionales que pueden ser descargados desde la interfaz de administración. Moodle estructura los datos de esta carpeta bien por usuarios o bien por cursos. Cada curso tiene una carpeta, nombrada con un número entero que representa el identificador único asignado a dicho curso en la base de datos. Podemos determinar fácilmente estos valores navegando en un curso a través de la página de Moodle e inspeccionando los parámetros en la URL. Por ejemplo, examinando esta URL de un curso: <http://localhost/workspace/moodle19/course/view.php?id=3>

Podemos ver que id=3, al final de la cadena, será el identificador de nuestro curso. Si hemos subido ficheros a este curso, existirá una carpeta moodledata/3. En esta carpeta, Moodle almacena datos de sus módulos en la subcarpeta moddata. Cuando un módulo necesita almacenar ficheros, los almacena dentro de esta carpeta en un directorio con el mismo nombre que el módulo. Por ejemplo, el módulo Wiki tendrá aquí una carpeta llamada wiki. Además, Moodle creará una carpeta llamada backupdata si se han realizado copias de seguridad de algún curso. Los ficheros que han sido subidos directamente por un usuario usando la interfaz de ficheros del curso se encontrarán en la raíz de esta carpeta. Los usuarios pueden crear también sus propias carpetas y subcarpetas dentro del directorio raíz.

Moodle 2.8 utilizará un modelo completamente nuevo para organizar los ficheros subidos por el usuario basado en un algoritmo hash. El principal objetivo de este nuevo método es conseguir un uso eficiente del espacio de almacenamiento en disco y una mayor flexibilidad a la hora de compartir ficheros entre múltiples cursos

### **Estructura del Moodle.**

Desde el punto de vista del usuario, un sitio Moodle está compuesto por: categorías, cursos, temas (o semanas) y actividades.

Los principales elementos que componen un sitio Moodle son los siguientes:

- ✓ **Categorías:** Las categorías son los contenedores de información de más alto nivel, están formadas por cursos y sirven para organizarlos de manera que sean más fácilmente localizables por el alumno en la pantalla inicial de la aplicación.
- ✓ **Cursos:** Los cursos son la parte más importante de la estructura de Moodle. Son creados por los administradores del sitio y dirigidos por los usuarios que se establezcan como profesores del curso. Para que un alumno, dado de alta en el sistema, pueda acceder a un curso deberá estar matriculado en él. Cuando un administrador crea un curso debe proceder a su configuración mediante un formulario proporcionado por Moodle en el que se establecen valores para los distintos campos, como por ejemplo el nombre del curso o el formato (semanal, por temas...).
- ✓ **Semanas y temas:** La organización de un curso podrá llevarse a cabo por semanas o bien por temas, según la preferencia del profesorado. Tras la creación del curso, su configuración, y la matriculación de usuarios, estos podrán acceder al mismo y observar una serie de bloques diferenciados que representan las semanas del curso o temas, según el formato que se haya establecido. Cada uno de estos bloques contendrá, a partir del momento en el que el profesor las añada, distintos tipos de actividades que los alumnos deberán realizar para su evaluación.

- ✓ Actividades: Moodle ofrece la posibilidad de añadir a cada semana o a cada tema distintos tipos de actividades. Existen siete actividades instaladas por defecto, a las que se pueden añadir todas las que el usuario desee.
- ✓ Recursos: Para complementar el contenido de las semanas o temas de cada curso, Moodle, además de actividades, ofrece la posibilidad de agregar otro tipo de recursos a las mismas.
- ✓ Bloques: Las funcionalidades extra fuera de las actividades y recursos son implementadas por los bloques, contenedores que se sitúan a los lados del sitio Web y que tienen una función concreta de carácter general (relacionada o no con los módulos de actividades. A continuación haremos un repaso por todas las actividades, recursos y bloques que componen el paquete estándar de instalación de Moodle.

### **Actividades**

Las actividades con las que cuenta Moodle por defecto son las que se muestran en la tabla siguiente. Desde la página de Moodle se pueden descargar otras muchas con distintas funcionalidades, e incluso nosotros mismos podemos desarrollar nuestra actividad personalizada. En el Anexo **A2** se profundizará en el proceso de desarrollo de un módulo de actividad para Moodle.

- ✓ Base de datos: El módulo Base de datos permite al profesorado y/o a los estudiantes construir un banco de registros sobre cualquier tema o asunto, así como realizar búsquedas y mostrar resultados. El formato, cuyo formulario puede ser creado por el profesor, y la estructura de estas entradas pueden ser prácticamente ilimitados, incluyendo imágenes, archivos, direcciones URL, números y texto, entre otras cosas.
- ✓ Chat: El chat de Moodle permite mantener conversaciones entre usuarios en tiempo real, generando así una interacción fluida mediante texto síncrono. Incluye las fotos de los perfiles en la ventana de chat y soporta direcciones URL, emoticonos, integración de HTML, imágenes, etc.

- ✓ Consulta: Es similar a una encuesta, donde el profesor formula una única pregunta y ofrece a los usuarios distintas elecciones. Se puede considerar una votación, y puede usarse para opinar sobre algún tema o para recibir una respuesta de cada estudiante (por ejemplo, para pedir su consentimiento para algo).
- ✓ Cuestionario: Permite la realización de exámenes de diferente tipo, donde los profesores pueden definir una base de datos de preguntas que podrán ser reutilizadas en diferentes cuestionarios.
- ✓ Encuesta: Es similar a la actividad Consulta, pero con varias preguntas de distinto tipo.
- ✓ Foro: Se trata de una actividad para el debate entre usuarios de un curso. Hay diferentes tipos de foros disponibles: exclusivos para los profesores, de noticias de un curso y abiertos a todos.
- ✓ Glosario: El glosario presenta un catálogo no exhaustivo de palabras relativas a una misma disciplina, campo de estudio, actividad, tema o área del saber, con su definición y/o comentarios, a veces bastante simples. Además puede o no estar clasificado en categorías y subcategorías.
- ✓ Hot Potatoes: Este módulo permite al profesorado elaborar y/o administrar los ejercicios elaborados con el programa Hot Potatoes a través de Moodle. Los ejercicios se crean con Hot Potatoes, y luego se incorporan al curso de Moodle.
- ✓ Lección: Una lección proporciona contenidos interactivos de forma interesante y flexible. Estos contenidos consisten en una serie de páginas (normalmente escritas con el editor HTML integrado en Moodle), y textos que el alumno ha de recorrer; al final de cada página se puede plantear opcionalmente una pregunta con varias posibles respuestas para comprobar de alguna manera que el alumno lo ha leído y/o comprendido. La navegación a lo largo de la lección depende de la interacción del estudiante con las preguntas planteadas.
- ✓ Recurso: Admite la presentación de cualquier contenido digital: Word, PowerPoint, Flash, vídeo, sonidos, etc. Los archivos pueden ser subidos y manejados en el servidor, o pueden ser creados sobre la marcha

