



UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS
FACULTAD 2 “TELECOMUNICACIONES Y SEGURIDAD INFORMÁTICA”

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

Título: Módulo de Actividad “Evaluación” para la plataforma MOODLE.

Autores:

Hamlet Soto García.

Alexander Raventos Liranza.

Tutor:

Lic. Yadilka Suárez-Inclán Rivero.

Ciudad de La Habana, mayo de 2010.

“Año 52 de la Revolución”.

Declaración de Autoría

Declaramos que _____ y _____ somos los únicos autores de este trabajo y autorizamos a la Universidad de las Ciencias Informáticas (UCI) y a la Facultad (2) para que hagan el uso que estimen pertinente con este trabajo.

Para que así conste firmamos la presente a los ____ días del mes de junio de 2010.

Firma del Autor
Hamlet Soto Garcia

Firma del Autor
Alexander Raventos Liranza

Firma del Tutor
Lic. Yadilka Suárez-Inclán Rivero.

“Cuando se es joven, se crea. Cuando se es inteligente, se produce. No se adapta, se innova: la medianía copia; la originalidad se atreve.”

José Martí.

AGRADECIMIENTOS

A mi familia y todo aquel que se considera parte de ella, en especial a mis padres, a mi abuela, a mis tíos José, Javier, Julito, por todo el apoyo y la confianza depositada en mí.

A mi abuela Paulina, por tanto cariño, por tanta paciencia, por tanto tesón, por tanta vida.

A mi madre Olivia, por el amor, la preocupación y el sacrificio.

A mi padre Alfredo, por ser mi ejemplo a seguir, el camino por donde voy.

A mis dos hermanas Amandita y Nana, por hacerme ver como un ejemplo para las dos.

A mis primos, Dine, Ille y sus niñas, Ale, Adriana, Javielito y Diane, por tanto cariño y alegrías.

Al Moro por tanta entrega y confianza.

A mi tía Enid Julia, Alina, a María.

A mis amigos y amigas, por la ayuda y la amistad que tengo con ellos.

A mi compañero de tesis y a su familia.

A los profesores que me formaron durante estos 5 años.

A aquellos que me dieron la espalda, por mostrarme quienes son los que realmente valen la pena.

A los que ya no están.

A mi Cuba.

Hamlet.

A mi familia, mi madre, mi tía Pilar por su apoyo y su amor.

A mis abuelos Magalys y Juan, a mi tío Oscar.

A Maruchi, Emilio, Carlo y Macary por ser mi familia aquí en La Habana.

A mis hermanos de corazón Alberto, Javier, René, Carlo Carlo, Hamlet y Otto.

A mis hermanas Blanca, Elaine, Bertha y Giselle.

A todos mis compañeros.

A todos los profesores que de una forma u otra me ayudaron en mi formación profesional.

A mi tutora y a Yadira por su ayuda incondicional.

A todos mis vecinos por su confianza y apoyo.

A todas las personas cuyos nombres no están aquí pero que los llevo en el corazón.

A mi país y a mi Revolución.

Alexander.

DEDICATORIA

Dedico este trabajo de diploma a toda mi familia, a mis padres, a mis tíos, a mis primos, en especial a una de las personas más importante para mí, a mi abuela Paulina.

Hamlet.

Le dedico este trabajo de diploma a mi familia, a mis abuelos y mi tío Oscar, a mis amigos y amigas y muy especial a mi madre y mi tía Pilar.

Alexander.

RESUMEN

Con el desarrollo de las Tecnologías de la Información y las Comunicaciones y su importante apoyo a la Educación, se introduce el término de e-learning. Surgen los Entornos Virtuales de Aprendizaje soportados por plataformas de e-learning. El surgimiento de estas plataformas fue muy diverso, y se produjo en muchos lugares del mundo casi simultáneamente.

MOODLE constituye una plataforma con un alto grado de utilización en el mundo del e-learning, sobre la cual está basado el Entorno Virtual de Aprendizaje que se utiliza en la Universidad de las Ciencias Informáticas. Esta permite la creación de actividades que permiten la interacción profesor-estudiante y estudiante-estudiante durante todo el proceso de enseñanza aprendizaje.

En este trabajo se realiza un análisis profundo del funcionamiento interno de la plataforma, priorizando la gestión de módulos, el manejo de idiomas y el sistema de roles y permisos, y se propone un módulo que se integre a MOODLE y permita crear una actividad donde el estudiante pueda auto-evaluarse y/o evaluar a sus compañeros.

Palabras Claves: e-learning, MOODLE, plataforma, módulo, actividad.

Índice

INTRODUCCIÓN	1
CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA	4
INTRODUCCIÓN.	4
1.1 ESTADO DEL ARTE.....	4
1.2 ¿QUÉ ES EL E-LEARNING?.....	4
1.3 SISTEMA DE ADMINISTRACIÓN DE APRENDIZAJE (LMS).....	6
1.4 SISTEMA DE ADMINISTRACIÓN DE CONTENIDOS DE APRENDIZAJE (LCMS).....	6
1.5 MOODLE	7
1.5.1 Entornos virtuales de aprendizaje.....	7
1.5.2 Significado de MOODLE y sus orígenes.....	8
1.5.3 MOODLE y Software libre.....	8
1.5.4 Resumen de características de MOODLE.....	9
1.5.5 Estadísticas sobre MOODLE.....	9
1.5.6 Módulos de MOODLE.....	14
1.6 PLATAFORMAS UTILIZADAS EN EL MUNDO.....	17
1.6.1 Atutor.....	18
1.6.2 Decebo LMS.....	18
1.6.3 Proyecto MERLIN.....	18
1.7 PLATAFORMAS UTILIZADAS EN CUBA.....	19
1.7.1 SEPAD.....	19
1.7.2 aprendDIST.....	20
1.7.3 Medcampus.....	20
1.8 INSTITUCIONES Y UNIVERSIDADES QUE UTILIZAN MOODLE EN CUBA.....	21
1.9 HERRAMIENTAS Y METODOLOGÍA.....	21
1.9.1 Metodología de desarrollo.....	21
1.9.2 Lenguaje de Modelado.....	22
1.9.3 Herramienta CASE.....	23
1.9.4 Herramienta para el almacenamiento de datos.....	23
1.9.5 Herramienta de Desarrollo (IDE).....	25
1.9.6 Servidor Web.....	25
1.9.7 Lenguajes de programación.....	25
1.10 DISEÑO METODOLÓGICO DE LA INVESTIGACIÓN.....	27
1.10.1 Métodos Teóricos utilizados.....	27
1.10.2 Métodos Empíricos utilizados.	28
CONCLUSIONES.....	29
CAPÍTULO 2. CARACTERÍSTICAS DEL SISTEMA	30
INTRODUCCIÓN.....	30
2.1 OBJETO DE ESTUDIO.....	30
2.2 PROPUESTA DEL SISTEMA.....	30
2.3 MODELO DE DOMINIO.....	31
2.3.1 Descripción del Problema de Dominio.....	31

2.3.2 Representación del modelo del dominio.....	32
2.4 CAPTURA DE REQUISITOS.....	32
2.4.1 Especificación de los requisitos de software.....	32
2.4.1.2 Requerimientos No Funcionales.....	34
2.5 DEFINICIÓN DE LOS ACTORES DEL SISTEMA.....	35
2.6 DIAGRAMA DE CASOS DE USO DEL SISTEMA.....	36
2.7 ESQUEMA DE UN MÓDULO.....	36
2.8 ESTILO DE CÓDIGO DE MOODLE.....	38
CONCLUSIONES.....	39
CAPÍTULO 3. DISEÑO DEL SISTEMA.....	40
INTRODUCCIÓN.....	40
3.1 ARQUITECTURA.....	40
3.2 PATRÓN DE DISEÑO.....	40
3.3 PATRÓN MODELO –VISTA-CONTROLADOR (MVC).....	41
3.4 MODELO DE DISEÑO.....	42
3.5 DISEÑO.....	42
3.5.1 Diagramas de Interacción.....	42
3.5.2 Diagramas del Diseño.....	42
3.5.3 Modelo de Datos.....	42
CONCLUSIONES.....	43
CAPÍTULO 4. IMPLEMENTACIÓN Y PRUEBA.....	44
INTRODUCCIÓN.....	44
4.1 DIAGRAMA DE DESPLIEGUE.....	44
4.2 PRUEBA.....	44
4.2.1 Casos de Prueba.....	44
4.3 PASOS PARA INSTALAR EL MÓDULO DE ACTIVIDAD “EVALUACIÓN”.....	53
CONCLUSIONES.....	54
CAPÍTULO 5. FACTIBILIDAD DEL SISTEMA.....	55
INTRODUCCIÓN.....	55
5.1 MÉTODO DE ESTIMACIÓN POR PUNTOS DE CASOS DE USO.....	55
5.1.1 Cálculo de Puntos de Casos de Uso sin ajustar.....	55
5.1.2 Cálculo de Puntos de Casos de Uso ajustados.....	58
5.1.3 Distribución del Esfuerzo entre las diferentes actividades del módulo.....	63
5.2 BENEFICIOS TANGIBLES E INTANGIBLES.....	63
5.3 ANÁLISIS DE COSTOS Y BENEFICIOS.....	63
CONCLUSIONES.....	64
CONCLUSIONES GENERALES.....	65
RECOMENDACIONES.....	66
BIBLIOGRAFÍA.....	67
ANEXOS.....	71
ANEXO # 1. DESCRIPCIÓN TEXTUAL DE LOS CASOS DE USO DEL SISTEMA.....	71

ÍNDICE.

ANEXO # 2. DIAGRAMAS DE INTERACCIÓN.	89
ANEXO # 3 DIAGRAMAS DE CLASES DEL DISEÑO.	92
ANEXO # 4 DIAGRAMAS DE COMPONENTES.	97
GLOSARIO DE TÉRMINOS.....	99

INTRODUCCIÓN

En la actualidad el desarrollo de las Tecnologías de la Informática y las Comunicaciones (en lo adelante TIC) ha tenido un gran auge en todas las esferas de la vida, tanto en lo económico, político y social. En nuestro país el uso de las TIC ha tomado un papel representativo en el plano educativo de nuestra sociedad y en todos los niveles de nuestro sistema de educación. En la Universidad de las Ciencias Informáticas (en lo adelante UCI) las TIC han desempeñado un papel importantísimo en el Proceso de Enseñanza Aprendizaje (PEA) y debe continuar desarrollándolo cada vez más, mediante el uso creativo y eficiente que se realice de estas en todos los aspectos.

Nuestra Universidad se encuentra involucrada, en esta era de las tecnologías, en un proceso de cambio en la integración de sus elementos fundamentales, la Formación, Producción e Investigación, con el objetivo de alinearlos estratégicamente en función de la Misión/Visión del centro para responder de una manera más eficiente al encargo y responsabilidad social que le han sido asignados. En este sentido, se ha propiciado un cambio de paradigma dentro del proceso de enseñanza-aprendizaje que apuesta por el desarrollo de un modelo de formación centrado en el aprendizaje, con una incorporación progresiva a la enseñanza semi-presencial en entornos virtuales de aprendizaje.

Esta concepción tiene un impacto importante en la utilización de las TIC asociadas al proceso de formación y dentro de él se está provocando un cambio en torno a la evaluación del aprendizaje, con la necesidad de garantizar una evaluación formativa en los estudiantes, donde se utilice verdaderamente la evaluación como estrategia para mejorar y favorecer los aprendizajes, en lugar de un simple indicador de éxito o fracaso en los mismos. Para esto el estudiante tiene que ser protagonista también de su evaluación, principalmente a través de la Auto-evaluación y de la evaluación de sus compañeros (Co-evaluación) y el profesor debe recolectar y analizar todas las evidencias posibles que le permitan valorar el avance del estudiante en el proceso de aprendizaje.

En la UCI se emplea como Entorno Virtual de Aprendizaje (en lo adelante EVA) la plataforma cuyo nombre es Entorno de Aprendizaje Dinámico Modular Orientado a Objetos, en inglés Modular Object-Oriented Dynamic Learning Environment (en lo adelante MOODLE) que se puede acceder a través de la dirección

<http://eva.uci.cu>. Esta plataforma brinda al profesor y al estudiante varias actividades y opciones para interactuar durante el PEA.

Sin embargo, a pesar de contar con actividades como el taller, que permite realizar por parte de los estudiantes una auto-calificación y la calificación de una tarea de sus compañeros, MOODLE no presenta actividades que permitan la Auto-evaluación y la Co-evaluación de los compañeros como se ha concebido en el modelo de formación centrado en el aprendizaje en la UCI.

Por lo antes descrito se determina como **Situación Problemática** que el entorno virtual de aprendizaje sustentado en la UCI por la plataforma MOODLE no contempla actualmente todos los elementos necesarios para contribuir a que el profesor y el estudiante participen en el desarrollo de una evaluación formativa.

Después de analizar la situación expuesta, se define como Problema a Resolver: ¿Cómo lograr que el entorno virtual de aprendizaje utilizado en la UCI contribuya a desarrollar una evaluación formativa en el PEA?

A partir del problema científico se puede inferir que el **Objeto de estudio** lo constituyen los procesos y funcionalidades del EVA utilizado en la UCI y como **Campo de Acción** los procesos que se llevan a cabo para el desarrollo de un nuevo módulo.

El presente trabajo tiene como **Objetivo General**

- Incorporar a la plataforma MOODLE un nuevo módulo que permita al estudiante desarrollar una Auto-evaluación y la evaluación de sus compañeros, para contribuir a realizar una evaluación formativa.

Y como **Objetivos Específicos:**

- ✓ Desarrollar un módulo que permita:
 - Gestionar una Actividad de Auto-evaluación y/o Co-evaluación.
 - Permitir a los estudiantes realizar una Actividad de Auto-evaluación y/o Co-evaluación.
- ✓ Integrar el nuevo módulo a la plataforma MOODLE

Para dar cumplimiento a estos objetivos se han trazado las siguientes **Tareas de la Investigación**:

- Realizar un análisis del módulo Actividad de la plataforma MOODLE.
- Diseñar una nueva actividad que permita la Auto-evaluación y Co-evaluación.
- Implementar la actividad de Auto-evaluación y Co-evaluación.

Se propone como **Idea a Defender** que el desarrollo del Módulo de Actividad “Evaluación” para MOODLE logra propiciar una evaluación formativa más completa.

El presente documento, está estructurado en cinco capítulos, a continuación se muestra una breve descripción de cada uno de ellos:

Capítulo No1 “Fundamentación teórica”. En este capítulo se realiza un estudio del Estado del Arte de varias plataformas existentes, así como una fundamentación de las herramientas, metodologías y lenguajes utilizados para desarrollar el módulo en cuestión.

Capítulo No2: “Características del Sistema”. En este capítulo se describen las características y los procesos que contienen toda la información relacionada con el sistema propuesto.

Capítulo No3: “Diseño del sistema”. En este capítulo se realiza el diseño del sistema, teniendo en cuenta las necesidades del usuario.

Capítulo No4: “Implementación y Prueba”. En este capítulo se muestran los diagramas generados en la Fase de Implementación y los Casos de Prueba realizados (caja negra).

Capítulo No5: “Estimación y Factibilidad”. En este capítulo se lleva a cabo la estimación basada en el método de estimación por Puntos de Casos de Usos.

CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

Introducción.

En este capítulo se realiza una profunda investigación y se exponen los temas relacionados con el objeto de estudio para una mejor comprensión de este. La utilización y desarrollo de plataformas y entornos virtuales de aprendizaje en el mundo, en Cuba, y en la Universidad de las Ciencias Informáticas.

Así como una fundamentación de las herramientas, metodologías y lenguajes utilizados para dar solución a la problemática anteriormente planteada.

1.1 Estado del Arte.

En la actualidad no se concibe la educación separada de las TIC, a nivel mundial se han desarrollado metodologías y nuevos paradigmas que fundamentan las maneras de interrelacionarlas. Lo cual deviene en un nuevo modelo pedagógico-tecnológico que asegura una educación acertada, cuyo mayor reto es mantener y elevar la calidad del proceso docente-educativo y donde la enseñanza está siendo reforzada con métodos más flexibles, en los que no es imprescindible el contacto físico profesor-alumno, en ocasiones incluso, como en la enseñanza a distancia, no es necesario este contacto, salvo algunas reuniones periódicas indispensables. Para el desarrollo de estos métodos de enseñanza es fundamental la utilización correcta y creativa de las TIC.