- usando formularios web (de texto o HTML). También se pueden enlazar contenidos externos en Web o incluirlos perfectamente en la interfaz del curso, o realizar enlaces con aplicaciones Web, transfiriéndoles datos.
- ✓ SCORM: Un paquete SCORM (Sharable Content Object Reference Model) es un bloque de material web empaquetado de una manera que sigue el estándar SCORM de objetos de aprendizaje. Estos paquetes pueden incluir páginas web, gráficos, programas Javascript, presentaciones Flash y cualquier otra cosa que funcione en un navegador web. El módulo SCORM permite cargar fácilmente cualquier paquete SCORM estándar y convertirlo en parte de un curso.
  - ✓ Tarea: permite al profesor calificar trabajos enviados por los alumnos. Puede especificarse la fecha final de entrega de una tarea y la calificación máxima que se le podrá asignar. Los estudiantes pueden subir sus tareas (en cualquier formato de archivo) al servidor, y se registra la fecha en la que se han subido.
  - ✓ Wiki: Un Wiki posibilita la creación colectiva de documentos en un lenguaje simple de marcas, utilizando un navegador web.

### **Recursos**

Para complementar el contenido de las semanas o temas de cada curso, Moodle, además de actividades, ofrece la posibilidad de agregar otro tipo de recursos a las mismas.

- ✓ Editar una página de texto. Una página de texto es un texto normal mecanografiado sin formato (texto plano), es decir, sin ningún tipo de estilos (negrita, cursiva...) y sin estructuras (listas, tablas...). Es recomendable su uso en aquellos contenidos de carácter obligatorio y relativamente estable durante en curso, como por ejemplo normas de seguridad.
- ✓ Editar una página Web. La página de texto y, especialmente, la página web son dos recursos muy adecuados para publicar todo tipo de

---

contenidos. En concreto, la creación de página web nos permite añadir materiales curriculares propios o personalizados utilizando el editor integrado de Moodle.

- ✓ Enlazar un archivo o una página Web. El enlace a una web es útil para añadir contenidos externos al sitio Moodle de forma rápida o para facilitar el acceso y conocimiento de sitios de interés general (buscadores, diccionarios, instituciones, etc.), mientras que el enlace a un archivo nos permite agregar contenidos con múltiples formatos, comprimidos o no, con la ventaja adicional de que no es obligatorio que estén en el sitio Moodle (aunque requiere la configuración por parte del administrador).
- ✓ Directorio. Este tipo de recurso es, como su nombre indica, simplemente un acceso a un directorio o carpeta particular del sitio web del curso o de la unidad didáctica. Permite mostrar un directorio completo (junto con sus subdirectorios), previamente creado por el profesor o administrador, desde el área de archivos de su curso. De esta manera, podemos facilitar que con un único enlace los alumnos tengan acceso a toda una lista de ficheros que se pueden descargar a sus ordenadores.
- ✓ Desplegar paquetes de contenido IMS. IMS es un esqueleto de especificaciones que ayuda a definir variados estándares técnicos, incluyendo materiales de e- learning. La especificación IMS (Content Packaging Specification) hace posible almacenar los contenidos en un formato estándar que puede ser reutilizado en diferentes sistemas sin necesidad de convertir dichos contenidos a otros formatos. El recurso de tipo IMS CP en Moodle permite utilizar ese tipo de paquetes de contenidos, cargándolos e incluyéndolos en los cursos de forma sencilla.
- ✓ Añadir una etiqueta. Las etiquetas nos permiten añadir pequeños fragmentos de texto HTML, gráficos o elementos multimedia entre los bloques de contenido (en la columna central si se trata de un curso en formato semanal o por temas, o en las columnas laterales si se trata de un curso en formato social).

## **Bloques**

El aspecto del entorno que envuelve un curso en Moodle puede variar en función de las necesidades de alumnos y profesores. Esta personalización se consigue mediante el uso, por parte del profesor del curso, de los denominados bloques de Moodle, que aparecen a la izquierda o la derecha de la pantalla. Existen varios tipos de bloques con funcionalidades muy diversas: informar, controlar, gestionar, etc. Algunos trabajan de forma independiente y otros colaboran entre sí. A continuación se describen los bloques establecidos por defecto en Moodle. A ellos podemos agregar cuanto deseemos, de acuerdo con la funcionalidad que queramos instalar en la plataforma Moodle.

- ✓ Administración: Mediante este bloque el usuario podrá acceder a diferentes herramientas administrativas en función de su perfil. Desde estas opciones el alumno podrá examinar sus calificaciones, cambiar su contraseña o darse de baja de un curso (el profesor se encargará de decidir qué opciones estarán disponibles para sus alumnos). Por otra parte, el profesor puede configurar el aspecto de un curso (incorporando bloques por ejemplo), añadir nuevas actividades, dar de alta a nuevos usuarios o hacer copias de seguridad. Cuanto mayor sea nuestro nivel de privilegios, de más opciones podremos disponer.
- ✓ Actividad reciente: El bloque Actividad reciente muestra, de una forma abreviada, los acontecimientos producidos en el curso desde la última visita. Lo podemos utilizar para tener una visión rápida del desarrollo del trabajo realizado, tanto por nosotros mismos, como por el resto de usuarios del curso.
- ✓ Actividades: El bloque de Actividades lista todas las categorías de actividades disponibles en el curso (foros, consultas, tareas, etc.).
- ✓ Buscar en los foros: El bloque Buscar en los foros permite buscar entre los mensajes publicados en los foros de un curso la información tecleada en el cuadro de texto del bloque.
- ✓ Calendario: Este bloque puede usarse como una agenda personal, sirve para mantener una visión organizada de las fechas y plazos importantes para el seguimiento de la asignatura o de un curso: fechas de

- exámenes, de entrega de trabajos, reuniones de tutoría, etc. Existen cuatro categorías de eventos temporales que podemos controlar con el calendario y cada uno de ellos está identificado con un color de fondo en el bloque: eventos globales, eventos de curso, eventos de grupo y eventos de usuario.
- ✓ Canales RSS remotos: El bloque Canales RSS remotos permite mostrar contenidos de canales RSS de sitios Web externos. El administrador debe configurar el bloque para permitir al profesorado agregar y gestionar los canales. En caso contrario, sólo el administrador puede añadir o gestionar los canales del sitio Moodle.
  - ✓ Cursos: El bloque Cursos muestra un listado de todos los cursos en los que estamos matriculados o somos profesores/as, y puede utilizarse para moverse rápidamente entre esos cursos. Debajo de esta lista, aparece el vínculo Todos los cursos..., que lleva a una página donde se muestran todos los cursos del sitio agrupados por categorías (esto no quiere decir que se pueda entrar en ellos).
  - ✓ Descripción del curso/sitio: El bloque Descripción de curso/sitio muestra el texto introducido en el resumen (informe) del curso presente en el formulario de Configuración del curso o en la descripción del sitio que se realiza en la página de Ajustes de portada. La cabecera del bloque sólo se muestra en el modo de edición. Se pueden incluir imágenes y enlaces.
  - ✓ Enlaces de sección: El bloque Enlaces de Sección facilita la navegación rápida por las diferentes secciones o semanas del curso, dependiendo del formato del curso usado, Temas o Semanal. Los números que aparecen en el bloque son enlaces a las secciones numeradas del curso. Este bloque es especialmente útil en los cursos con un gran número de secciones y cuando tenemos una sola sección visible.
  - ✓ Entrada aleatoria del glosario: Es un bloque que permite mostrar, por ejemplo cada vez que se accede a la página principal de un curso, una nueva entrada del glosario elegido, o una frase célebre, una cita, o un refrán, de forma sencilla y colaborativa.

- ✓ **Eventos próximos:** El bloque Eventos próximos muestra una lista de los acontecimientos próximos (ya sea un evento global, de curso o de usuario) en el calendario, con enlaces al contenido del acontecimiento señalado. Los eventos se pueden programar en el calendario y todas las actividades de Moodle con fecha límite generan un evento automáticamente.
- ✓ **HTML:** El bloque HTML se utiliza para presentar en la página principal del sitio o de un curso información textual, gráfica, etc. Facilita un bloque flexible que puede incorporar variedad de funciones y usos, proporcionando un editor HTML para formatear el texto, añadiendo imágenes o creando enlaces. También es posible trabajar directamente con el código fuente, lo que permite introducir marcadores del lenguaje HTML. Eso hace posible incluir video, sonido, flash, y otros ficheros que añadirán elementos únicos a la página del curso.
- ✓ **Memtees:** El bloque Mentees, sólo disponible a partir de Moodle 1.8, es básicamente un bloque para que los padres de los alumnos puedan acceder a Moodle y puedan ver las actividades o calificaciones de sus hijos. Para añadir este bloque debemos tener instalada la versión 1.8, o superior, de Moodle, pues sólo funciona con esta versión.
- ✓ **Mensajes:** El bloque Mensajes constituye un sistema de mensajería interna del sistema que permite la comunicación directa entre todos los usuarios del sitio sin necesidad de usar el correo electrónico.
- ✓ **Novedades:** El bloque Novedades presenta las cabeceras de las últimas noticias o mensajes publicados por el profesor o profesora en el Foro de Noticias. Siguiendo el hipervínculo más... se accede al foro donde está publicada la noticia y se puede ver el mensaje completo. En principio, todos los estudiantes de un curso están suscritos a este foro, de forma que también recibirán estos mensajes en su correo particular.
- ✓ **Personas:** El bloque Personas contiene el enlace a Participantes, que es un listado de la totalidad de participantes del curso. Por defecto, aparece primero el profesorado y después los alumnos, ordenados por el último acceso al curso. Al hacer clic sobre el nombre de algún participante nos muestra su perfil personal.

- ✓ Resultados del cuestionario: El bloque Resultados del cuestionario muestra información sobre los resultados alcanzados por los usuarios en un determinado cuestionario en tiempo real.
- ✓ Usuarios en línea: El bloque de los Usuarios en línea muestra a los usuarios matriculados (no muestra los invitados) del curso actual y presentes en un período de tiempo fijado por el administrador del sitio (por defecto son los últimos 5 minutos).

### **Control de acceso, inicio de sesión y roles de usuario**

Al ser una plataforma tan abierta, Moodle debe ser riguroso con su seguridad. Por ello hace grandes esfuerzos para asegurar que ninguna persona acceda al sistema si no debe hacerlo, y que los usuarios que acceden lo hagan de la manera correcta. Moodle cuenta con un potente sistema de permisos y control de acceso, cuyo núcleo son las cuentas de usuario. Aunque es posible permitir el acceso a un curso a cualquier visitante sin necesidad de autenticación, a estos usuarios invitados no les está permitido hacer algunas cosas importantes en el sistema. Lo usual es que queramos saber quiénes son nuestros usuarios, y que cada uno tenga su propia cuenta.

Una cuenta de usuario proporciona acceso individual al sistema mediante un nombre de usuario y una contraseña. Estas cuentas se crean utilizando los plugins de autenticación. La manera más simple de crear una cuenta de usuario es hacerlo de forma manual, creando un usuario mediante la interfaz de administración. Sin embargo, el sistema más común de autenticación es el basado en e-mails, que permite a los usuarios crear sus propias cuentas mediante la verificación de e-mail. En cualquier caso, se requiere un nombre de usuario y una dirección de correo electrónico, además de una contraseña. Las contraseñas en Moodle se codifican con una función hash MD5 para hacerlas ilegibles y difíciles de averiguar.

Para acceder al sistema, un usuario introduce su usuario y contraseña. Si son correctos se le permite el acceso al sitio. El sistema de sesiones de Moodle utiliza funciones de cookies de PHP para introducir cookies en la sesión actual y ayudar a identificar al usuario durante toda su visita al sistema.