1.2 ¿Qué es el e-learning?

El e-learning consiste en la educación y capacitación a través de Internet. Este tipo de enseñanza online permite la interacción del usuario con el material mediante la utilización de diversas herramientas informáticas.

Este nuevo concepto educativo es una revolucionaria modalidad de capacitación que posibilitó Internet, y que hoy se posiciona como la forma de capacitación predominante en el futuro. Este sistema ha transformado la educación, abriendo puertas al aprendizaje individual y organizacional. Es por ello que hoy en día está ocupando un lugar cada vez más destacado y reconocido dentro de las organizaciones empresariales y educativas.

El término "e-learning" es la simplificación de Electronic Learning. El mismo reúne a las diferentes tecnologías, y a los aspectos pedagógicos de la enseñanza y el aprendizaje.

Los beneficios del e-learning son:

- ✓ Reducción de costos: Permite reducir y hasta eliminar gastos de traslado, alojamiento, material didáctico, etc.
- ✓ Rapidez y agilidad: Las comunicaciones a través de sistemas en la red confieren rapidez y agilidad a las comunicaciones.
- ✓ Acceso just-in-time: Los usuarios pueden acceder al contenido desde cualquier conexión a Internet, cuando les surge la necesidad.
- ✓ Flexibilidad de la agenda: No se requiere que un grupo de personas coincidan en tiempo y espacio. (e-ABC)

También abordamos otras definiciones comunes tales como:

- ✓ Técnicamente, el e-Learning es la entrega de material educativo vía cualquier medio electrónico, incluyendo el Internet, Intranet, Extranet, audio, vídeo, red satelital, televisión interactiva, CD y DVD, entre otros medios.
- ✓ Para los educadores, e-Learning es el uso de tecnologías de redes y comunicaciones para diseñar, seleccionar, administrar, entregar y extender la educación.
- ✓ Para los elocuentes, el e-Learning es el empleo del poder de la red mundial para proporcionar educación, en cualquier momento, en cualquier lugar.
- ✓ Siendo descriptivos, la educación electrónica es la capacitación y adiestramiento de estudiantes y empleados usando materiales disponibles para Web a través del Internet, llegando a ofrecer sofisticadas facilidades como flujo de audio y vídeo, presentaciones en

PowerPoint, vínculos a información relativa al tema publicado en el Web, animación, libros electrónicos y aplicaciones para la generación y edición de imágenes.

- ✓ Para los epigrafistas, el e-Learning representa la convergencia del aprendizaje y el Internet.
- ✓ Y finalmente, para los visionarios y futuristas, el e-Learning es a la educación convencional lo que el e-Business a los negocios ordinarios. (Milenium)

1.3 Sistema de Administración de Aprendizaje (LMS).

Una solución e-Learning se puede conformar principalmente por un entorno de software diseñado para automatizar y gestionar el desarrollo de actividades formativas, también llamado Sistema de Administración de Aprendizaje (LMS). Un LMS, entre otras funciones, gestiona usuarios y recursos, administra el acceso, organiza catálogos de cursos, almacena datos de los usuarios, gestiona servicios de comunicación y provee informes para la gestión. Esta aplicación reside en un servidor web, donde alumnos, tutores, profesores o coordinadores, se conectan a través de un navegador web y pueden acceder a los contenidos existentes, ver el programa de asignaturas, debatir en un foro, participar en una tutoría, configurar los cursos, habilitar servicios, etc. teniendo en cuenta sus respectivos roles o accesos. Un LMS contiene además herramientas de comunicación, servicios y áreas para los diferentes elementos de cada curso, alumno o materia, gestión académica y administrativa para un control de acciones de los usuarios, necesario para el análisis posterior de los resultados, y un sistema que permita gestionar evaluaciones cuantitativas y cualitativas tanto de conocimiento como de asistencia.

Aunque un sistema LMS posee grandes ventajas, carece de facilidades para la creación de contenidos educativos, por lo que se hace necesaria la integración con un Sistema de Gestión de Contenidos (CMS, del inglés Content Management System). En consonancia, un Sistema de Administración de Contenidos de Aprendizaje (LCMS) engloba todas estas funcionalidades, facilitando la creación de los contenidos educativos y la gestión de éstos.

1.4 Sistema de Administración de Contenidos de Aprendizaje (LCMS).

Los sistemas de administración de contenidos educativos o LCMS, a diferencia de los LMS, están enfocados a la creación y administración de contenidos, a diferentes niveles, permitiendo de esta manera reestructurar la información y los objetivos de los contenidos de manera dinámica, para crear y modificar los objetos de aprendizajes (OA) que atiendan a necesidades y estilos de aprendizaje específicos. Estos contenidos, una vez dentro del sistema pueden ser combinados, asignados a distintos cursos, descargados, etc. Además, un LCMS permite el diseño de contenidos interactivos sin necesidad de conocimientos de programación o diseño web, es una herramienta especializada en contenidos de e-Learning que gestiona recursos educativos, evaluaciones, archivos multimedia, itinerarios formativos, etc.

Un LCMS brinda colecciones de recursos digitales que contienen, a manera de base de datos, contenidos digitales y objetos de información y aprendizaje que conforman las lecciones, unidades didácticas y cursos generados. Estas colecciones son llamadas repositorios de objetos de aprendizaje (ROA), que están dispuestos de tal manera que permiten consultar y reutilizar distintos contenidos, sin dañar la integridad de la información. Otras facilidades brindadas por un LCMS son:

- ✓ Herramientas de auditoría y publicación, para crear OA que puedan ser revisados por otros usuarios, considerando los estándares XML y SCORM.
- ✓ Herramientas de colaboración que promueven la educación compartida; una interfaz dinámica para entregar la información, evaluaciones, actividades, etc.
- ✓ Una aplicación administrativa que permite llevar un seguimiento del desempeño de los usuarios, almacenar sus perfiles y características personales.

El e-Learning está basado principalmente en los LMS y LCMS. Existen varios ejemplos de plataformas con este fin, dentro de las que se encuentra MOODLE, constituyendo un LCMS con mucho auge en la actualidad y la base para el desarrollo de este trabajo.

1.5 MOODLE

1.5.1 Entornos virtuales de aprendizaje.

Técnicamente, MOODLE es una aplicación que pertenece al grupo de los Gestores de Contenidos Educativos (LCMS), también conocidos como Entornos de Aprendizaje Virtuales (EVA), un subgrupo de los Gestores de Contenidos (CMS, Content Management Systems).

De una manera más coloquial, podemos decir que MOODLE es una aplicación para crear y gestionar plataformas educativas, es decir, espacios donde un centro educativo, institución o empresa, gestiona recursos educativos proporcionados por unos docentes y organiza el acceso a esos recursos por los estudiantes, y además permite la comunicación entre todos los implicados (alumnado y profesorado). (Sancho, 2007)

1.5.2 Significado de MOODLE y sus orígenes.

MOODLE fue creado por Martin Dougiamas de Perth, Australia. La palabra MOODLE, en inglés significa Entorno de Aprendizaje Dinámico Modular, Orientado a Objetos (Modular Object-Oriented Dynamic Learning Environment).

También es un verbo anglosajón que describe el proceso ocioso de dar vueltas sobre algo, haciendo las cosas como vienen a la mente... una actividad amena que muchas veces conlleva al proceso de comprensión y, finalmente, a la creatividad. Las dos acepciones se aplican a la manera en que se desarrolló MOODLE y a la manera en que un estudiante o docente podría aproximarse al estudio o enseñanza de un curso on-line. (Sancho, 2007)

1.5.3 MOODLE y Software libre.

MOODLE se distribuye gratuitamente como Software Libre (Open Source), bajo Licencia pública GNU. Esto significa que MOODLE tiene derechos de autor (copyright), pero que tenemos algunas libertades: podemos copiar, usar y modificar MOODLE siempre que aceptemos proporcionar el código fuente a otros, no modificar la licencia original y los derechos de autor, y aplicar esta misma licencia a cualquier trabajo derivado de él. Es fácil de instalar en casi cualquier plataforma con un servidor Web que soporte PHP. Sólo requiere que exista una base de datos (y se puede compartir). Con su completa abstracción de bases de datos, soporta las principales bases de datos (en especial MySQL). Finalmente, es importante destacar que, al ser MOODLE una aplicación Web, el usuario sólo necesita para acceder al sistema un ordenador con un navegador Web

instalado (Mozilla Firefox, Internet Explorer, o cualquier otro) y una conexión a Internet. Por supuesto, también se necesita conocer la dirección Web (URL) del servidor donde MOODLE se encuentre alojado y disponer de una cuenta de usuario registrado en el sistema. (Sancho, Moodle 1.8 Manual del Profesor, 2007)

1.5.4 Resumen de características de MOODLE.

- ✓ Entorno de aprendizaje modular y dinámico orientado a objetos, sencillo de mantener y actualizar.
- ✓ Excepto el proceso de instalación, no necesita prácticamente de mantenimiento por parte del administrador.
- ✓ Dispone de una interfaz que permite crear y gestionar cursos fácilmente.
- ✓ Los recursos creados en los cursos se pueden reutilizar.
- ✓ La inscripción y autenticación de los estudiantes es sencilla y segura.
- ✓ Resulta muy fácil trabajar con él, tanto para el profesorado como el alumnado.
- ✓ Detrás de él hay una gran comunidad que lo mejora, documenta y apoya en la resolución de problemas.
- ✓ Está basado en los principios pedagógicos constructivistas: el aprendizaje es especialmente efectivo cuando se realiza compartiéndolo con otros. (Sancho, Moodle 1.8 Manual del Profesor, 2007)

1.5.5 Estadísticas sobre MOODLE.

MOODLE tiene más de 48,166 sitios registrados oficialmente de diferentes tamaños (número de usuarios por sitio). A continuación se muestran algunos datos que reflejan la importancia que tiene MOODLE en el desarrollo del e-learning. (Moodle.org)

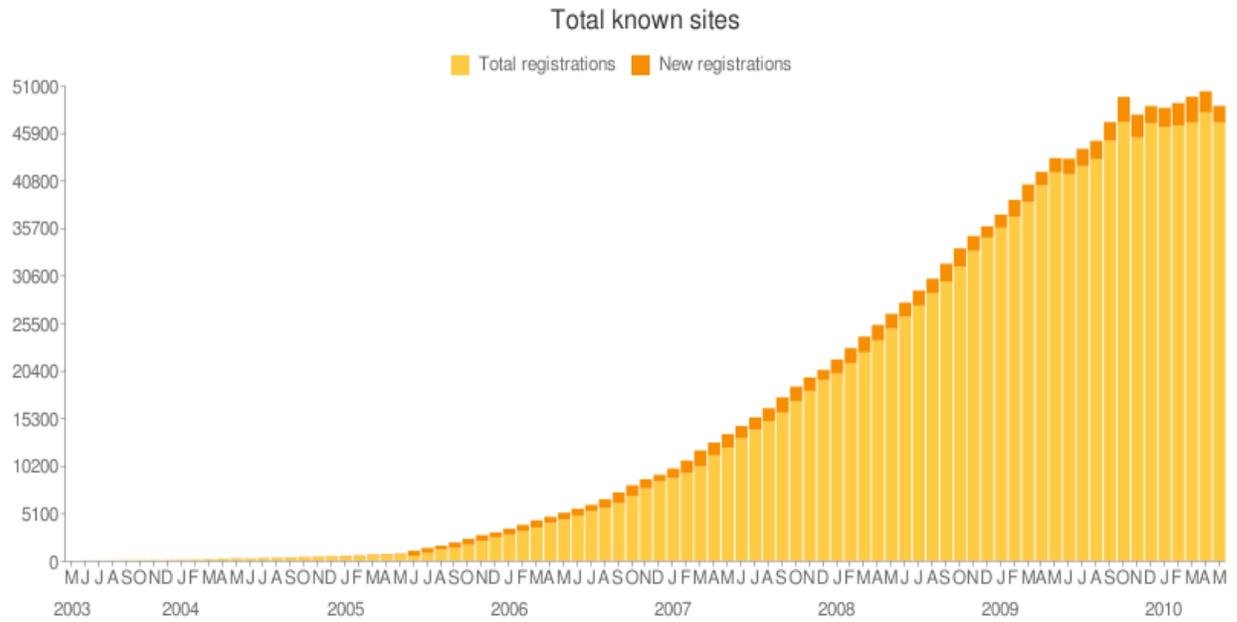


Figura 1: Total de sitios conocidos.

Sitios registrados	48,166
Número de países	212
Cursos	3,506,759
Usuarios	34,864,071
Profesores	1,224,485
Inscripciones	19,640,527
Foro de mensajes	55,742,221
Recursos	29,040,765
Cuestionario de preguntas	48,217,058

Tabla # 1: Estadísticas generales.



Figura 2: Registros por mes.

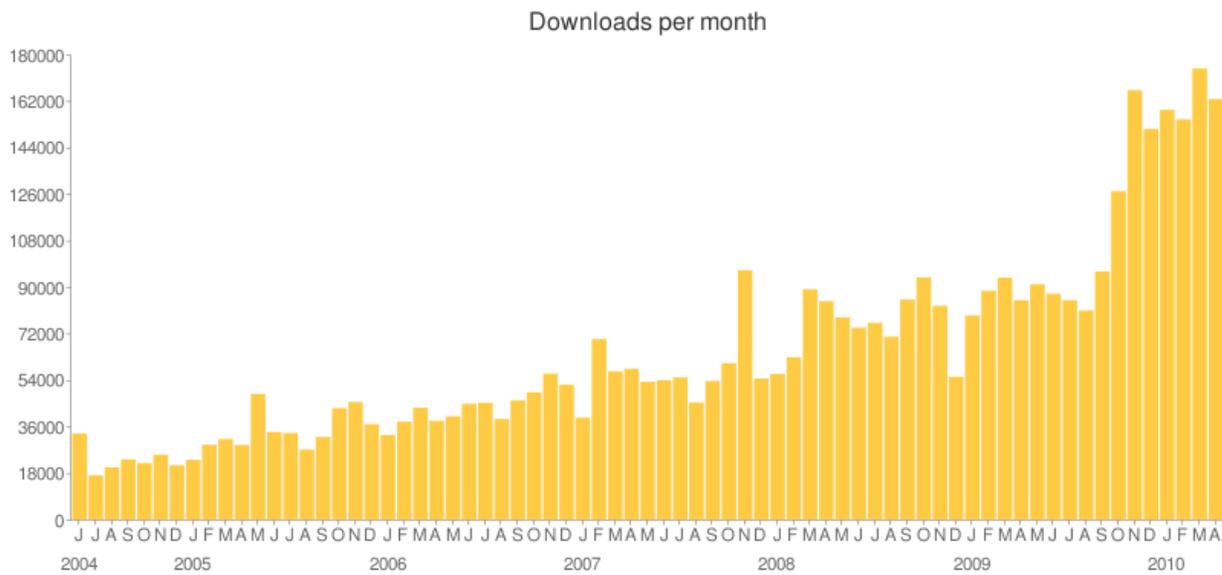


Figura 3: Descargas por mes.

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA.