## **Permisos**

Los permisos en Moodle pueden ser asignados dentro de seis contextos: sitio/global, categoría de curso, curso, bloques y actividades, usuario y portada. Existen siete roles integrados: administrador, profesor, profesor no editor, estudiante, creador de cursos, usuario autenticado e invitado, y todos ellos pueden ser asignados en uno o más de los contextos mencionados. Como desarrolladores, podemos crear capacidades para controlar el acceso a las nuevas funcionalidades que despleguemos en el sistema, y también se puede crear cualquier número de roles personalizados a partir de la lista de unas doscientas capacidades del sistema. A cada capacidad se le puede asignar uno de estos cuatro niveles de acceso: No configurada, permitida, prohibida y prevenida. Cada usuario puede tener múltiples roles que heredan permisos de todos los niveles de contexto aplicables a una petición de acceso realizada por el usuario. La combinación de todo lo anterior proporciona una solución poderosa y flexible para los administradores.

Los roles estándar del sistema son:

- ✓ Administrador: El administrador del sistema tiene todos los permisos.
- ✓ Creador de cursos: Puede crear cursos en el sistema y puede estar limitado a una categoría de cursos.
- ✓ Profesor: Puede administrar un curso, y además puede desarrollar y actualizar su contenido.
- ✓ Profesor no editor: Puede administrar un curso, pero no puede modificar su estructura.
- ✓ Estudiante: Puede estar matriculado en un curso.
- ✓ Usuario autenticado: Todo usuario que haya iniciado sesión en el sistema tiene este rol.
- ✓ Invitado: Los usuarios no autenticados que tiene permiso de acceso al sistema.

Se pueden crear multitud de roles personalizados utilizando la interfaz de Moodle. Para comprobar y forzar un inicio de sesión dentro de nuestro código, se debe realizar una llamada a `require_login`. Esta función permite comprobar

si un usuario ha iniciado sesión, y forzar a que lo haga si lo requiere el elemento al que está tratando de acceder. Por ejemplo, podemos crear una actividad específica en un módulo que requiera que el usuario haya iniciado sesión en el sistema. Si no lo ha hecho, será redireccionado a la función de login adecuada.

### **Capacidades**

Las capacidades están asociadas con niveles de contexto y son reglas específicas de acceso que pueden ser concedidas a los roles. Ejemplos de capacidades son los siguientes:

- ✓ moodle/site:manageblocks. Puede administrar bloques en el nivel de contexto sitio.
- ✓ moodle/user:viewdetails. Puede ver los detalles de un usuario en el nivel de contexto usuario.
- ✓ moodle/course:view. Puede ver un curso en el nivel de contexto curso.

Cuando se desarrollen nuevas capacidades para el código personalizado, se debe considerar cuidadosamente en qué contexto estarán mejor localizadas. Las capacidades normalmente se colocan en el nivel de contexto más bajo en el que puedan funcionar.

### **Roles**

Los roles son identificadores específicos asociados con todos los contextos. Principalmente se usan para agrupar capacidades de un contexto, de forma que este grupo de capacidades puede ser asignado a los usuarios. Las capacidades se asignan a roles en contextos específicos bien por defecto o mediante una asignación específica.

### **Asignación de roles**

Por último, a los usuarios se les puede asignar roles en contextos específicos. Esta asignación les proporciona el acceso definido por las capacidades en ese rol para ese contexto.

Para resumir:

- ✓ Los contextos son elementos específicos en Moodle.
- ✓ Los roles están asociados con todos los contextos.
- ✓ Las capacidades se asignan a roles en un contexto dado.
- ✓ A los usuarios se les asignan roles en un contexto dado.

Son la asignación de roles a nivel de contexto y las capacidades que un rol tiene en ese contexto lo que determina si un usuario puede realizar la acción requerida.

Como desarrolladores de código para Moodle, necesitaremos tener esta funcionalidad en mente, para que de esta manera podamos diseñar y construir el control de acceso que deseamos para nuestras funciones.

## **A2. Guía para desarrolladores.**

Para facilitar la labor del desarrollador de Moodle inexperto, también se detallará la manera de llevar a cabo la instalación de la plataforma, preparando para ello el sistema con todos los requisitos necesarios.

Pasos:

1. Instalación del Moodle. (Instalación de Moodle y de sus requisitos.)

Los requerimientos de instalación de Moodle son los siguientes:

- ✓ Un servidor web. La mayoría de los usuarios utilizan Apache, pero Moodle debe funcionar bien en cualquier servidor web que soporte PHP, como el IIS (Internet Information Server) de las plataformas Windows.
- ✓ Una instalación de PHP en funcionamiento que este soportado a partir de Moodle utilizado.
- ✓ Una base de datos: MySQL o PostgreSQL, que están completamente soportadas y recomendadas para su uso con Moodle. MySQL es la elección preferida para mucha gente. Hay algunos argumentos a favor de PostgreSQL, especialmente si está planificando instalaciones de grandes dimensiones.

## 2. Estilo de código.

Todo el código nuevo definitivamente deberá adherirse a estos estándares de la forma más exacta posible. Para ello Moodle propone reglas generales a sus desarrolladores. Consultado en:

([http://docs.moodle.org/es/Manual\\_de\\_Estilo\\_de\\_Código#Reglas\\_Generales](http://docs.moodle.org/es/Manual_de_Estilo_de_Código#Reglas_Generales))

## 3. Estructura de la base de datos.

En cuanto a las tablas que nuestro módulo creará e insertará en la base de datos de Moodle, se debe tener en cuenta las siguientes consideraciones:

- ✓ Cada tabla debe tener un campo autonumérico id (INT10) como clave primaria.
- ✓ La tabla principal que contiene instancias de cada módulo debe tener el mismo nombre que el módulo y contener, por lo menos, los siguientes campos: id - clave primaria , course - el identificador del curso al que la instancia pertenece y name - el nombre completo de la instancia
- ✓ El resto de las tablas asociadas con un módulo que contiene información sobre cualquier otra cosa (p.ej. ,respuestas'), deberían ser llamadas modulo\_respuestas (en plural).
- ✓ Los nombres de las tablas y de los campos tienen que evitar el uso de palabras reservadas por las bases de datos. Esto se debe comprobar antes de crearlas.
- ✓ Los nombres de los campos (columnas) deberían ser sencillos y cortos, siguiendo las mismas reglas que los nombres de las variables.
- ✓ Cuando sea posible, las columnas que contengan una referencia al campo id de otra tabla (por ejemplo, modulo) debería ser llamado moduloid. (esta norma es nueva y no es seguida por algunas tablas antiguas).
- ✓ Los campos booleanos serán implementados como enteros cortos (por ejemplo, INT4) con los valores 0 o 1, para permitir la futura expansión de los valores si fuera necesario.