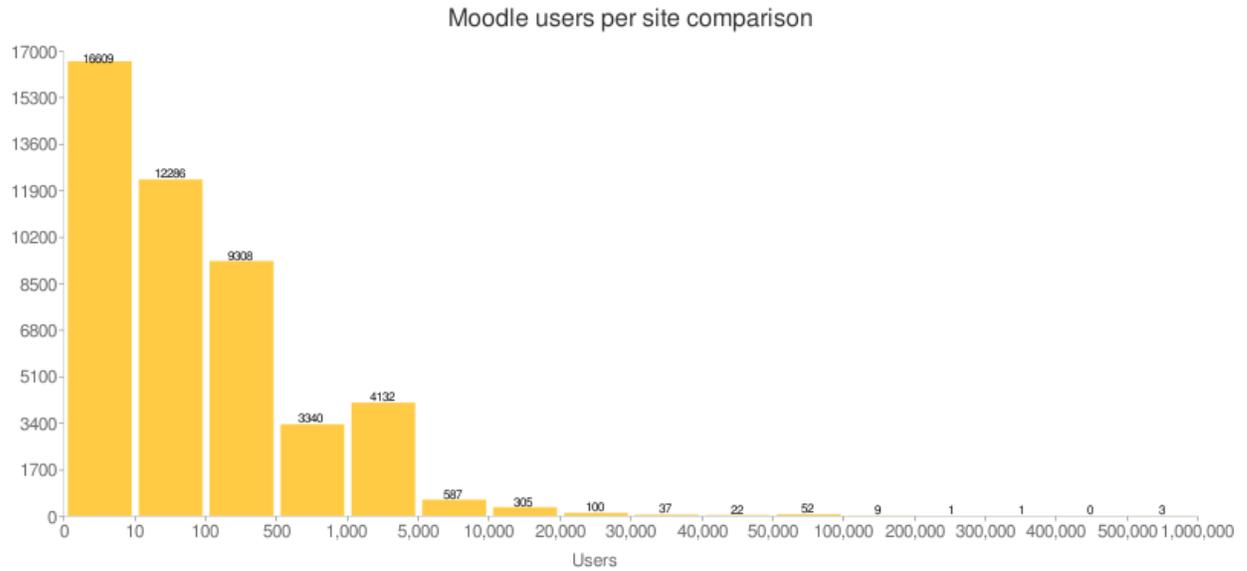


Figura 4: Cantidad de usuarios por sitio.

Sitio	Usuarios	Cursos
Moodle.org	921,621	65
OU online	651,727	5,332
MyLinE - Online Resources for Learning in English	204,166	46
Christian Courses	181,457	177
Campus Virtual de la Universidad de Barcelona	139,925	9,622
Learn Greek Online!	128,544	9
CQU Moodle	127,232	4,940
OpenLearn LearningSpace	101,254	580

Tabla # 2: Primeros 10 sitios por cantidad de usuarios.

Sitio	Usuarios	Cursos
VDU Moodle	3,566	59,920

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA.

Minha UFMG	62,887	56,385
UNCC Moodle	60,319	26,112
Universidad de Presbiteriana Mackenzie	54,978	23,813
ATENEA - Campus Virtual de la UPC	70,474	21,948
Ming Chuan University portal	62,027	21,338
Universidad Europea de Madrid	18,720	19,797
Concordia Course Web Sites	95,219	17,999
Moodle @ Lehigh	27,447	16,689

Tabla # 3: Primeros 10 sitios por cursos.

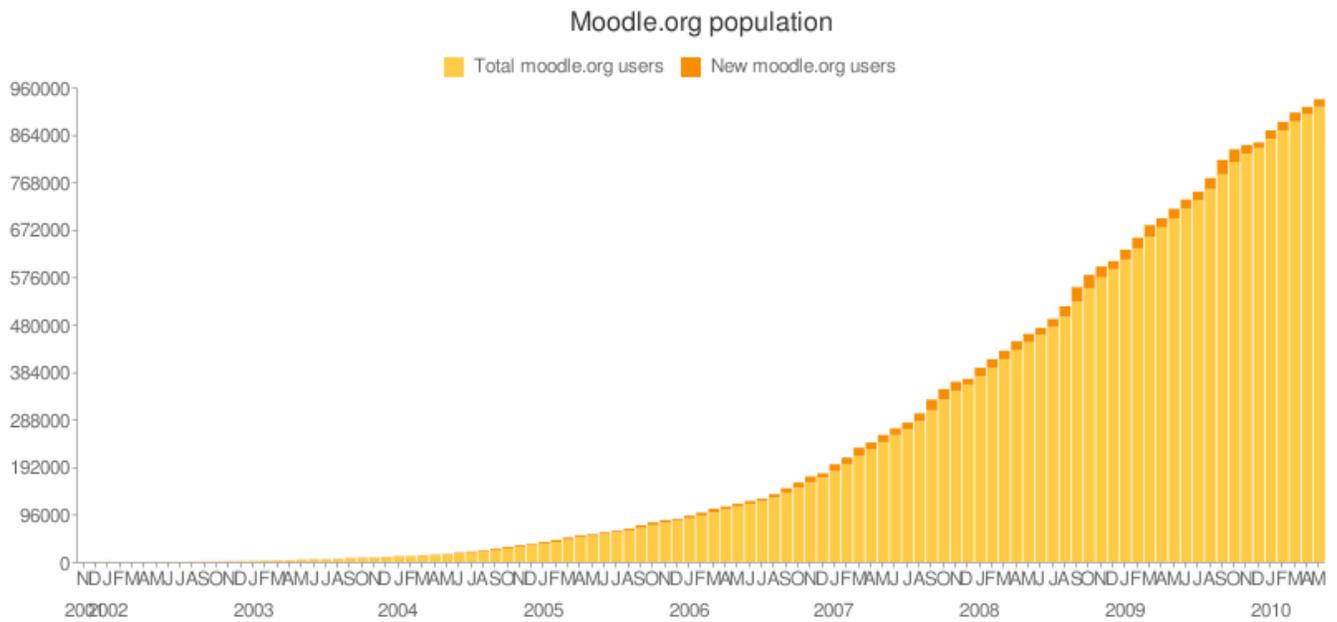


Figura 5: Cantidad de usuarios de Moodle.org.



Figura 6: Principales países.

País	Inscripciones
Estados Unidos	8,734
España	4,270
Reino Unido	3,096
Brasil	3,056
Alemania	2,237
Portugal	1,793
México	1,689
Australia	1,276
Italia	1,162
Colombia	1,124

Tabla # 4: Principales países.

1.5.6 Módulos de MOODLE

Los módulos son componentes auto-controlados que extienden las funcionalidades de una aplicación. Éstos deben ser fáciles de instalar y mantener, asegurando aislar el impacto que pueda

tener un fallo sobre el resto del programa. La lógica del funcionamiento de MOODLE se encierra en tres grandes grupos de módulos: los de comunicación, de recursos y de actividades.

- ✓ **Módulos de comunicación:** Estos módulos permiten que los alumnos puedan comunicarse con el profesor (hacer preguntas, plantear dudas, etc.) y entre ellos construir su propia comunidad de aprendizaje.

- ✓ **Módulos de recursos:** Estos módulos representan los contenidos y materiales del curso. Son todo tipo de textos, libros, apuntes, presentaciones de diapositivas, enlaces a páginas web externas etc., pensados para que los estudiantes los lean y estudien sobre ellos.

- ✓ **Módulos de actividades:** Estos módulos son la parte activa y colaborativa donde el estudiante tiene que hacer algo más que leer un texto. Debates y discusiones, resolución de problemas propuestos, redacción de trabajos, talleres, cuestionarios en línea, etc.

Existe una gran variedad de módulos para MOODLE que constituyen el eje principal de este sistema de gestión de aprendizaje. Estos módulos dan la posibilidad al usuario de enriquecer el diseño de cursos, permitiendo calificar de forma automática o manual, trabajar de forma colaborativa, además de brindar la posibilidad de debates y consultas. Por defecto al terminar la instalación de la aplicación existen diecisiete módulos estándares, y otros, no estándares, pueden ser añadidos al sistema aprovechando la flexibilidad de su arquitectura. Estos últimos pueden ser descargados desde la base de datos "Modules and Plugins" del sitio oficial de MOODLE e instalados descomprimiendo el paquete dentro del directorio /mod en la raíz del sistema.

El objetivo de las actividades en MOODLE se enfoca a la gestión del aprendizaje del estudiante, están pensadas para que el alumno trabaje y ejecute acciones, dando la posibilidad al profesor de seguir su progreso en el proceso de estudio y aprendizaje.

A continuación se explican brevemente los módulos más comunes:

- ✓ **Libro:** Este módulo facilita la elaboración de materiales sencillos de estudio compuestos por múltiples páginas, a la manera de un libro tradicional. Este módulo no es interactivo aunque existe la posibilidad de enlazar consultas, foros etc., así como de incluir objetos multimedia.
- ✓ **Tarea:** Este módulo permite asignar trabajos con objetivos individuales y se pueden especificar la fecha final de entrega y la calificación máxima. Las observaciones del profesor se adjuntan a la página de la tarea de cada estudiante y se le envía un mensaje de notificación.
- ✓ **Foro:** Son la principal herramienta de comunicación entre usuarios, ideal para publicar mensajes y entablar discusiones.
- ✓ **Chat:** Permite mantener conversaciones en tiempo real con otros usuarios.
- ✓ **Consulta:** Es una actividad que permite realizar encuestas rápidas y simples entre los miembros del curso.
- ✓ **Hotpot:** Este módulo permite al profesorado administrar los ejercicios elaborados con “Hot Potatoes” a través de MOODLE.
- ✓ **Etiqueta:** Este módulo permite añadir texto HTML y pequeños gráficos en los bloques de contenido. Puede servir para identificar las partes del curso y dar una estructura lógica a la página.
- ✓ **Encuesta:** Este módulo proporciona encuestas ya preparadas y contrastadas como instrumentos para el análisis de las clases en línea.
- ✓ **Taller:** Es una actividad para el trabajo en grupo con un vasto número de opciones. Permite a los participantes diversas formas de evaluar los proyectos de los demás, así como

proyectos-prototipo. También coordina la recopilación y distribución de esas evaluaciones de varias formas.

- ✓ **Cuestionario:** Es un módulo que permite definir una base de datos de preguntas que podrán ser reutilizadas en diferentes cuestionarios. Las preguntas y las respuestas de los cuestionarios pueden ser mezcladas aleatoriamente. Son capaces de calificarse automáticamente.
- ✓ **Recurso:** Este módulo admite la presentación de cualquier contenido digital, Word, Power Point, Flash, vídeo, sonido, etc. Los archivos pueden subirse y manejarse en el servidor, o ser creados en la plataforma usando formularios web (de texto o HTML). Además, es posible enlazar aplicaciones web para transferir datos.
- ✓ **Glosario:** Permite estructurar la información en conceptos y explicaciones como un diccionario o enciclopedia.
- ✓ **Wiki:** Este módulo permite a los participantes trabajar juntos en páginas web para añadir, expandir o modificar su contenido.
- ✓ **Lecciones:** Este módulo proporciona contenidos de forma flexible y normalmente termina con una pregunta y un número de respuesta posible. (Moodle.org)

1.6 Plataformas utilizadas en el mundo.

Existen múltiples plataformas que emplean funcionalidades donde el usuario puede realizar su auto-aprendizaje o ser parte de comunidades o grupos de aprendizaje, entre las que se encuentran:

- ✓ Atutor.
- ✓ MOODLE.
- ✓ Decebo LMS.
- ✓ Proyecto MERLIN.

1.6.1 Atutor.

Es un Sistema de Gestión de Contenidos de Aprendizaje (Learning Content Management System LCMS) de Código Abierto basado en la Web y diseñado con el objetivo de lograr accesibilidad y adaptabilidad. Los administradores pueden instalar o actualizar Atutor en minutos. Los educadores pueden rápidamente ensamblar, empaquetar y redistribuir contenido educativo, y llevar a cabo sus clases online. Los estudiantes pueden aprender en un entorno de aprendizaje adaptativo. (Fedapas.org)

1.6.2 Decebo LMS.

Es una plataforma de e-Learning LCMS por sus siglas en inglés, usada en los mercados corporativos y de educación superior.

Es un sistema para e-learning realizado en PHP, desarrollado originalmente en Italia, ahora con un equipo de desarrollo internacional. Es de código abierto y de uso gratuito. Se puede integrar con el gestor de contenidos Decebo CMS.

Este producto, desarrollado bajo licencia GNU/GPL, está dirigido a la administración de cursos académicos online.

La plataforma incluye 12 idiomas, cumple con el estándar SCORM 1.2 y puede ser usada bajo diferentes modelos didácticos, que incluyen: Blended learning, aprendizaje auto dirigido y colaborativo; incluso soporta el aprendizaje social por medio de chat, Wiki, foros y otras 53 funciones.

1.6.3 Proyecto MERLIN.

Plataforma de aprendizaje a distancia basada en tecnología de Portlets y Web 2.0 para una enseñanza participativa.

El principal objetivo científico del proyecto es la aplicación de la tecnología de Portlets a la creación de una plataforma tecnológica de aprendizaje a distancia, que se beneficie de las

facilidades que ofrece dicha tecnología para la integración de nuevos módulos y de implantación. Además, el proyecto afronta esta plataforma con una concepción Web 2.0 de la misma, ofreciendo interfaces con una interacción natural. (Merlin)

1.7 Plataformas utilizadas en Cuba.

Nuestro país también se ha insertado en este nuevo paradigma y ha desarrollado algunas plataformas propias o en conjunto con otros países:

- ✓ SEPAD.
- ✓ aprenDIST.
- ✓ Medcampus.

1.7.1 SEPAD.

El Sistema de Enseñanza Personalizado A Distancia es una plataforma para la tele-formación cuya aspiración principal es llevar la educación a todos independientemente de su capacidad tecnológica o de conectividad. Para ello cuenta con varias interfaces que van desde el clásico ambiente Web para usuarios que tienen la posibilidad de conexión en línea, o un cliente para acceder a los servicios de la plataforma a través de protocolos de correo electrónico o la versión multimedia capaz de ejecutarse sin necesidad de conexión alguna.

Además cuenta con una herramienta para la elaboración de los cursos que no requiere de conexión en línea. La plataforma cuenta con un aula virtual donde se acceden a los materiales didácticos, búsquedas, auto evaluaciones, calificaciones y los servicios de tutorías como son la mensajería interna, los foros de debates, el sistema de anuncios, las noticias y las salas de chat temáticas. Desde el punto de vista de los tutores y profesores el sistema cuenta con ambientes donde estos pueden seguir el proceso de aprendizaje de sus alumnos.

Desarrollado por un grupo de investigadores de la Universidad Central “Martha Abreu” de las Villas la plataforma SEPAD cuenta ya con otras herramientas vinculadas a ella como son SEPADHP, herramienta para la elaboración de los cursos, SEPADMedia, cliente para acceso fuera de línea a los servicios de la plataforma y el directorio SEPAD que es un catálogo nacional que permite la búsqueda y localización de los cursos disponibles en la plataforma. (SEPAD)

1.7.2 aprenDIST.

Es una plataforma de educación a distancia, desarrollada en el Instituto Superior Politécnico “José Antonio Echeverría”. Es un sistema diseñado para un alumnado con determinadas características, que tornan diferente el proceso de enseñanza- aprendizaje tradicional (o sea presencial), además de ser una plataforma flexible y ajustable a toda una variedad de cursos, por lo que el usuario de este sistema, puede dominarlo, con sólo tener algún conocimiento de navegación Web. (Oliveros Guntín, Rosete Suárez., & Peña Sepúlveda, 2005).

Cuenta además con otros módulos incorporados como son:

- aprenDIST Sender, módulo que permite añadir la posibilidad de que la comunicación con el servidor se realice a través del protocolo SMTP en lugar del protocolo HTTP, lo que posibilita que el estudiante pueda contar en la versión local con casi todas las funcionalidades de la versión online.
- Módulo offline de la biblioteca: Este módulo permite la edición fuera de línea de los documentos a cargar en la biblioteca.
- Herramienta de autor: Con el objetivo de facilitar el proceso de montaje de los cursos.
- Suite del Profesor: Este módulo además de permitir la creación de los cursos le permite al profesor interactuar con la plataforma local permitiendo que el profesor pueda ir viendo la forma en que se mostrará su curso en la plataforma según lo va creando.

1.7.3 Medcampus.

Plataforma desarrollada por un grupo de investigadores de la Universidad Virtual de la Salud en Cuba y de la Universidad de Holguín con los requerimientos técnicos y pedagógicos del sector de la salud en el país que posibilita el soporte a la actividad docente de pregrado y posgrado en todos los niveles del Sistema Nacional de Salud Pública.

Medcampus es una plataforma descentralizada, que puede instalarse en un clúster de servidores LINUX, con varios gigabytes de RAM y un gran poder operacional; en un servidor regional de Infomed para soportar cursos a nivel municipal o provincial, con diferentes niveles de funcionamiento, la configuración general del sistema permite regular la entrada de los usuarios locales o para un rango de máquinas (números IP) o a un servidor Proxy que brinde

acceso a cierto número de usuarios o a nivel nacional con un gran número potencial de usuarios del país y extranjeros. El sistema se ha calculado para una media de 10 000 peticiones/segundo de operaciones para 100 usuarios-sistemas. (Rodríguez Cáceres & Torres Acosta)

1.8 Instituciones y Universidades que utilizan MOODLE en Cuba.

De todas las plataformas ya mencionadas, MOODLE ha tenido gran aceptación en las universidades cubanas, por lo que muchas de estas junto a otros centros de investigación la utilizan adaptándola a sus necesidades gracias a las facilidades que brinda esta plataforma como sistema de código abierto. Algunas instituciones cubanas que usan MOODLE son:

- ✓ Facultad de Educación a Distancia de la UH.
- ✓ AulaEnRed para la enseñanza de la Física de la UH.
- ✓ Centro Nacional de Bioinformática.
- ✓ Instituto de Ciencia Animal.
- ✓ Universidad de Camagüey.
- ✓ Universidad de Holguín. Oscar de Lucero Moya.
- ✓ Universidad de las Ciencias Informáticas.
- ✓ Universidad de Pinar del Río "Hermanos Saíz Montes de Oca".

1.9 Herramientas y Metodología.

1.9.1 Metodología de desarrollo.

Para la construcción de un software con calidad es necesario utilizar al menos una metodología de desarrollo. Las metodologías que a continuación se detallan han tenido resultados relevantes en la implementación de proyectos de menor y gran escala.

1.9.1.1 RUP.

La metodología RUP, llamada así por sus siglas en inglés Rational Unified Process, divide en 4 fases el desarrollo del software:

- ✓ Inicio: El Objetivo en esta etapa es determinar la visión del proyecto.
- ✓ Elaboración: En esta etapa el objetivo es determinar la arquitectura óptima.
- ✓ Construcción: En esta etapa el objetivo es llegar a obtener la capacidad operacional inicial.
- ✓ Transmisión: El objetivo es llegar a obtener la liberación del proyecto.

Cada una de estas etapas es desarrollada mediante el ciclo de iteraciones, la cual consiste en reproducir el ciclo de vida en cascada a menor escala. Los objetivos de una iteración se establecen en función de la evaluación de las iteraciones precedentes. (Sanchez, 2004.)

1.9.2 Lenguaje de Modelado.

1.9.2.1 Lenguaje Unificado de Modelación (UML).

El desarrollo del Unified Modeling Lenguaje (UML) es la consecuencia de la unión de los métodos de Booch (Metodología de Grady Booch) y la OMT (Object Modeling Techniques) para producir lo que primero se conoció como Método Unificado.

UML es un lenguaje gráfico para visualizar, especificar, construir y documentar los artefactos de un sistema de software con tecnología orientado a objetos, proporciona una forma estándar de escribir los planos de un sistema, cubriendo tanto los conceptos de los procesos del negocio y funciones del sistema, como las acciones concretas: las clases escritas en un lenguaje de programación específico, esquemas de bases de datos y componentes software reutilizables.

UML no es una salida de Booch, OMT, OOSE, sino una evolución y síntesis de estos tres, es el resultado de la experiencia sumada. (Clikear.com)

1.9.3 Herramienta CASE.

1.9.3.1 Visual Paradigm.

Visual Paradigm for UML es una herramienta UML profesional que soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientados a objetos, construcción, pruebas y despliegue. El software de modelado UML ayuda a una más rápida construcción de aplicaciones de calidad, mejores y a un menor coste. Permite dibujar todos los tipos de diagramas de clases, código inverso, generar código desde diagramas y generar documentación. La herramienta UML CASE también proporciona abundantes tutoriales de UML, demostraciones interactivas y proyectos UML. (Visual Paradigm)

1.9.4 Herramienta para el almacenamiento de datos.

1.9.4.1 PostgreSQL.

PostgreSQL es un sistema de gestión de bases de datos objeto-relacional, distribuido bajo licencia BSD y con su código fuente disponible libremente. Actualmente se reconoce como el sistema de gestión de bases de datos de código abierto más potente del mercado.

PostgreSQL utiliza un modelo cliente/servidor y usa multiprocesos en vez de multi-hilos para garantizar la estabilidad del sistema. Un fallo en uno de los procesos no afectará el resto y el sistema continuará funcionando.

A continuación, algunas de las características más importantes y soportadas por PostgreSQL:

- ✓ Es una base de datos 100% ACID.
- ✓ Integridad referencial.
- ✓ Espacio de almacenamiento para los datos reales, Tablespaces.
- ✓ Transacciones anidadas, Nested transactions (savepoints).
- ✓ Replicación asíncrona.
- ✓ Actualizar en dos fases, Two phase commit.
- ✓ Copias de seguridad en caliente (Online/hot backups).
- ✓ Estándar de codificación Unicode.
- ✓ Juegos de caracteres internacionales.

- ✓ Control de concurrencia multi-versión, Multi-Version Concurrency Control (MVCC).
- ✓ Acceso encriptado vía SSL.
- ✓ Completa documentación.
- ✓ Licencia BSD.
- ✓ Disponible para Linux y UNIX en todas sus variantes (AIX, BSD, HP-UX, SGI IRIX, Mac OS X, Solaris, Tru64) y Windows.

1.9.4.2 EMS SQL Manager for PostgreSQL.

EMS SQL Manager for PostgreSQL es una herramienta de alto rendimiento para la administración de bases de datos PostgreSQL. Funciona con cualquier versión de PostgreSQL, y soporta las últimas características de PostgreSQL incluyendo enumerados, búsqueda de texto, XML y tipos de datos UUID, el índice de PostgreSQL, orden de clasificación clave, matrices de tipos de compuestos, y otros. SQL Manager for PostgreSQL ofrece una amplia variedad de herramientas potentes de base de datos tales como Visual Database Designer para crear bases de datos PostgreSQL de manera sencilla, Visual Query Builder para construir consultas complejas de PostgreSQL, un potente editor BLOB y muchas otras características útiles para una eficiente administración de PostgreSQL.

1.9.4.2.1 Características Principales.

- ✓ Soporte de datos UTF8.
- ✓ Excelente interfaz y herramientas de texto para la construcción de consultas.
- ✓ La rápida gestión de bases de datos y navegación.
- ✓ Herramientas de manipulación avanzada de datos.
- ✓ Soporte completo de PostgreSQL hasta la versión 8.4.
- ✓ Mejora de explorador de base de datos para facilitar la gestión de todos los objetos PostgreSQL.
- ✓ Administración efectiva de seguridad.
- ✓ Conexión a través del reenvío de puerto local a través del túnel SSH.

- ✓ El acceso a PostgreSQL Server a través del protocolo HTTP.(SQL Manage.net)

1.9.5 Herramienta de Desarrollo (IDE).

1.9.5.1 Zend Studio for Eclipse.

Zend Studio for Eclipse combina la tecnología de Zend y PHP de Eclipse Tools (PDT) para crear el más potente IDE para el desarrollo de aplicaciones Web. Que incluye depuración integrada, de perfiles y de código; probar capacidades de cobertura; un equipo de control de versiones de apoyo extensible; el Zend Framework; soporte para Web Services. Presenta múltiples idiomas y extensibilidad de la vibrante comunidad de código abierto de Eclipse. Los desarrolladores tienen las herramientas necesarias para el apoyo de todo el ciclo de vida de aplicación PHP. (TARINGA)

1.9.6 Servidor Web.

1.9.6.1 Apache.

Apache se basó originalmente en la codificación e ideas plasmadas en uno de los servidores HTTP más popular, el NCSA httpd 1.3 (principios de 1995) y ha puesto en marcha un sistema que puede rivalizar (y probablemente sobrepasar) a casi cualquier otro servidor basado en UNIX HTTP en cuanto a funcionalidad, eficacia y rapidez. Desde su comienzo, se caracterizado por realizar actualizaciones constantes de su código fuente, por lo que se ha vuelto a escribir completamente, e incluye muchos rasgos nuevos. Apache, actualmente, es el servidor Web más popular en Internet, según el Netcraft Survey. (FACILNET)

1.9.7 Lenguajes de programación.

1.9.7.1 HTML.

El HTML es un Lenguaje de Marcas Hipertextuales, diseñado para estructurar textos para generar páginas web. Gracias a Internet y a los navegadores web, el HTML se ha convertido en el formato más fácil para la creación de páginas web debido a su sencillez.

La mayoría de las etiquetas del lenguaje HTML son semánticas. La interpretación de las etiquetas es realizada por el navegador web. El lenguaje HTML es extensible, se le pueden

añadir características, etiquetas y funciones adicionales para el diseño de páginas web, generando un producto vistoso, rápido y sencillo. (Hooping.net)

1.9.7.2 JavaScript.

JavaScript es un lenguaje de programación que se utiliza principalmente para crear páginas web dinámicas.

Una página web dinámica es aquella que incorpora efectos como texto que aparece y desaparece, animaciones, acciones que se activan al pulsar botones y ventanas con mensajes de aviso al usuario.

Técnicamente, JavaScript es un lenguaje de programación interpretado, por lo que no es necesario compilar los programas para ejecutarlos. En otras palabras, los programas escritos con JavaScript se pueden probar directamente en cualquier navegador sin necesidad de procesos intermedios. (Pérez)

1.9.7.3 CSS Hojas de Estilo en Cascada.

Es un mecanismo simple que describe cómo se va a mostrar un documento en la pantalla, o cómo se va a imprimir, o incluso cómo va a ser pronunciada la información presente en ese documento a través de un dispositivo de lectura. Esta forma de descripción de estilos ofrece a los desarrolladores el control total sobre estilo y formato de sus documentos.

CSS se utiliza para dar estilo a documentos HTML y XML, separando el contenido de la presentación. Los estilos definen la forma de mostrar los elementos HTML y XML. CSS permite a los desarrolladores Web controlar el estilo y el formato de múltiples páginas Web al mismo tiempo. Cualquier cambio en el estilo marcado para un elemento en la CSS afectará a todas las páginas vinculadas a esa CSS en las que aparezca ese elemento. (World Wide Web Consortium (W3C) - España)

1.9.7.4 PHP.

PHP es el acrónimo de "PHP: Hypertext Preprocessor", es decir, un "preprocesador del hipertexto" que se basa en una sintaxis parecida al C, al Java y al Perl, con unos añadidos más. Nació en 1994 como proyecto "personal" y la primera versión se utilizó públicamente en 1995 con el nombre "Personal Home Page". Como cada buen proyecto que llama la atención de los usuarios y de los programadores, el lenguaje se ha desarrollado como proyecto open-source hasta el punto que, en 1996, ya se estaba utilizando en 15.000 sitios web.

El PHP es un lenguaje de scripting server-side. Se trata de un lenguaje " embedded" en las páginas que lo incluyen, por citar un ejemplo, el lenguaje JavaScript. La principal diferencia es que el PHP lo ejecuta en el servidor en lugar de ejecutarlo en el cliente directamente. Por lo tanto, no se necesitan compatibilidades particulares o estándar definidos por otros (como el ejemplo más clásico del JavaScript). El mecanismo de realización de los scripts, es parecido al lenguaje ASP (Active Server Pages). (HTMELPOINT.com)

1.9.7.5 XML.

XML fue creado al amparo del Word Wide Web Consortium (W3C), organismo que vela por el desarrollo de WWW partiendo de las amplias especificaciones de SGML (Generalized_Markup Language).

XML no es más que un conjunto de reglas para definir etiquetas semánticas que nos organizan un documento en diferentes partes. XML es un metalenguaje que define la sintaxis utilizada para definir otros lenguajes de etiquetas estructurados. (Arenas)

1.10 Diseño Metodológico de la Investigación.

1.10.1 Métodos Teóricos utilizados.

- **Histórico-lógico:** En la primera parte de la investigación se desarrolló un estudio del estado del arte de la problemática; y se analizaron las herramientas utilizadas para desarrollar el módulo en cuestión.
- **Modelación:** Este método permite la creación de modelos, (propuestas, alternativas, estrategias, etc.) que visualiza una reproducción simplificada de la realidad, que consiente descubrir y estudiar nuevas relaciones y cualidades del objeto de estudio. La modelación es justamente el proceso mediante el cual se crean modelos con vistas a investigar la realidad. Aplicando este método utilizamos la metodología RUP para el desarrollo de la solución y el lenguaje UML para la modelación gráfica.
- **Método Analítico-Sintético:** Permitió extraer y analizar profundamente toda la información referente a la plataforma MOODLE, sus características principales, funcionamiento, así como hacer un estudio detallado de todos los elementos relacionados con la propuesta.

1.10.2 Métodos Empíricos utilizados.

- **Entrevista:** Se utilizó la entrevista como una conversación planificada con los clientes, para obtener información acerca del problema en cuestión. Su uso constituyó sin dudas un medio para el conocimiento cualitativo de las características particulares del proceso e influyó en el posterior análisis y diseño del producto de software.
- **Método de la observación:** La observación científica es la percepción planificada dirigida a un fin y relativamente prolongada de un hecho o fenómeno. Se realiza de forma consciente y orientada a un objetivo determinado. Para el desarrollo del presente trabajo estuvimos involucrados mediante la observación, en el proceso de creación y configuración de actividades en MOODLE por asesores del Departamento Central de Programación.

Conclusiones.

Después del estudio de la fundamentación teórica de las herramientas y utilizando como punto de partida la implementación de sistemas similares al nuestro, concluimos que utilizaremos como metodología para el desarrollo del sistema el Proceso Unificado de Desarrollo (RUP) que a su vez hará uso del Lenguaje Unificado de Modelado (UML) y el Zend Studio for Eclipse para realizar toda la implementación de nuestra aplicación con PHP y HTML y como gestor de base de datos PostgreSQL.

Como resultado de la investigación realizada se ha podido comprobar que al menos en las versiones estándar de las plataformas e-learning estudiadas no se encuentra un módulo que permita la creación y realización de Actividades de Auto y/o Co-evaluación. En este aspecto no se generaliza, ya que estas plataformas al ser distribuidas como sistemas de código abierto permiten que se le agreguen funcionalidades nuevas por cualquier desarrollador en el mundo. En el caso de este trabajo de diploma que se desarrolla en MOODLE, se puede decir que existen módulos ya implementados que de alguna manera sirven de punto de partida para el desarrollo del mismo, como es el caso del módulo Cuestionario.

Capítulo 2. CARACTERÍSTICAS DEL SISTEMA

Introducción.

En este capítulo se describen las características y los procesos que contienen toda la información relacionada con el sistema propuesto. Se recogen aspectos importantes de todo el entorno de trabajo para un mayor entendimiento del problema planteado.

Se describe el proceso estudiando todas las actividades, responsables de las mismas y como se llevan a cabo. Se extraen los requerimientos necesarios para el cumplimiento de los objetivos generales y específicos así como la definición de los casos de uso del sistema.

En general el capítulo aborda todos los temas relacionados con la ubicación del problema y necesidades que se presentan para resolverlo sin tener definida una solución.

2.1 Objeto de estudio.

El objeto de estudio es un módulo que permita la Auto-evaluación y Co-evaluación de los alumnos, para que el profesor tenga en cuenta mayores elementos a la hora de emitir una evaluación formativa centrada en el aprendizaje.

2.2 Propuesta del Sistema.

Se pretende desarrollar un nuevo módulo que contribuya a una evaluación formativa centrada en el aprendizaje en el Entorno Virtual de Aprendizaje que se emplea en La Universidad de Ciencias Informáticas.

El estudiante podrá evaluarse y a la vez evaluar a sus compañeros, dado una serie de elementos y aspectos definidos por un grupo de especialistas del Departamento Central de Programación.

2.3 Modelo de Dominio.

El modelo del dominio tiene como objetivo comprender y describir las clases más importantes dentro del contexto del sistema y relacionarlas unas con otras:

- ✓ Objetos del negocio que representan cosas que se manipulan en este.
- ✓ Objetos del mundo real y conceptos de los que el sistema debe hacer un seguimiento.
- ✓ Sucesos que ocurrirán o han ocurrido.

El módulo propuesto tiene su base en un sistema desarrollado, por esta razón realizamos un modelo del dominio.

2.3.1 Descripción del Problema de Dominio

El Entorno Virtual de Aprendizaje utilizado en la Universidad permite tanto al estudiante como al profesor interactuar con este mediante una serie de recursos y funcionalidades que brinda la plataforma, le permite al profesor realizar actividades, evaluaciones y cursos para que el estudiante ponga en práctica los conocimientos adquiridos, estos recursos emiten una información recogida en el Libro de Calificación donde el profesor puede darle seguimiento a los estudiantes y emitir una evaluación guiándose por estos valores.