- ✓ La mayoría de las tablas tienen que tener un campo `timemodified` (INT10) que será actualizado con la fecha actual (timestamp de UNIX) obtenida con la función `time()` de PHP.
- ✓ Debemos definir siempre un valor por defecto para cada campo (y hacer que tenga sentido).
- ✓ Cada tabla debe comenzar con el prefijo de la base de datos (`$CFG->prefix`). En muchos casos esto es gestionado automáticamente. Además, bajo PostgreSQL, el nombre de cada índice debe empezar también con el prefijo.
- ✓ Para garantizar la compatibilidad entre bases de datos, se deben seguir las reglas siguientes sobre el uso del comando AS (solo si necesitamos alias en tablas/campos, por supuesto): No debemos utilizar el comando AS para alias de tablas. Utilizaremos el comando AS para alias de campos (columnas).
- ✓ Nunca debemos crear UNIQUE KEYs (restricciones) para nada. En su lugar utilizaremos UNIQUE INDEXes. En el futuro, si se decide añadir integridad referencial a Moodle y si se necesitan UNIQUE KEYs, serán utilizadas, pero no por ahora. El editor XMLDB permite especificar tanto restricciones UNIQUE como FOREIGN (y eso es bueno, teniendo el XML bien definido), pero solo los índices subyacentes serán realmente generados en la base de datos.
- ✓ El uso de UNIQUE KEYs creadas en el Editor XMLDB (punto anterior) solo debe ser definido si el campo/campos van a ser el objetivo para alguna FOREIGN KEY (a nivel de editor). En caso contrario, las crearemos como UNIQUE INDEXes.
- ✓ Nunca debemos realizar cambios a la base de datos en ramas estables. Si lo hacemos, los sitios actualizados de una versión estable a la siguiente pueden encontrarse con cambios por duplicado, lo cual puede producir errores serios.
- ✓ Cuando hagamos referencia a una variable entera en consultas SQL, no debemos entrecomillar el valor. Por ejemplo, `get_records_select('question', "category=$catid")` es correcto.

`get_records_select('question', "category='$catid'")` es incorrecto. Ese uso oculta posibles errores cuando `$catid` está sin definir.

#### 4. Normas de seguridad y de control de información.

A la hora de desarrollar nuestro propio módulo, también hay que tener en cuenta una serie de normas de seguridad, especialmente en lo que se refiere a la información obtenida del usuario por medio de formularios y de parámetros en la URL. Es muy importante mantener la seguridad en estos procedimientos para poder garantizar la integridad de los datos manejados por nuestro módulo. Algunas de las normas a seguir en este sentido son las siguientes:

- ✓ No debemos basar nuestro código en `register_globals`. Cada variable debe ser correctamente inicializada en cada fichero de código. Debe ser obvia la procedencia de cada variable.
- ✓ Debemos inicializar todos los arrays y objetos aunque estén vacíos. `$a = array()` o `$obj = new stdClass()`.
- ✓ No debemos utilizar la función `optional_variable()`. En su lugar, utilizaremos la función `optional_param()`.
- ✓ Seleccionaremos la opción `PARAM_XXXX` apropiada al tipo de parámetro que esperamos. Para comprobar y definir un valor opcional para una variable, utilizaremos la función `set_default()`.
- ✓ No utilizaremos la función `require_variable()`. En su lugar, utilizaremos la función `required_param()`.
- ✓ Seleccionaremos la opción `PARAM_XXXX` apropiada al tipo de parámetro que esperamos.
- ✓ Utilizaremos `data_submitted()`, con cuidado. La información todavía debe ser limpiada antes de utilizarla.
- ✓ No debemos utilizar `$_GET`, `$_POST` o `$_REQUEST`. En su lugar, utilizaremos las funciones `required_param()` u `optional_param()` apropiadas.
- ✓ No debemos comprobar las acciones con código como: `if (isset($_GET['algo']))`. Utilizaremos, por ejemplo, `$algo = optional_param( algo, -1, PARAM_INT )` y entonces comprobaremos

que está dentro de los valores esperados, por ejemplo, `if ($something>=0) {...`

- ✓ Cuando sea posible agruparemos todas las llamadas a `required_param()`, `optional_param()` y el resto de inicialización de variables en el principio de cada fichero (o función) para que sea fácilmente localizable.
- ✓ Utilizaremos el mecanismo `sesskey` para proteger el envío de formularios de ataques. Un ejemplo de uso: cuando el formulario es generado, incluiremos `<input type="hidden" name="sesskey" value="<?php echo sesskey(); ?>" />`. Cuando el formulario es procesado, comprobaremos `if (!confirm_sesskey()) {error('Bad Session Key');}`.
- ✓ Todos los nombres de ficheros deben ser limpiados de caracteres extraños utilizando la función `clean_filename()`, si esto no ha sido realizado con el uso de las funciones `required_param()` y `optional_param()` con anterioridad.
- ✓ Cualquier información leída desde la base de datos debe tener la función `addslashes()` aplicada antes de volver a enviar la información a la base de datos. Un objeto completo puede ser procesado con la función `addslashes_object()`.
- ✓ No debemos utilizar información obtenida de `$_SERVER` si podemos evitarlo. Presenta algunos problemas de portabilidad.
- ✓ Si no ha sido realizado en ningún otro lugar, debemos asegurarnos de que la información enviada a la base de datos ha sido filtrada mediante la función `clean_param()` utilizando la opción `PARAM_XXXX` apropiada.
- ✓ Si escribimos código SQL, nos aseguraremos completamente de que es correcto. En particular, comprobaremos la falta de comillas en las variables utilizadas. Es un punto de entrada para ataques de tipo SQL injection.
- ✓ Comprobaremos toda la información (especialmente la que es enviada a la base de datos) en cada archivo que es utilizada. Nunca debemos confiar en que otro código estará haciendo ese trabajo.

- ✓ Los bloques de código que se incluyan deben presentar una estructura PHP correcta (por ejemplo, con una declaración de una clase, de funciones, etc.) Los bloques de código lineales suelen tender a utilizar variables sin inicializar (y son menos legibles).
- ✓ Si necesitamos usar `shell_exec()` (o cualquier otra función que invoque un shell), nos aseguraremos de que se han limpiado los parámetros anteriormente con `escapeshellcmd()/escapeshellarg()` (de lo contrario abrimos la puerta a ataques de inyección de shell).