Actualmente en la UCI existe el Departamento Docente Central Técnicas de Programación (DDCTP) encargado de dirigir metodológicamente el proceso de enseñanza en las asignaturas que conforman la disciplina de Programación, este departamento presenta la necesidad de incorporar un nuevo tipo de actividad llamada Actividad de Auto-evaluación y/o Co-evaluación, en la cual los estudiantes pueden ser partícipes de su propia evaluación y la de sus compañeros de grupo.

La Actividad de Auto y/o Co-evaluación estará conformada por aspectos previamente definidos por los especialistas que pertenecen al departamento. El profesor será el encargado de conformar y publicar esta actividad y asignarla al grupo de estudiantes al cual le imparte la asignatura. Luego el estudiante podrá realizar la actividad y esta se registrará en el Libro de Calificación del Curso.

MOODLE contiene varios tipos de actividades que se encuentran separadas por módulos dentro de la plataforma, se necesita incorporar un nuevo módulo que permita realizar una Actividad de Auto y/o Co-evaluación ya que la plataforma no presenta un espacio donde los estudiantes puedan ser partícipes de sus propias evaluaciones, es decir autoevaluarse y evaluar a sus compañeros, y de esta manera propiciar que se tenga en cuenta una evaluación formativa.

2.3.2 Representación del modelo del dominio.

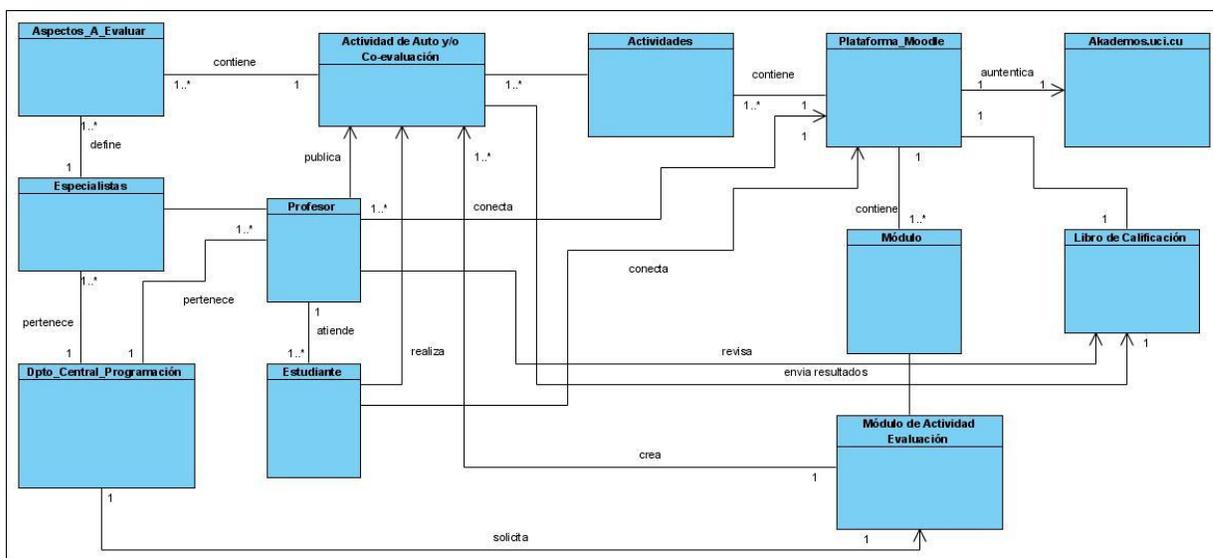


Figura 7 Modelo de Dominio.

2.4 Captura de Requisitos.

2.4.1 Especificación de los requisitos de software.

2.4.1.1 Requerimientos Funcionales.

RF1 Insertar Actividad de Auto y/o Co-evaluación.

- 1.1 Insertar Nombre.
- 1.2 Descripción.
- 1.3 Seleccionar Aspecto.
- 1.4 Seleccionar Grupo.

1.5 Tipo de Actividad.

1.5.1 Auto-evaluación.

1.5.2 Co-evaluación.

1.5.3 Ambas.

RF2 Modificar Actividad de Auto y/o Co-evaluación.

2.1 Nombre.

2.2 Descripción.

2.3 Seleccionar Aspecto.

2.4 Seleccionar Grupo.

2.5 Tipo de Actividad.

2.5.1 Auto-evaluación.

2.5.2 Co-evaluación.

2.5.3 Ambas.

RF3 Mostrar Actividad de Auto y/o Co-evaluación.

RF4 Eliminar Actividad de Auto y/o Co-evaluación.

RF5 Insertar aspectos a evaluar en la Actividad de Auto y/o Co-evaluación.

5.2 Descripción.

RF6 Eliminar aspectos a evaluar en la Actividad de Auto y/o Co-evaluación.

RF7 Modificar aspectos a evaluar en la Actividad de Auto y/o Co-evaluación.

7.2 Descripción.

RF8 Mostrar aspectos a evaluar en Actividad de Auto y/o Co-evaluación.

RF9 Insertar evaluaciones Actividad de Auto y/o Co-evaluación.

RF10 Evaluar Actividad de Auto y/o Co-evaluación.

RF11 Mostrar resultados en el Libro de Calificación.

RF12 Crear Formulario de una instancia del Módulo de Actividad "Evaluación".

RF13 Definir permisos de acceso al Módulo de Actividad "Evaluación".

RF14 Presentar una la lista de todas las instancias en un curso.

RF15 Añadir una nueva instancia.

RF16 Eliminar una instancia.

RF17 Mostrar una instancia en particular.

RF18 Actualizar una instancia.

2.4.1.2 Requerimientos No Funcionales.

2.4.1.2.1 Software.

- ✓ Servidor web Apache 2.5.6, pero MOODLE debe funcionar bien en cualquier servidor web que soporte PHP.
- ✓ Una instalación de PHP que esté funcionando (versión 4.1.0 o posterior), con las siguientes características:
 - GD library activada, con soporte para los formatos JPG y PNG.
 - Soporte para sesiones (sessions) activado.
 - Habilitada la posibilidad de enviar (upload) archivos.
 - Modo seguro (safe mode) desactivado.
- ✓ Una base de datos funcionando: Se recomiendan MySQL o PostgreSQL .

2.4.1.2.2 Hardware.

- ✓ El Hardware mínimo requerido:
 - Pentium Core 2 Duo mayor de 2,0 GHz.
 - 1GB de memoria RAM.
 - 180 GB de disco duro.

2.4.1.2.3 Interfaz de usuario.

- ✓ Mostrar Actividad de Auto y/o Co-evaluación de forma entendible al estudiante y al profesor.
- ✓ Gestionar Actividad de Auto y/o Co-evaluación por el profesor.
- ✓ Gestionar Aspecto.

2.4.1.2.4 Usabilidad.

- ✓ Recurso de fácil manejo.

2.4.1.2.5 Seguridad y Privacidad.

- ✓ Asegurar que sea el profesor quien solo pueda crear una nueva Actividad de Auto y/o Co-evaluación.
- ✓ Mantener la integridad de las respuestas de los estudiantes.

2.5 Definición de los actores del sistema.

Actor	Justificación
Administrador del Sistema	Es el actor que va a administrar el sistema, es el que va a configurar el módulo y hacer modificaciones en el mismo

Tabla # 5: Actor Administrador del Sistema.

Actor	Justificación
Estudiante	Es el actor interesado en resolver la Actividad de auto y co-evaluación, es el que se beneficia ya que influye en su evaluación. Esta actividad se visualiza y le brinda la posibilidad de responder a las preguntas.

Tabla # 6: Actor Estudiante.

Actor	Justificación
Profesor	Es el actor que crea la Actividad de auto y co-evaluación, definiendo las preguntas o aspectos que son de interés para emitir una evaluación formativa centrada en el aprendizaje.

Tabla # 7: Actor Profesor.

Actor	Justificación
Usuario	Es el actor que interactúa con la plataforma, el cual tiene diferentes niveles de acceso.

Tabla # 8: Actor Usuario.

2.6 Diagrama de casos de uso del sistema.

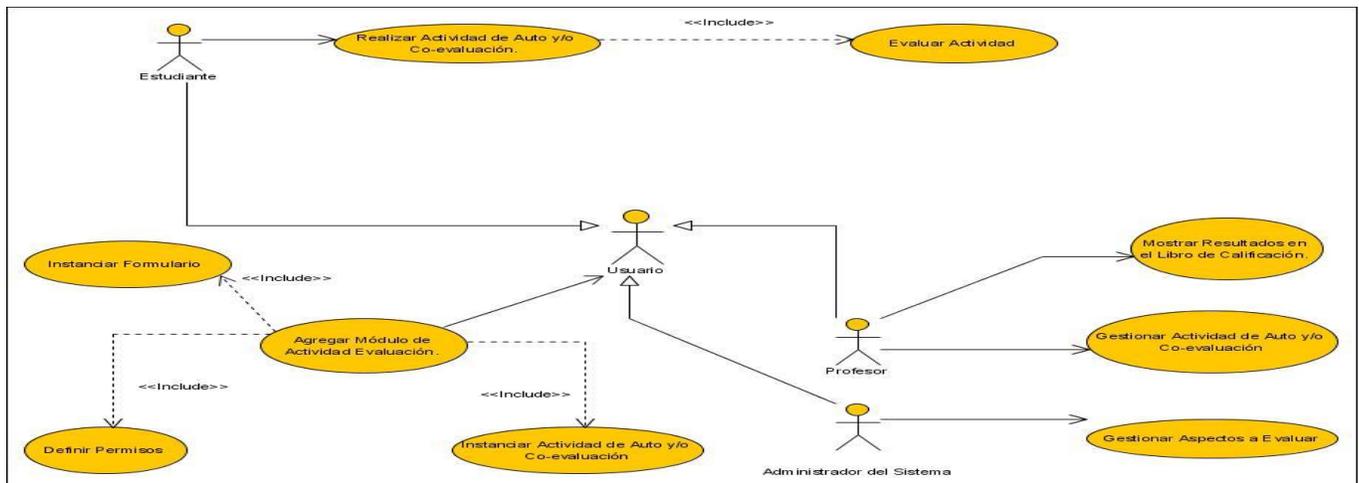


Figura 8 Diagrama de CU del Sistema.

La descripción textual de los casos de uso del sistema se puede detallar en el Anexo # 1.

2.7 Esquema de un Módulo

En MOODLE todos los módulos están conformados por un paquete de ficheros organizados que responden a sus necesidades específicas. Según la documentación oficial de MOODLE, todo módulo posee algunos elementos obligatorios:

- ✓ mod_form.php: un formulario para establecer o actualizar una instancia de este módulo.
- ✓ version.php: define alguna meta-información y proporciona código de actualización.

- ✓ db/install.xml: define la estructura de las tablas para todos los tipos de bases de datos. Se utiliza cuando se instala el módulo.
- ✓ db/upgrade.php: define los cambios en la estructura de las tablas. Se utiliza cuando se actualiza el módulo.
- ✓ db/access.php: define los permisos.
- ✓ index.php: una página para presentar la lista de todas las instancias en un curso.
- ✓ view.php: una página para ver una instancia en particular.
- ✓ 'lib.php': cualquiera/todas las funciones definidas para el módulo deben estar aquí. Si el módulo se llama "new_module", entonces las funciones requeridas incluyen:
 - new_module_add_instance () - código para añadir una nueva instancia de "new_module".
 - new_module_update_instance () - código para actualizar una instancia existente.
 - new_module_delete_instance () - código para borrar una instancia.
 - new_module_user_outline () - dada una instancia, devuelve un resumen de una contribución de un usuario.
 - new_module_user_complete () - dada una instancia, imprime detalles sobre la contribución de un usuario.
- ✓ Otras funciones interesantes pero no obligatorias son:
 - new_module_delete_course () - para borrar todo lo que sea necesario tras borrar todas las instancias de un curso.
 - new_module_process_options () - para pre-procesar la información de los ajustes de la instancia.
- * Para evitar posibles conflictos, cualquiera de las funciones de un módulo debe ser nombrada comenzando con new_module_ (el nombre del módulo más un guión bajo) y cualquier constante que usted defina debe comenzar con NEW_MODULE_.
- ✓ config.html - (opcional) un formulario para ajustar las preferencias globales del módulo.

En estos ficheros están presentes las funciones que comunican al módulo con la plataforma y las que garantizan su funcionamiento, gestionando las peticiones realizadas a las tablas que contienen la información y los recursos externos a la base de datos, para crear o mostrar las instancias de estos módulos. Además, puede encontrarse el directorio /lang con las especificaciones del lenguaje. (Moodle.org)

2.8 Estilo de Código de MOODLE.

Debido a que MOODLE es un proyecto colaborativo con una extensa comunidad de desarrolladores que aportan a su funcionamiento, se necesita que el código insertado se rija por estándares de codificación definidos para el sistema, posibilitando la uniformidad en la estructura y apariencia del código.

Según la documentación oficial de MOODLE, las variables y funciones deben tener nombres en minúsculas con significado en inglés y fácil de leer. Las variables que representan arreglos o matrices deben usar nombres en plural. Si el nombre es compuesto por varias palabras, estas deben escribirse juntas. Las constantes deben ser nombradas en mayúsculas y comenzar con el nombre del módulo al que pertenecen, seguido de un guión bajo.

Para nombrar las funciones debe utilizarse como prefijo el nombre del módulo al que pertenecen. Es importante no dejar espacio alguno entre el nombre y la apertura de paréntesis. Las funciones deben estar comentadas, utilizando el formato phpDoc, para explicar el flujo del código, el propósito y las variables significativas.

Los bloques de código deben estar siempre entre llaves, aunque aparezca solo una sentencia y se pueden usar espacios en blanco con bastante libertad, para ganar en claridad del código, aunque debería haber un espacio entre llaves y líneas normales, y ninguno entre llaves y variables o funciones. (Moodle.org)

2.9 Manejo de Roles y Capacidades.

El sistema de roles usado por MOODLE está basado en cuatro pilares fundamentales para gestionar la seguridad:

- ✓ Un rol, que es un identificador del estatus del usuario en un contexto particular.

- ✓ Una habilidad, que es una descripción de una funcionalidad particular de MOODLE. Las habilidades (o capacidades) están asociadas a los roles.
- ✓ Un permiso, que es un valor que se asigna a una capacidad para un rol en particular.
- ✓ Un contexto, que es un área en MOODLE en la cual se pueden asignar roles a los usuarios.
- ✓ Los contextos se organizan de forma jerárquica y sus permisos se transfieren desde los contextos superiores a los inferiores.
- ✓ La jerarquía de contexto se establece de la siguiente forma
- ✓ Contexto de sistema - accesible a través del bloque de administración (nivel superior).
- ✓ Contexto de sitio - accesible a través del bloque de administración (padre => sistema).
- ✓ Contexto de categoría de curso - accesible a través de la página de categorías de cursos (padre => sitio).
- ✓ Contexto de curso - accesible a través del bloque de administración del curso (padre => categoría de curso o sitio).
- ✓ Contexto de módulo - accesible mientras se edita el módulo (padre => curso).
- ✓ Contexto de bloque - accesible mientras el modo de edición está activado (padre => sitio o curso).
- ✓ Contexto de usuario - accesible a través de la pestaña de Roles en el perfil de usuario (padre => sitio). (Moodle.org)

Conclusiones

Se realizó un estudio del modelo de dominio, a partir de este, se generaron los requerimientos funcionales y no funcionales del Módulo de Actividad "Evaluación". Se modeló el diagrama de casos de uso del sistema, y se describieron los casos de uso identificados, para su comprensión. Se expone además el esquema del módulo, el estilo de código, y el manejo de roles y capacidades estandarizados por MOODLE.

Capítulo 3. DISEÑO DEL SISTEMA

Introducción

En este capítulo se realiza el diseño del sistema, teniendo en cuenta las necesidades del usuario. Aquí se definen los diagramas del diseño, además se muestran los diagramas de interacción por cada caso de uso. Este capítulo es la antesala de la fase de implementación según RUP.

3.1 Arquitectura.

Es la organización fundamental de un sistema encarnada en sus componentes, las relaciones entre ellos y el ambiente y los principios que orientan su diseño y evolución. También denominada arquitectura lógica, consiste en un conjunto de patrones y abstracciones coherentes que proporcionan el marco de referencia necesario para guiar la construcción del Software.

En una definición más completa, sería, una vista estructural de alto nivel, que ocurre muy tempranamente en el ciclo de vida y define los estilos o grupos de estilos adecuados para cumplir con los requerimientos no funcionales.

3.2 Patrón de Diseño.

Descripción de clases y objetos comunicándose entre sí adaptada para resolver un problema de diseño general en un contexto particular.

Identifica: Clases, Instancias, Roles, Colaboraciones y la distribución de responsabilidades.

3.3 Patrón Modelo –Vista-Controlador (MVC).

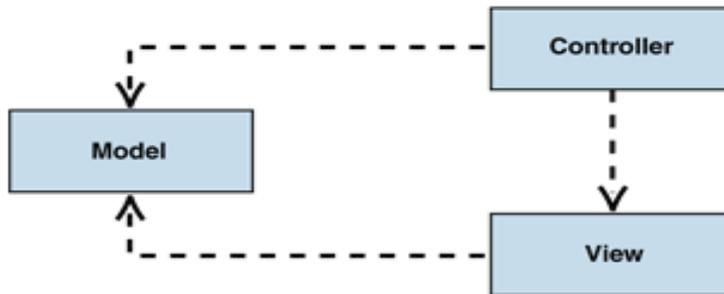


Figura 9 Modelo-Vista-Controlador.

- ✓ **Modelo:** Representación específica del dominio de la información sobre la cual funciona la aplicación. El modelo es otra forma de llamar a la capa de dominio. La lógica de dominio añade significado a los datos. El modelo encapsula los datos y las funcionalidades. El modelo es independiente de cualquier representación de salida y/o comportamiento de entrada.
- ✓ **Vista:** Presenta el modelo en un formato adecuado para interactuar, usualmente un elemento de interfaz de usuario. Muestra la información al usuario. Pueden existir múltiples vistas del modelo. Cada vista tiene asociado un componente controlador.
- ✓ **Controlador:** Responde a eventos, usualmente acciones del usuario e invoca cambios en el modelo y probablemente en la vista. Los eventos son traducidos a solicitudes de servicio (service requests) para el modelo o la vista.

Una gran cantidad de aplicaciones utilizan un mecanismo de almacenamiento persistente (como puede ser una base de datos) para almacenar los datos. El Patrón Modelo – Vista – Controlador no menciona específicamente esta capa de acceso a datos.

3.4 Modelo de Diseño.

En el diseño se modela el sistema y se encuentra la forma para que soporte todos los requisitos. Los propósitos fundamentales en esta fase son:

- ✓ Compresión con profundidad de los aspectos relacionados con los requisitos no funcionales y las restricciones relacionadas con los lenguajes de programación.
- ✓ Creación de una entrada apropiada y un punto de partida para actividades de implementación subsiguientes capturando los requisitos, interfaces y clases.

3.5 Diseño.

3.5.1 Diagramas de Interacción.

Ver Anexo # 2.

3.5.2 Diagramas del Diseño.

Ver Anexo # 3.

3.5.3 Modelo de Datos.

3.5.3.1 Modelo Lógico de Datos.

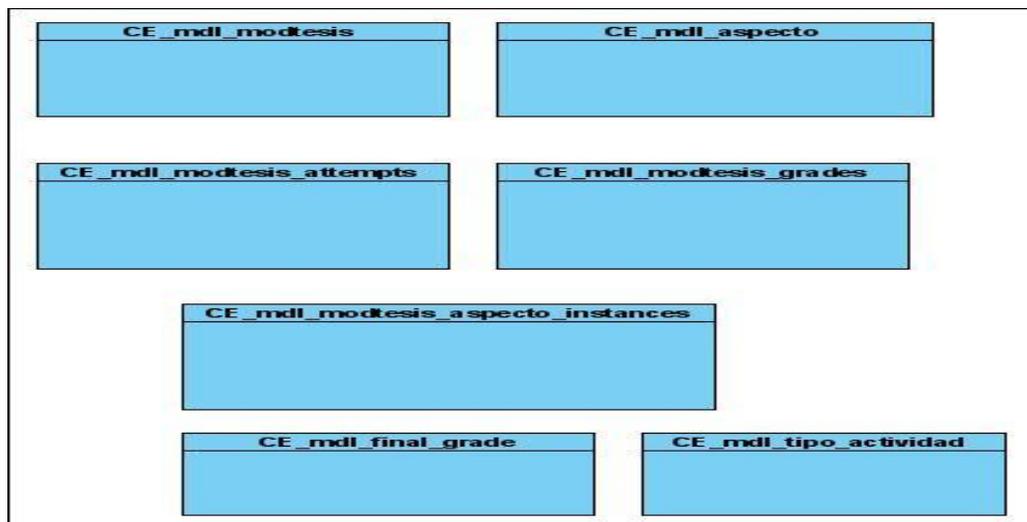


Figura 10 Modelo Lógico de Datos.

2.5.3.2 Modelo Físico de Datos.

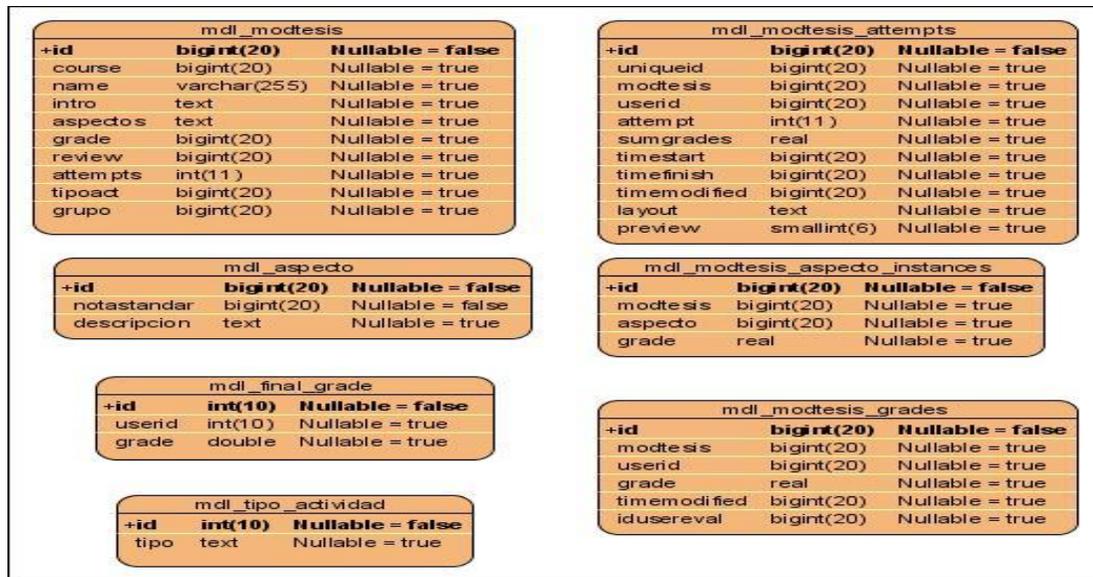


Figura 11 Modelo Físico de Datos.

Conclusiones.

Se generaron los artefactos requeridos para la fase de Diseño, entre los cuales se encuentran los diagramas de estereotipos web, el modelo de datos lógico y físico. Se analizó la arquitectura de la plataforma MOODLE y se definió el patrón de diseño a emplear en el desarrollo del módulo.

Capítulo 4. IMPLEMENTACIÓN Y PRUEBA

Introducción.

En este capítulo se muestran los diagramas generados en la fase de Implementación y los Casos de Prueba realizados (caja negra). Se explica de forma breve y concisa como se instala el módulo desarrollado en la aplicación oficial de nuestra Universidad.

4.1 Diagrama de Despliegue.

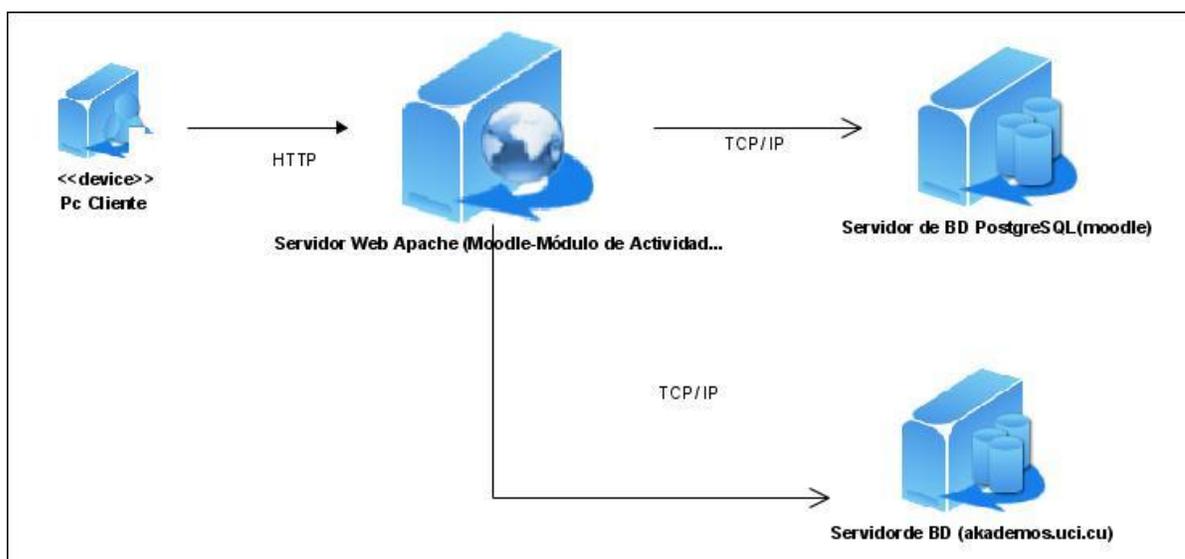


Figura 12 Diagrama de Despliegue.

Los Diagramas de Componentes se pueden observar en el Anexo # 4.

4.2 Prueba.

4.2.1 Casos de Prueba.

Un caso de prueba es un conjunto de entradas de pruebas, condiciones de ejecución y resultados esperados desarrollados para cumplir un objetivo en particular o una función esperada. Es la entidad más simple que siempre se ejecuta como una unidad, desde el comienzo hasta el final.

CAPÍTULO 4: IMPLEMENTACIÓN Y PRUEBA.

Los casos de prueba deben verificar si el producto satisface los requerimientos del usuario, tal y como se describe en las especificaciones de los requerimientos. Y si se comporta como se desea, tal y como se describe en las especificaciones funcionales del diseño.

Var 1: Descripción.

Escenario	Var 1	Respuesta del Sistema	Resultado de la Prueba	Flujo Central
Adicionar aspecto correctamente.	V	Se muestran todos los aspectos adicionados.	Se muestran todos los aspectos adicionados.	<ol style="list-style-type: none">1. Se introduce la descripción del aspecto.2. Pulsar el botón "Adicionar".3. Se listan los aspectos.4. Se almacena el aspecto en la base de datos.
Adicionar aspecto	I	Muestra el mensaje "Debe llenar el campo descripción".	Muestra el mensaje "Debe llenar el campo descripción".	<ol style="list-style-type: none">1. Pulsar el botón "Adicionar".2. No realiza la acción de adicionar.

incorrectamente.	V	Muestra el mensaje “El aspecto ya existe”.	Muestra el mensaje “El aspecto ya existe”.	<ol style="list-style-type: none"> 1. Pulsar el botón “Adicionar”. 2. No realiza la acción de adicionar.
------------------	---	--	--	--

Tabla # 9: Diseño de las pruebas de caja negra para el CU Gestionar Aspecto. Sección Adicionar.

Var1: Descripción.

Escenario	Var 1	Respuesta del Sistema	Resultado de la Prueba	Flujo Central
<p>Editar aspecto Correctamente.</p>	V	<p>Se muestran todos los aspectos adicionados.</p> <p>Se selecciona la opción “editar”. Se va al formulario de editar aspecto.</p>	<p>Se muestran todos los aspectos adicionados.</p> <p>Se selecciona la opción editar. Se va al formulario de editar aspecto.</p>	<ol style="list-style-type: none"> 1. Se introduce la descripción del aspecto. 2.1 Pulsar el botón “Editar”. <ol style="list-style-type: none"> 2.1.1 Se listan los aspectos con los cambios realizados. 2.1.2 Se almacena el cambio en la base de datos. 2.2 Pulsar el botón

				<p>“Cancelar”.</p> <p>2.3 No se realizan cambios</p>
<p>Editar aspecto incorrectamente.</p>	I	<p>Muestra el mensaje "Debe llenar el campo descripción".</p>	<p>Muestra el mensaje "Debe llenar el campo descripción".</p>	<p>1. Pulsar el botón “Adicionar”.</p> <p>2. No realiza la acción de adicionar.</p>
	V	<p>Muestra el mensaje “El aspecto ya existe”.</p>	<p>Muestra el mensaje “El aspecto ya existe”.</p>	<p>1. Pulsar el botón “Adicionar”.</p> <p>2. No realiza la acción de adicionar.</p>

Tabla # 10: Diseño de las pruebas de caja negra para el CU Gestionar Aspecto. Sección Editar.

Escenario	Respuesta del Sistema	Resultado de la Prueba	Flujo Central
<p>Eliminar aspecto.</p>	<p>Se muestran todos los aspectos existentes.</p> <p>Se selecciona la opción "Eliminar". Se pide confirmación de la acción.</p>	<p>Se muestran todos los aspectos existentes.</p> <p>Se selecciona la opción "Eliminar". Se pide confirmación de la acción.</p>	<p>1. Pulsar el botón “Sí”</p> <p>1.2 Se elimina el aspecto seleccionado</p> <p>2. Pulsar el botón “No”</p> <p>2.1 No se elimina el aspecto seleccionado</p>

Tabla # 11: Diseño de las pruebas de caja negra para el CU Gestionar Aspecto. Sección Eliminar.

Var1: Nombre de la Actividad.

Var2: Descripción.

Var3: Aspectos.

Escenario	Var1	Var 2	Var3	Respuesta del Sistema	Resultado de la Prueba	Flujo Central
Adicionar Actividad correctamente.	V	V	V	Se muestran todos los aspectos adicionados.	Se muestran todos los aspectos adicionados.	<ol style="list-style-type: none"> 1. Pulsar el botón "Guardad cambios y regresar al curso". <ol style="list-style-type: none"> 1.1 Se guarda la Actividad en la base de datos 1.2 Se listan todas las actividades existentes. 2. Pulsar el botón "Guardar cambios y mostrar". <ol style="list-style-type: none"> 2.1 Se muestra en un formulario la descripción de la Actividad. 3. Pulsar el botón "Cancelar".

CAPÍTULO 4: IMPLEMENTACIÓN Y PRUEBA.

						3.1 No se adiciona la Actividad.
Adicionar Actividad incorrectamente.	I	V	V	Muestra el mensaje " Debe suministrar un valor aquí".	Muestra el mensaje " Debe suministrar un valor aquí".	<p>1. Pulsar el botón "Guardar cambios y regresar al curso".</p> <p>1.2. No realiza la acción.</p> <p>2. Pulsar el botón "Guardar cambios y mostrar".</p> <p>2.1 No realiza la acción.</p>
	V	I	V	Muestra el mensaje "Obligatorio".	Muestra el mensaje "Obligatorio".	<p>Pulsar el botón "Guardar cambios y regresar al curso".</p> <p>1.2. No realiza la acción.</p> <p>2. Pulsar el botón "Guardar cambios y mostrar".</p> <p>2.1 No realiza la acción.</p>
	V	V	I	Muestra el mensaje "Debe seleccionar al menos un aspecto".	Muestra el mensaje "Debe seleccionar al menos un aspecto".	<p>Pulsar el botón "Guardar cambios y regresar al curso".</p> <p>1.2. No realiza la acción.</p> <p>2. Pulsar el botón "Guardar cambios y mostrar".</p>

CAPÍTULO 4: IMPLEMENTACIÓN Y PRUEBA.

						2.1 No realiza la acción.
--	--	--	--	--	--	---------------------------

Tabla 12: Diseño de las pruebas de caja negra para el CU: Gestionar Actividad de Auto y/o Co-evaluación. Sección Adicionar.

Var1: Nombre de la Actividad.

Var2: Descripción.

Var3: Aspectos.