## 5. Estilo de interfaz.

En este apartado se ofrece una colección de ideas para adaptar la interfaz de nuestro módulo al estilo global de Moodle, de la mejor manera posible:

- ✓ Simplicidad: Debemos usar el mínimo de interfaz necesario para obtener la funcionalidad que necesitamos.
- ✓ Páginas estándar: Tanto en módulos de actividad como en bloques, existen una serie de ficheros que se deben crear obligatoriamente. Debe haber un script por función o página principal.
- ✓ Plantilla de página: Imprimiremos las cabeceras con `print_heading()`, usaremos el CSS para IDs y Clases. Imprimiremos las cajas alrededor del texto usando `print_simple_box()`.
- ✓ Plantilla de formulario: Mostraremos las opciones más importantes en la parte superior. Cada entrada debe tener una etiqueta y, si es necesario, un archivo de ayuda. Si hay más de 10 opciones, las desglosaremos en los parámetros necesarios y opcionales, extra o avanzados.
- ✓ Manejo de tablas: Usaremos la función `print_table()` cuando sea posible.
- ✓ Herramientas de navegación estándar: Todas las páginas deberían llamar a `print_header()`, y suministrar una ruta de navegación estándar que apareciera allí. Donde sea posible, debería verse como: COURSE >> INDEX >> INSTANCE >> SUBPAGES... Las páginas incluidas en módulos de actividad deberían llamar a `navmenu()` para generar el menú de navegación apropiado.

- ✓ Direcciones URL: Las direcciones URL deben ser lo más cortas posible. No debemos usar subrayado en nombres de parámetros o nombres de archivos. Nunca debemos usar dos palabras cuando una sea suficiente.
  - ✓ Botones o enlaces: El Web Accelerator de Google proporciona algunas sugerencias: Las acciones que puedan modificar el estado de Moodle (archivos de datos, base de datos, información de sesión) deben ser llevadas a cabo por medio de botones. Como mínimo, tales acciones, que son implementadas como enlaces, deben remitir a una página de confirmación que sí utilice botones.
  - ✓ Enlaces a archivos de ayuda: Los botones de ayuda deben estar a la derecha del objeto (excepcionalmente pueden estar a la izquierda, si el objeto está alineado a la derecha).
6. Librerías usadas en la implementación del módulo con algunas de sus principales funciones comentadas.

**Weblib. php:** En esta librería se encuentran las funciones para la creación de elementos HTML y salida web en general. Es considerada como una de las principales librerías del núcleo de Moodle, de manera que es incluida en el fichero setup.php. Sus funciones más importantes son:

- ✓ ***print\_heading()***, como sustituto del conocido <h1>.
- ✓ ***print\_heading\_with\_help()***, idem, pero proporcionando un botón de ayuda.
- ✓ ***helpbutton()*** imprime un botón de ayuda.
- ✓ ***print\_simple\_box()*** imprime una caja en la que se puede colocar cualquier tipo de contenido.
- ✓ ***choose\_from\_menu()*** imprime un menú de selección.
- ✓ ***choose\_from\_radio()*** imprime botones de radio.
- ✓ ***print\_textbox()*** imprime una caja de texto.
- ✓ ***print\_textfield()*** imprime un campo de texto.

El contenido de esta librería es muy extenso y resultará de gran ayuda al nuevo desarrollador. Se aconseja leerla detenidamente, prestando especial atención a la forma en la que Moodle facilita la creación de controles de formularios así

como a las funciones que se encargan de recoger las variables de los mismos (*required\_param()* y *data\_submitted()* especialmente).

**accesslib.php:** esta librería describe funciones para el soporte de capacidades, definiendo las capacidades de los usuarios en función a los roles para este módulo, en este caso al profesor del curso y al administrador. En el fichero View.php se define el contenido que se mostrará a cada usuario de la actividad mediante el uso de capacidades o permisos. Contiene la ***get\_context\_instance*** : función que devuelve el contexto de una instancia como un objeto.

**datalib.php** : en esta librería se encuentran las funciones para el manejo de la información de la base de datos. Con un funcionamiento similar a la función *get\_users()* *datalib.php* implementa funciones para obtener los cursos del sitio o de un determinado usuario, los módulos de un determinado curso, los usuarios que han sido confirmados en el sistema, los distintos grupos establecidos, los usuarios que conforman cada grupo y más información similar, todas ellas documentadas en el archivo *datalib.php*.

Establece los Parámetros:

- ✓ ***\$get:*** Si se establece a *false*, solo retornará una cuenta del número de registros coincidentes.
- ✓ ***\$search:*** Una determinada cadena a buscar.
- ✓ ***\$confirmed:*** Si se establece a *true*, solo devolverá los usuarios que hayan sido confirmados en el curso.
- ✓ ***\$exceptions:*** Una lista en forma de array conteniendo ciertos identificadores de usuario a ignorar.
- ✓ ***\$sort:*** Fragmento SQL para la ordenación de los registros a devolver.
- ✓ ***\$firstinitial:*** Inicial del primer nombre de los usuarios a buscar.
- ✓ ***\$lastinicial:*** Inicial del apellido de los usuarios a buscar.
- ✓ ***\$recordsperpage:*** Número de registros a obtener en cada llamada, para evitar saturación.
- ✓ ***\$fields:*** Campos de los registros a obtener (igual que el contenido de una sentencia SELECT).

**ddl.lib.php:** en esta librería se pueden encontrar las funciones para el manejo de la estructura de la base de datos. Proporciona funcionalidades avanzadas para hacer la manipulación de la estructura de la base de datos lo más portable posible. Dichas estructuras deben modificarse usando funciones de ésta biblioteca y nunca con sentencias propias de determinada base de datos.

Se proporcionan funciones como las siguientes, cuyo nombre, en ocasiones, recuerda a las sentencias SQL habituales:

- ✓ **create\_table()** para crear una tabla.
- ✓ **drop\_table()** para borrar el contenido de una tabla.
- ✓ **rename\_table()** para cambiar el nombre a una tabla.
- ✓ **add\_field()** para añadir un campo a una tabla.
- ✓ **change\_field\_tipo()** para cambiar el tipo de un campo.
- ✓ **add\_key()** para añadir una clave.
- ✓ **add\_index()** para añadir un índice.