Escenario	Var1	Var 2	Var3	Respuesta del Sistema	Resultado de la Prueba	Flujo Central
Editar Actividad correctamente.	V	V	V	Se muestran todos los aspectos adicionados	Se muestran todos los aspectos adicionados	1. Pulsar el botón “Guardar cambios y regresar al curso”. 1.1 Se guarda la Actividad en la base de datos 1.2 Se listan todas las actividades existentes. 2. Pulsar el botón “Guardar cambios y mostrar”. 2.1 Se muestra en un formulario la descripción de la Actividad.

						3. Pulsar el botón "Cancelar" 3.1 No se adiciona la Actividad
Editar Actividad incorrectamente.	I	V	V	Muestra el mensaje "Debe suministrar un valor aquí".	Muestra el mensaje "Debe suministrar un valor aquí".	1. Pulsar el botón "Guardar cambios y regresar al curso". 1.2. No realiza la acción. 2. Pulsar el botón "Guardar cambios y mostrar". 2.1 No realiza la acción.
	V	I	V	Muestra el mensaje "Obligatorio".	Muestra el mensaje "Obligatorio".	Pulsar el botón "Guardar cambios y regresar al curso". 1.2. No realiza la acción. 2. Pulsar el botón "Guardar cambios y mostrar". 2.1 No realiza la acción.
	V	V	I	Muestra el mensaje "Debe seleccionar al menos un aspecto".	Muestra el mensaje "Debe seleccionar al menos un aspecto".	Pulsar el botón "Guardar cambios y regresar al curso". 1.2. No realiza la acción. 2. Pulsar el botón "Guardar cambios y mostrar". 2.1 No realiza la acción.

Tabla # 13: Diseño de las pruebas de caja negra para el CU: Gestionar Actividad de Auto y/o Co-evaluación. Sección Editar.

Escenario	Respuesta del Sistema	Resultado de la Prueba	Flujo Central
Eliminar aspecto.	Se muestran todos las Actividades existentes. Se selecciona la opción "Eliminar". Se pide confirmación de la acción.	Se muestran todos los aspectos existentes. Se selecciona la opción "Eliminar". Se pide confirmación de la acción.	1. Pulsar el botón "Sí" 1.2 Se elimina la Actividad seleccionada. 2. Pulsar el botón "No" 2.1 No se elimina la Actividad seleccionada.

Tabla # 14: Diseño de las pruebas de caja negra para el CU: Gestionar Actividad de Auto y/o Co-evaluación. Sección Eliminar.

Var1: Estudiante a evaluar.

Var2: Evaluación.

Escenario	Var 1	Var 2	Respuesta del Sistema	Resultado de la Prueba	Flujo Central
Realizar Actividad correctamente.	V	V	Se guardan los datos de la Actividad y se prosigue a seguir evaluando a sus	Se guardan los datos de la Actividad y se prosigue a seguir evaluando a	1. Pulsar el botón "Enviar todo y terminar"

			compañeros (Si queda alguno sin evaluar).	sus compañeros (Si queda alguno sin evaluar).	
Editar aspecto incorrectamente	V	I	Muestra el mensaje "Debe evaluar todos los aspectos".	Muestra el mensaje "Debe evaluar todos los aspectos".	1. Pulsar el botón "Enviar todo y terminar"
	I	V	Muestra el mensaje "Debe seleccionar un estudiante".	Muestra el mensaje "Debe seleccionar un estudiante".	1. Pulsar el botón "Enviar todo y terminar"

Tabla # 15: Diseño de las pruebas de caja negra para el CU: Realizar Actividad de Auto y/ Co-evaluación.

4.3 Pasos para Instalar el Módulo de Actividad "Evaluación".

- ✓ Copiar el Módulo de Actividad "Evaluación" en la carpeta /mod de la instalación de MOODLE.
- ✓ Acceder a la plataforma como Administrador del Sistema.
- ✓ Ir a la Administración del Sitio.
- ✓ Dar clic en el enlace Notificaciones.
- ✓ Esperar que se instale el fichero db/install.xml del Módulo de Actividad "Evaluación", el cual contiene las tablas que necesita el módulo para su funcionamiento, las cuales se generan en el Servidor de Base de Datos.
- ✓ Clic en el botón continuar y estará listo para su uso.

Conclusiones

Se modeló el diagrama de Despliegue y los de Componentes. Se les diseñó los Casos de Prueba (caja negra) a los casos de uso donde se gestiona información proveniente de los usuarios y que está expensa a errores. Se especificaron los pasos necesarios para incorporar el Módulo de Actividad “Evaluación” de manera sencilla.

Capítulo 5. FACTIBILIDAD DEL SISTEMA

Introducción.

En el presente Capítulo se lleva a cabo la estimación basada en el método de estimación por Puntos de Casos de Usos, que permite al finalizar el análisis conocer si es factible o no llevar a cabo o no la realización del proyecto. Es de gran importancia para realizar un estudio previo del período de duración del proyecto que se va a llevar a cabo, así el equipo de proyecto puede organizar una planificación óptima de las horas laborales que deben ser cumplidas para terminar el desarrollo del proyecto en tiempo y con calidad.

5.1 Método de estimación por Puntos de Casos de Uso.

La estimación mediante el análisis de Puntos de Casos de Uso es un método propuesto originalmente por Gustav Karner de Objectory AB, y posteriormente refinado por muchos otros autores.

Se trata de un método de estimación del tiempo de desarrollo de un proyecto mediante la asignación de "pesos" a un cierto número de factores que lo afectan, para finalmente, contabilizar el tiempo total estimado para el proyecto a partir de esos factores. (CP2 Estimación)

A continuación se explican los pasos a seguir para la aplicación del método de estimación Puntos por Casos de Uso

5.1.1 Cálculo de Puntos de Casos de Uso sin ajustar.

Se calcula a partir de la siguiente ecuación:

$$\text{➤ } \mathbf{UUCP = UAW + UUCW}$$

Donde:

- ✓ **UUCP:** Puntos de Casos de Uso sin ajustar
- ✓ **UAW:** Factor de Peso de los Actores sin ajustar
- ✓ **UUCW:** Factor de Peso de los Casos de Uso sin ajustar

5.1.1.1 Factor de Peso de los Actores sin ajustar (UAW).

Este valor se calcula mediante un análisis de la cantidad de Actores presentes en el sistema y la complejidad de cada uno de ellos. La complejidad de los Actores se establece teniendo en cuenta en primer lugar si se trata de una persona o de otro sistema, y en segundo lugar, la forma en la que el actor interactúa con el sistema.

Los criterios se muestran en la siguiente tabla:

Tipo de Actor	Descripción	Factor de Peso
Simple	Otro sistema que interactúa con el sistema a desarrollar mediante una interfaz de programación (API, Application Programming Interface)	0*1
Medio	Otro sistema que interactúa con el sistema a desarrollar mediante un protocolo o una interfaz basada en texto.	0*2
Complejo	Una persona que interactúa con el sistema mediante una interfaz gráfica.	5*3
Total		15

Tabla # 16: Cálculo del factor de peso de los actores sin ajustar.

Ecuación para calcular el **UAW**:

$$UAW = \sum(\text{actores} * \text{Peso})$$

$$UAW = 15$$

5.1.1.2 Factor Peso de los Casos de Uso sin ajustar (UUCW).

Este valor se calcula mediante un análisis de la cantidad de Casos de Uso presentes en el sistema y la complejidad de cada uno de ellos. La complejidad de los Casos de Uso se establece teniendo en cuenta la cantidad de transacciones efectuadas en el mismo, donde una *transacción* se entiende como una secuencia de actividades atómica, es decir, se efectúa la secuencia de actividades completa, o no se efectúa ninguna de las actividades de la secuencia y está

representada por uno o más pasos del flujo de eventos principal del Caso de Uso, pudiendo existir más de una transacción dentro del mismo Caso de Uso.

Los criterios se muestran en la siguiente tabla:

No	Nombre de Caso de Uso	Cantidad transacciones	Tipo
1	Realizar Actividad de Auto y/o Co-evaluación.	9	Complejo
2	Evaluar Actividad.	2	Simple
3	Insertar Respuestas.	3	Simple
4	Actualizar Libro de Calificación.	2	Simple
5	Agregar Módulo de Actividad "Evaluación".	10	Complejo
6	Gestionar Instancia.	14	Complejo
7	Gestionar Actividad de Auto y/o Co-evaluación.	30	Complejo
8	Gestionar Aspecto.	21	Complejo
9	Mostrar Resultados en el Libro de Calificación.	2	Simple

Tabla # 17: Cantidad de transacciones por Casos de Uso.

Tipo	Descripción	Peso	Cant * peso
Simple	El Caso de Uso contiene de 1 a 3 transacciones.	5	4*5
Medio	El Caso de Uso contiene de 4 a 7 transacciones.	10	0*10
Complejo	El Caso de Uso contiene más de 8 transacciones.	15	5*15
Total:			95

Tabla # 18: Cálculo del factor de peso de los casos de uso sin ajustar.

La cantidad de transacciones se determina a partir de la descripción textual de los casos de uso. Entre más detallada esté la descripción textual, más transacciones se pueden encontrar y la estimación será más exacta.

Ecuación para calcular el **UUCW**:

$$\mathbf{UUCW = \sum CU * Peso}$$

$$\mathbf{UUCW = 95}$$

Entonces ya tenemos UUCP.

Luego:

$$\mathbf{UUCP = UAW + UUCW}$$

$$\mathbf{UUCP = 15 + 95}$$

$$\mathbf{UUCP = 110}$$

5.1.2 Cálculo de Puntos de Casos de Uso ajustados.

$$\mathbf{UCP = UUCP * TCF * EF}$$

Donde:

UCP: Puntos de casos de uso ajustados.

UUCP: Puntos de casos de uso sin ajustar.

TCF: Factor de complejidad técnica.

EF: Factor de ambiente.

5.1.2.1 Factor de complejidad técnica (TCF).

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.

Significado de los valores:

0: No presente o sin influencia.

1: Influencia incidental o presencia incidental.

2: Influencia moderada o presencia moderada.

3: Influencia media o presencia media.

4: Influencia significativa o presencia significativa.

5: Fuerte influencia o fuerte presencia.

Factor	Descripción	Peso	Valor	Comentario	Σ (Peso _i * Valor _i)
T1	Sistema distribuido.	2	0	El sistema es distribuido.	0
T2	Objetivos de performance o tiempo de respuesta.	1	5	Se requiere que el sistema tenga un buen rendimiento y una respuesta rápida.	5
T3	Eficiencia del usuario final.	1	2	No hay restricciones de eficiencia.	2
T4	Procesamiento interno complejo.	1	3	No hay cálculos Complejos.	3
T5	El código debe ser reutilizable.	1	5	Es reutilizable.	5
T6	Facilidad de instalación.	0.5	5	El sistema debe ser fácil de instalar.	2.5
T7	Facilidad de uso.	0.5	5	Debe ser un sistema amigable.	2.5
T8	Portabilidad.	2	3	Se requiere que el sistema sea portable.	6
T9	Facilidad de cambio.	1	5	Se requiere que sea un sistema flexible ante	5

				cambios.	
T10	Concurrencia.	1	3	Si hay concurrencia.	3
T11	Incluye objetivos especiales de seguridad.	1	5	El sistema gestiona información cuya confidencialidad es de carácter limitado.	5
T12	Provee acceso directo a terceras partes.	1	3	Provee acceso directo a terceras partes.	3
T13	Se requieren facilidades especiales de entrenamiento a los usuarios.	1	0	No se requieren facilidades especiales de entrenamiento a los usuarios.	0
Total:					42

Tabla # 19: Cuantificación de factores que determinan la complejidad técnica del sistema

Para calcular Factor de Complejidad Técnica (TCF):

$$TCF = 0.6 + 0.01 * \Sigma (\text{Peso}_i * \text{Valor}_i) \text{ (Donde Valor es un número del 0 al 5)}$$

$$TCF = 0.6 + 0.01 * 42$$

$$TCF = 1.02$$

5.1.2.2 Factor de Ambiente (EF).

Las habilidades y el entrenamiento del grupo involucrado en el desarrollo tienen un gran impacto en las estimaciones de tiempo. Estos factores son los que se contemplan en el cálculo del Factor de ambiente. El cálculo del mismo es similar al cálculo del Factor de complejidad técnica, es decir, se trata de un conjunto de factores que se cuantifican con valores de 0 a 5.

Factor:	Descripción:	Peso:	Valor:	Comentario:	Σ (Peso_i * Valor_i):
E1	Familiaridad con el modelo de proyecto utilizado.	1.5	1	El grupo no está familiarizado con el modelo de proyecto.	1.5
E2	Experiencia en la aplicación.	0.5	1	Experiencia media en trabajo con ese tipo de aplicaciones.	0.5
E3	Experiencia en orientación a objetos.	1	5	La mayoría del grupo ha programado Orientado a Objetos.	5
E4	Capacidad del analista líder.	0.5	3	El analista es un estudiante con conocimientos adquiridos en clases.	1.5
E5	Motivación.	1	5	El equipo se siente motivado.	5
E6	Estabilidad de los requerimientos.	2	3	Se esperan cambios.	6
E7	Personal part-time.	-1	4	Todo el equipo es full-time.	-4
E8	Dificultad del lenguaje de programación.	-1	3	Se usará el lenguaje PHP orientado a objetos, estudiado. No es difícil de aprender.	-3
Total:					12.5

Tabla # 20: Factores habilidades y entrenamiento.

Para Calcular EF:

$$EF = 1.4 - 0.03 * \Sigma (\text{Peso}_i * \text{Valor}_i) \text{ (Donde Valor es un número del 0 al 5)}$$

$$EF = 1.4 - 0.03 * 12.5$$

$$EF = 1.025$$

Luego:

$$UCP = UUCP * TCF * EF$$

$$UCP = 110 * 1.02 * 1.025$$

$$UCP = 115.005$$

5.1.2.3 Cálculo de Esfuerzo.

El esfuerzo en horas-hombre viene dado por:

$$E = UCP * CF$$

Donde:

E: esfuerzo estimado en horas-hombre.

UCP: Puntos de Casos de Uso ajustados.

CF: Factor de Conversión.

Este método proporciona una estimación del esfuerzo en horas-hombre contemplando sólo el desarrollo de la funcionalidad especificada en los casos de uso.

Para una estimación más completa de la duración total del proyecto, hay que agregar a la estimación del esfuerzo obtenida por los puntos de casos de uso, las estimaciones de esfuerzo de las demás actividades relacionadas con el desarrollo de software.

Para calcular Factor de Conversión (**CF**):

CF = 20 horas-hombre (si Total EF ≤ 2)

CF = 28 horas-hombre (si Total EF = 3 ó Total EF = 4)

CF = abandonar o cambiar proyecto (si Total EF ≥ 5)

Total EF = Cant. EF < 3 (entre E1 – E6) + Cant. EF > 3 (entre E7 – E8)

Como:

Total EF = 2 + 0

Total EF = 2

CF = 20 horas-hombre (porque Total EF = 2)

Luego:

$$E = 115.005 * 20 \text{ horas-hombre}$$

$$E = 2300.1 \text{ horas-hombre}$$

5.1.3 Distribución del Esfuerzo entre las diferentes actividades del módulo.

Actividad	% esfuerzo	Valor esfuerzo
Análisis	10	575,025
Diseño	20	1150,05
Implementación	40	2300.1
Prueba	15	862,5375
Sobrecarga	15	862,5375
Total	100	5750,25

Tabla # 21: Distribución del esfuerzo estimado entre los flujos de trabajo de RUP.

5.2 Beneficios Tangibles e Intangibles.

El desarrollo del Módulo de Actividad “Evaluación” para MOODLE aporta para la Universidad de las Ciencias Informáticas un considerable beneficio en la esfera docente, con dicho módulo se contribuye al desarrollo del PEA facilitando de esta manera el trabajo a los profesores y una mayor participación de los estudiantes en su evaluación y la de sus compañeros.

5.3 Análisis de costos y beneficios.

Salario medio de la fuerza de trabajo: \$100.

Cantidad de trabajadores: 2.