**dml.lib.php:** en esta librería se encuentran las funciones necesarias para obtener, borrar, actualizar, contar e insertar registros en la base de datos. Las funciones que se describirán a continuación serán las que el desarrollador use en un 99% de los casos. El fichero Versión.php resulta importante a la hora de modificar las tablas de la base de datos.

Funciones generales:

- ✓ **execute\_sql** : ejecuta una sentencia SQL e imprime el resultado si se especifica.
- ✓ **record\_exists** : devolverá *true* si existe en la tabla *\$table* al menos un registro cuyos valores en los campos *\$fieldx* se correspondan con *\$valuex*.
- ✓ **record\_exists\_sql** : devolverá *true* si la sentencia SQL pasada como parámetro retorna algún resultado.

- ✓ `count_records` : devolverá el número de registros que haya en la tabla `$table` cuyos valores en los campos `$fieldx` se correspondan con `$valuex`.
- ✓ `count_records_select` : devolverá el número de registros que existen en la tabla `$table` cuyos valores se correspondan con el fragmento `WHERE` pasado como parámetro.
- ✓ `insert_record` : inserta un registro pasado como objeto, cuyos nombres de atributos se corresponderán con los campos del registro, en la tabla pasada como parámetro.
- ✓ `update_record` : actualiza un registro en la tabla cuyo nombre es pasado como parámetro.

**Lib.php:** El fichero es la biblioteca de funciones del módulo a desarrollar, en ella se implementará toda (o casi toda) la funcionalidad del módulo. Contiene ficheros usados tales como:

- ✓ `Mod_form.php`: Mediante el fichero se configuró cada instancia del módulo.
- ✓ `Upgrade.php`: Este fichero mantiene un registro de los cambios que es necesario realizar entre distintas versiones del módulo.

Será necesario profundizar en otras librerías y las funciones que incluye según la funcionalidad que se quiera desarrollar.

### A3

<b>Caso de Uso</b>	Gestionar nivel de asimilación por pregunta.
<b>Actores</b>	Profesor del curso.
<b>Resumen</b>	El caso de uso se inicia cuando el Profesor de la UCF una vez logueado en el Moodle entra en su curso y desea insertar, modificar o eliminar nivel de asimilación a una pregunta. El caso de uso finaliza cuando se haya insertado, modificado o eliminado el nivel de asimilación a cada pregunta y actualizado los datos.
<b>Responsabilidades</b>	Insertar, eliminar o modificar un nivel de asimilación para cada pregunta.
<b>CU asociados</b>	No hay
<b>Precondiciones</b>	Inicialmente el profesor del curso deberá autenticarse con éxito en el Moodle. El profesor

	del curso debe haber insertado preguntas anteriormente mediante la interfaz del Moodle.
<b>Descripción del flujo normal de eventos</b>	
<p><b>Acción del Actor (es)</b></p> <p>1- El profesor entra a su curso, elige en Agregar Actividad, el módulo Genex.</p> <p>2- El profesor del curso selecciona Gestionar Nivel de asimilación.</p> <p>3- El profesor selecciona una pregunta de las listadas. Luego presiona la opción Ver</p> <p>4- El profesor termina el trabajo con la pregunta.</p>	<p><b>Respuesta del Sistema</b></p> <p>1.1- El sistema muestra la interfaz principal.</p> <p>2.1- El sistema muestra la interfaz Listar Preguntas.</p> <p>3.1- El sistema muestra la interfaz Pregunta.</p> <p>3.2- El sistema ejecuta alguna de las siguientes acciones:</p> <p>a- Si el profesor del curso decide insertar un nivel de asimilación, ir a la sección “Insertar”.</p> <p>b- Si el profesor del curso decide modificar el nivel de asimilación, ir a la sección “Modificar”.</p> <p>c- Si el profesor del curso decide eliminar el nivel de asimilación, ir a la sección “Eliminar”.</p> <p>4.1- El sistema finaliza el caso de uso.</p>
<b>Sección “Insertar”</b>	
<p><b>Acción del actor</b></p> <p>1- El profesor del curso presiona la opción insertar nivel de asimilación.</p> <p>2- El profesor selecciona y presiona el botón insertar.</p>	<p><b>Respuesta del sistema</b></p> <p>1.1- El sistema le muestra una interfaz de seleccionar nivel de asimilación.</p> <p>2.1- El sistema verifica que los datos estén correctos.</p> <p>2.2- El sistema inserta el nivel de asimilación para la pregunta seleccionada en la base de datos y muestra un mensaje de éxito.</p>

<p><b>Cursos Alternos</b></p> <p>En la línea 2.1 si al sistema validar los datos introducidos por el profesor encuentra errores o campos vacíos, le hace un señalamiento en los campos haciéndoselo saber. El profesor vuelve a introducir los datos y continúa con la acción2.</p>	
<p><b>Sección “Eliminar”</b></p>	
<p><b>Acción del actor</b></p> <p>1- El profesor presiona la opción eliminar nivel de asimilación.</p> <p>2- El profesor presiona el botón aceptar y finaliza el caso de uso.</p>	<p><b>Respuesta del sistema</b></p> <p>1.1 El sistema elimina el nivel de asimilación y muestra un mensaje de éxito.</p>
<p><b>Cursos Alternos</b></p> <p>En la línea 1.1 si el sistema no puede eliminar el nivel muestra un mensaje al profesor.</p>	
<p><b>Sección “Modificar”</b></p>	
<p><b>Descripción del flujo normal de eventos</b></p>	
<p><b>Acción del actor</b></p> <p>1- El profesor presiona la opción modificar nivel de asimilación.</p> <p>2- El profesor introduce los datos y presiona el botón insertar.</p>	<p><b>Respuesta del sistema</b></p> <p>1.2- El sistema le muestra una interfaz para seleccionar datos del nivel de asimilación.</p> <p>2.1- El sistema verifica que los datos estén correctos.</p> <p>2.2- El sistema modifica el nivel de asimilación para la pregunta seleccionada en la base de datos y muestra un mensaje de éxito.</p>
<p><b>Cursos alternos</b></p>	<p>En la línea 2.1 si al sistema validar los datos introducidos por el profesor encuentra errores o campos vacíos, le hace un señalamiento en los campos haciéndoselo saber. El profesor vuelve a introducir los datos y continúa con la acción2.</p>
<p><b>Requisitos funcionales</b></p>	<p>R1,R2,R3,R4</p>
<p><b>Postcondiciones</b></p>	<p>Queda actualizada la base de datos del sistema.</p>

A4

<b>Caso de Uso</b>	Generar Examen.
<b>Actores</b>	Profesor del curso.
<b>Resumen</b>	El caso de uso se inicia cuando el Profesor de la UCF una vez logueado en el Moodle entra en su curso y desea generar un examen. Para ello selecciona el módulo Genex. El caso de uso finaliza cuando se haya generado el examen y actualizado los datos en el curso.
<b>Responsabilidades</b>	Generar examen.
<b>CU asociados</b>	No hay
<b>Precondiciones</b>	Inicialmente el profesor del curso deberá autenticarse con éxito en el Moodle y en su curso.
<b>Descripción del flujo normal de eventos</b>	
<b>Acción del Actor (es)</b> 1- El profesor entra a su curso, elige en Agregar Actividad, el módulo Genex.  2- El profesor del curso selecciona Generar Examen automático.  3- El profesor edita datos de examen, eligiendo el número de preguntas y asigna para cada una de ellas el nivel de asimilación por categoría. Luego presiona la opción Aceptar	<b>Respuesta del Sistema</b> 1.1- El sistema muestra la interfaz principal.  2.1- El sistema muestra la interfaz de editar.  3.1- El sistema genera el examen agregando preguntas aleatoriamente basado en la selección anterior. 3.2-El sistema muestra el examen en una nueva interfaz.
<b>Cursos alternos</b>	En la línea 3.1 si el sistema encuentra que el examen ya fue generado muestra un mensaje al profesor del curso. El profesor vuelve a insertar datos y continúa con la acción 3.1.  En la línea 3.1 si el sistema no encuentra preguntas que asignar a la categoría y el nivel de asimilación seleccionado muestra un mensaje al profesor indicando la pregunta del error. El profesor selecciona datos nuevamente y continúa la acción 3.1.
<b>Requisitos funcionales</b>	R5
<b>Postcondiciones</b>	Queda actualizada la base de datos del sistema con un nuevo examen.

**A5**

<b>Caso de Uso</b>	Exportar Examen.
<b>Actores</b>	Profesor del curso.
<b>Resumen</b>	El caso de uso se inicia cuando el Profesor de la UCF una vez logueado en el Moodle entra en su curso y desea generar un examen. Para ello selecciona el módulo Genex. El caso de uso finaliza cuando se haya exportado el examen a PDF y actualizado los datos en el curso.
<b>Responsabilidades</b>	Generar e imprimir un examen.
<b>CU asociados</b>	No hay
<b>Precondiciones</b>	Inicialmente el profesor del curso deberá autenticarse con éxito en el Moodle y en su curso.
<b>Descripción del flujo normal de eventos</b>	
<b>Acción del Actor (es)</b> 1- El profesor entra a su curso, elige en Agregar Actividad, el módulo Genex.  2- El profesor del curso selecciona Exportar Examen.  3- El profesor selecciona el o los exámenes y la cantidad de copias que desea. Luego presiona la opción Exportar.	<b>Respuesta del Sistema</b> 1.1- El sistema muestra la interfaz principal.  2.1- El sistema muestra la interfaz de Exportar Examen  3.1- El sistema muestra un mensaje de éxito
<b>Cursos alternos</b>	En la línea 2.1 si el sistema no encuentra exámenes generados muestra un mensaje al profesor del curso.
<b>Requisitos funcionales</b>	R6
<b>Postcondiciones</b>	Queda un registro de archivos exportados.

## A6

### Agregando pregunta verdadero/falso

[▶ Expandir todo](#)

▼ General

Categoría

Nombre de la pregunta\*

Niveles de Asimilación

Enunciado de la pregunta\*

*I*       

Ilustración 12. Prototipo de Interfaz de Asignar nivel de asimilación a pregunta

## A7

### Pregunta aleatoria desde una categoría existente

Categoría

Incluir también preguntas de subcategorías

Número de preguntas al azar

Niveles de Asimilación

### Pregunta aleatoria utilizando una categoría nueva

Nombre

Ilustración 13. Prototipo de Interfaz de Generar examen



A10

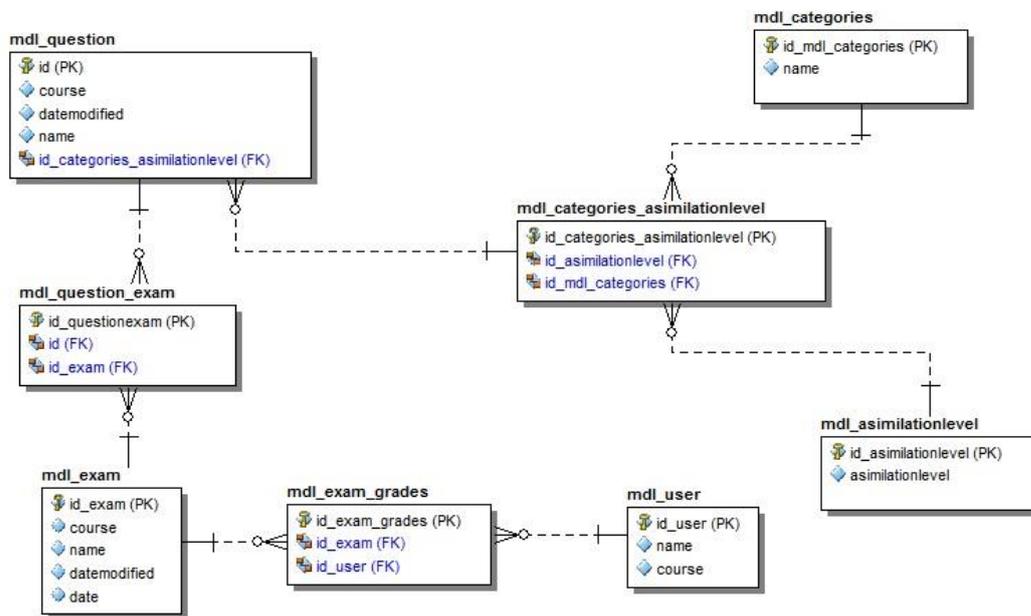


Ilustración 16. Modelo físico de datos.