Como se tiene 5750,25 horas-hombre según lo estimado, se divide este valor entre 12 horas (cantidad real de horas que le dedica cada trabajador al módulo) para calcular la cantidad de días de trabajo-hombre que se necesitan para realizar el módulo. Al realizar esta operación matemática se obtiene que el módulo de planificación tenga 479 días de trabajo-hombre. Este valor se divide entre la cantidad de trabajadores para obtener la cantidad de días de trabajo que tiene cada trabajador para el desarrollo del módulo. De este resultado se concluye que con 2 trabajadores que trabajan 12 horas diarias, el módulo se terminaría en 239,5 días. Debido a que sólo se trabajan 6

días a la semana, se debe sumar a esta estimación 14 días más, que se corresponden con los 14 domingos que están dentro de esa planificación y que no se trabajan. Por tanto, finalmente para realizar el módulo se necesitan aproximadamente 253,5 días (9 meses y medio). El costo final del módulo es:

Costo total del módulo = salario medio * cantidad de trabajadores * duración del proyecto (meses)

Costo total del módulo = \$ 100 * 2 * 9 = \$ 1800.

El desarrollo de software en el mundo actual es muy costoso, por los altos precios que implantan las grandes compañías en este sentido. Las herramientas que se utilizan en la elaboración del sistema son totalmente libres, no es necesario el pago de licencias, por tanto, no se incurre en gasto alguno en la utilización de las mismas.

Debido a lo que se refleja anteriormente, se concluye que el desarrollo del Módulo de Actividad “Evaluación” para MOODLE, es factible.

Conclusiones.

En este capítulo se realizó un estudio de la factibilidad del sistema que se presenta. Esto permite llegar a la conclusión de que resulta factible desarrollar el Módulo de Actividad “Evaluación” para MOODLE. Se analizaron los beneficios tangibles e intangibles que proporciona la incorporación del módulo en el EVA utilizado por la Universidad de las Ciencias Informáticas.

Conclusiones Generales

En este trabajo se realizó un análisis detallado de las funcionalidades de MOODLE, así como un estudio del empleo de las plataformas e-learning utilizadas nacional e internacionalmente. Cumpliéndose los objetivos tanto generales como específicos, pues se logró incorporar un nuevo módulo de actividad para MOODLE que permite realizar a los estudiantes una Actividad de Auto y/o Co-evaluación; el cual contribuye a una evaluación formativa centrada en el aprendizaje dentro de la Universidad de Ciencias Informáticas.

RECOMENDACIONES

- ✓ Se recomienda hacer un nuevo diseño de la base de datos de MOODLE, ya que esta no cuenta con llaves foráneas, por lo tanto identificar las relaciones se hace un trabajo complejo.
- ✓ Diseñar una nueva interfaz para el Libro de Calificación.

Bibliografía

Moodle.org. (s.f.). Recuperado el 1 de junio de 2010, de Moodle.org: <http://moodle.org/stats/>

Clikear.com. (s.f.). Recuperado el 25 de enero de 2010, de Clikear.com: <http://www.clikear.com/manuales/uml/introduccion.aspx>.

Arenas, M. I. (s.f.). *Geneura*. Recuperado el 8 de febrero de 2010, de Geneura: <http://geneura.ugr.es/~maribel/xml/introduccion/index.shtml>.

e-ABC. (s.f.). Recuperado el 25 de noviembre de 2009, de e-ABC: <http://www.e-abclearning.com/content/view/6/56/>.

FACILNET. (s.f.). Recuperado el 7 de febrero de 2010, de FACILNET: <http://www.facilnet.net/matriz/web2/apache.html>.

Fedapas.org. (s.f.). Recuperado el 13 de enero de 2010, de Fedapas.org: <http://www.fedapas.org/ATutor/about.php>.

Hooping.net. (s.f.). Recuperado el 8 de febrero de 2010, de Hooping.net: <http://www.hooping.net/faq.html.aspx>.

HTMELPOINT.com. (s.f.). Recuperado el 9 de febrero de 2010, de HTMELPOINT.com: http://www.htmlpoint.com/php/guida/php_02.htm.

Merlin. (s.f.). Recuperado el 14 de enero de 2010, de Merlin: <http://merlin.germinus.com/objetivos>.

Bibliografía.

Milenium. (s.f.). Recuperado el 25 de noviembre de 2009, de Milenium: <http://www.informaticamilenium.com.mx/paginas/mn/articulo78.htm>.

Moodle.org. (s.f.). Recuperado el 13 de enero de 2010, de Moodle.org: <http://moodle.org/mod/data/view.php?id=6009>.

Moodle.org. (s.f.). Recuperado el 24 de febrero de 2010, de Moodle.org: [http://docs.moodle.org/es/M%C3%B3dulos de actividades %28desarrollador%29](http://docs.moodle.org/es/M%C3%B3dulos_de_actividades_%28desarrollador%29)

Moodle.org. (s.f.). Recuperado el 2 de marzo de 2010, de Moodle.org: [http://docs.moodle.org/es/Manual de Estilo de C%C3%B3digo](http://docs.moodle.org/es/Manual_de_Estilo_de_C%C3%B3digo)

(2005). aprenDIST. En I. Y. Oliveros Guntín, D. A. Rosete Suárez., & M. J. Peña Sepúlveda, *Plataforma de Teleformación aprenDIST*. (pág. 5). Ciudad de La Habana: MES.

Pérez, J. E. (s.f.). *librosweb.es*. Recuperado el 9 de febrero de 2010, de librosweb.es: <http://www.librosweb.es/javascript/Capítulo1.html>

Ramos, R., Sepúlveda, J. C., & La O, A. (2002). *Mundicampus, un modelo de formación a distancia a través de Web*. Recuperado el 20 de enero de 2010, de Mundicampus, un modelo de formación a distancia a través de Web :

http://webcache.googleusercontent.com/search?q=cache:n6GmwoAs21AJ:espejos.unesco.org.uy/simplac_2002/Ponencias/Inforededu/IE105%2520InforededuAmnia.doc+mundicampus&cd=3&hl=es&ct=clnk&gl=cu

Rodríguez Cáceres, A. O., & Torres Acosta, R. (s.f.). *Medcampus: un proyecto de plataforma para la educación a distancia*. Recuperado el 22 de enero de 2010, de Medcampus: un proyecto de plataforma para la educación a distancia: http://bvs.sld.cu/revistas/aci/vol12_3_04/aci02304.htm

Sanchez, M. A. (7 de junio de 2004.). *informatizate*. Recuperado el 23 de enero de 2010, de informatizate: [http://www.informatizate.net/articulos/metodologias de desarrollo de software 07062004.html](http://www.informatizate.net/articulos/metodologias_de_desarrollo_de_software_07062004.html).

Bibliografía.

Sancho, J. B. (2007). Moodle 1.8 Manual del Profesor. En J. B. Sancho, *MANUAL DE CONSULTA PARA EL PROFESORADO (Versión 1.8)* (pág. 9). Getafe: IES Satafi.

Sancho, J. B. (2007). Moodle 1.8 Manual del Profesor. En J. B. Sancho, *MANUAL DE CONSULTA PARA EL PROFESORADO (Versión 1.8)* (pág. 11). Getafe: IES Satafi.

Sancho, J. B. (2007). Moodle 1.8 Manual del Profesor. En J. B. Sancho, *MANUAL DE CONSULTA PARA EL PROFESORADO (Versión 1.8)* (pág. 13). Getafe: IES Satafi.

Sancho, J. B. (2007). Moodle 1.8 Manual del Profesor. En J. B. Sancho, *MANUAL DE CONSULTA PARA EL PROFESORADO (Versión 1.8)* (pág. 10). Getafe: IES Satafi.

SEPAD. (s.f.). Recuperado el 14 de enero de 2010, de SEPAD: <http://www.cdict.uclv.edu.cu/cat-servicios-desde-las-facultades>.

SQL Manage.net. (s.f.). Recuperado el 2 de febrero de 2010, de SQL Manage.net: <http://sqlmanager.net/products/postgresql/manager&ei=mf73S6StDIGglAeAkMH4Cg&sa=X&oi=translate&ct=result&resnum=1&ved=0CBgQ7gEwAA&prev=/search%3Fq%3Dems%2Bsql%2Bmanager%2Bfor%2Bpostgresql%2B%252B%2Bfeatures%26hl%3Des> .

TARINGA. (s.f.). Recuperado el 7 de febrero de 2010, de TARINGA: http://www.taringa.net/posts/downloads/2063235/Zend-Studio-for-Eclipse-6_1_0-%28con-Keygen-incluido%29+-Regalo.html.

Temas de Investigación. (s.f.). Recuperado el 13 de enero de 2010, de Temas de Investigación: <http://internet-educativa.pbworks.com/Docebo> .

Visual Paradigm. (s.f.). Recuperado el 1 de febrero de 2010, de Visual Paradigm: <http://www.visual-paradigm.com/product/vpum/> .

Bibliografía.

World Wide Web Consortium (W3C) - España. (s.f.). Recuperado el 7 de febrero de 2010, de World Wide Web Consortium (W3C) - España: <http://www.w3c.es/divulgacion/guiasbreves/HojasEstilo>.

ANEXOS

Anexo # 1. Descripción textual de los casos de uso del sistema.

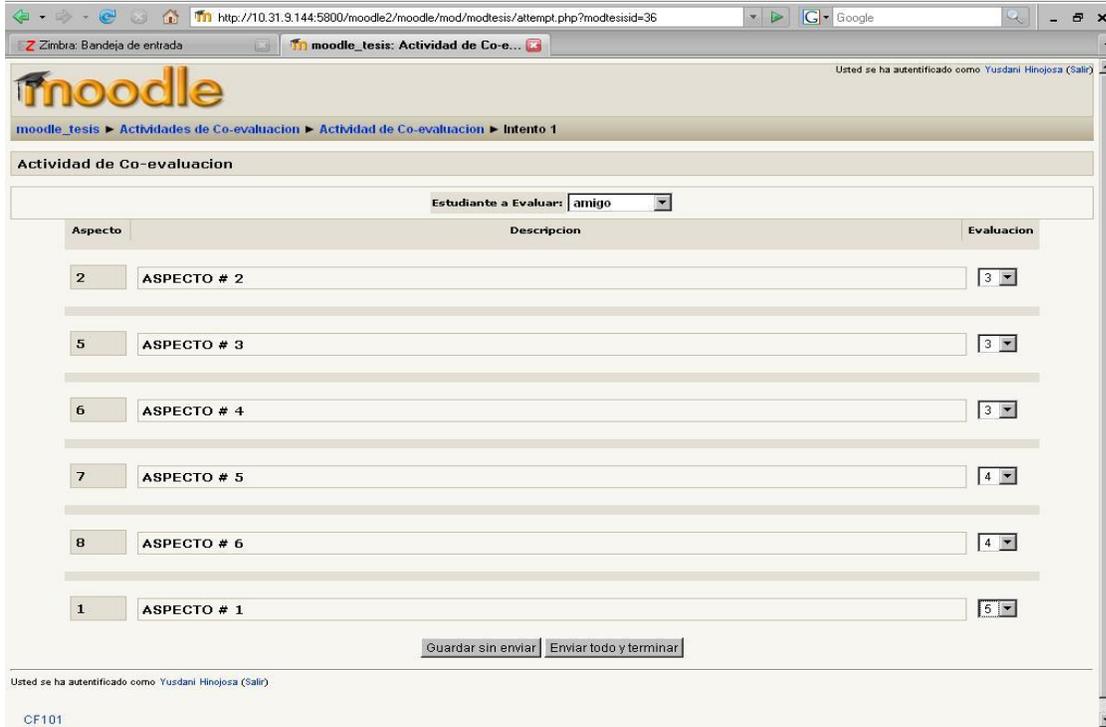
Caso de Uso:	Realizar Actividad de Auto y/o Co-evaluación.
Actores:	Estudiante.
Resumen:	El estudiante elige realizar la Actividad de Auto y/o Co-evaluación, esta se visualiza, la resuelve, guarda las respuestas y si ha terminado las envía.
Precondiciones:	Estar autenticado en la plataforma.
Referencias	
Prioridad	

Flujo Normal de Eventos

Sección “”

Acción del Actor	Respuesta del Sistema
El estudiante selecciona la Actividad de Auto y/o Co-evaluación a realizar. El estudiante realiza la Actividad de Auto y/o Co-evaluación evaluando al estudiante seleccionado. Envía los resultados y termina.	Verifica si el usuario autenticado ha realizado algún intento de la Actividad de Auto y/o Co-evaluación. Muestra la Actividad de Auto y/o Co-evaluación a realizar correspondiente al grupo del usuario con los siguientes campos: Estudiante a Evaluar. Evaluación (la cantidad depende de los aspectos asociados a la Actividad de Auto y/o Co-evaluación). Invocar caso de uso Evaluar Actividad.

Prototipo de Interfaz



Flujos Alternos

Acción del Actor	Respuesta del Sistema
2.1. a Guarda los resultados sin enviar. 2.1. a.1 Se repite el flujo normal de eventos a partir del #2.	

Caso de Uso:	Evaluar Actividad.
Actores:	Estudiante.
Resumen:	El sistema evalúa la actividad y muestra los resultados en el Libro de Calificaciones
Precondiciones:	Estar autenticado en la plataforma.
Referencias	
Prioridad	

Flujo Normal de Eventos

Sección “”

Acción del Actor	Respuesta del Sistema
	<ol style="list-style-type: none"> 1. El Módulo de Actividad “Evaluación” evalúa la Actividad. 2. Guarda las evaluaciones de la Actividad realizada. <ol style="list-style-type: none"> 2.1 Invocar caso de uso Realizar Actividad de Auto y/o Co-evaluación (realiza esta acción en dependencia de la cantidad de estudiantes del grupo asociado a la Actividad y del tipo de Actividad).

Prototipo de Interfaz

Flujos Alternos

Acción del Actor	Respuesta del Sistema
.	

Caso de Uso:	Agregar Módulo de Actividad “Evaluación”.
Actores:	
Resumen:	Se implementa todo lo relacionado con el manejo de las instancias en el Módulo de Actividad “Evaluación”.
Precondiciones:	
Referencias	
Prioridad	

Flujo Normal de Eventos

Sección “”

Acción del Actor	Respuesta del Sistema
	Invocar Caso de Uso Instanciar Formulario. Instanciar Caso de Uso Definir Permisos. Invocar Caso de Uso Instanciar Aspectos. Invocar Caso de Uso Instanciar Actividad de Auto y/o Co-evaluación.

Prototipo de Interfaz

Flujos Alternos

Acción del Actor	Respuesta del Sistema

Caso de Uso:	Instanciar Formulario.
Actores:	
Resumen:	Se definen todos los elementos que contendrá el formulario para el Módulo de Actividad “Evaluación” como módulo de actividad.

Precondiciones:	
Referencias	
Prioridad	
Flujo Normal de Eventos	
Sección “”	
Acción del Actor	Respuesta del Sistema
	<p>Se definen los elementos que conforman el formulario del Módulo de Actividad “Evaluación”:</p> <ul style="list-style-type: none"> *name (tipo texto). *intro (tipo htmleditor). *aspecto (tipo checkbox). *grupos (tipo select).
<i>Prototipo de Interfaz</i>	
Flujos Alternos	
Acción del Actor	Respuesta del Sistema

Caso de Uso:	Definir Permisos.
Actores:	
Resumen:	Aquí es donde se define el acceso de los usuarios, se crean los formularios, se define la librería, se crean las instancias, todo lo referente a la funcionalidad del módulo.
Precondiciones:	
Referencias	

Prioridad	
Flujo Normal de Eventos	
Sección “”	
Acción del Actor	Respuesta del Sistema
	<p>Se definen los siguientes permisos para los diferentes roles que dispone MOODLE y los eventos del Módulo de Actividad “Evaluación”:</p> <ul style="list-style-type: none">Capability name.Riskbitmask.Capttype.Contextlevel.Legacy. <p>Los roles de los usuarios son:</p> <ul style="list-style-type: none">Admin.Editingteacher.Teacher.Student.Guest. <p>Los eventos son los siguientes:</p> <ul style="list-style-type: none">View.Attempt.Submit.Edit.Manage.Viewreports.Preview.Participate.Grade.

Prototipo de Interfaz	
Flujos Alternos	
Acción del Actor	Respuesta del Sistema

Caso de Uso:	Instanciar Actividad de Auto y/o Co-evaluación.
Actores:	
Resumen:	Se implementa todo lo relacionado con el manejo de las instancias de las Actividades de Auto y/o Co-evaluación.
Precondiciones:	
Referencias	
Prioridad	

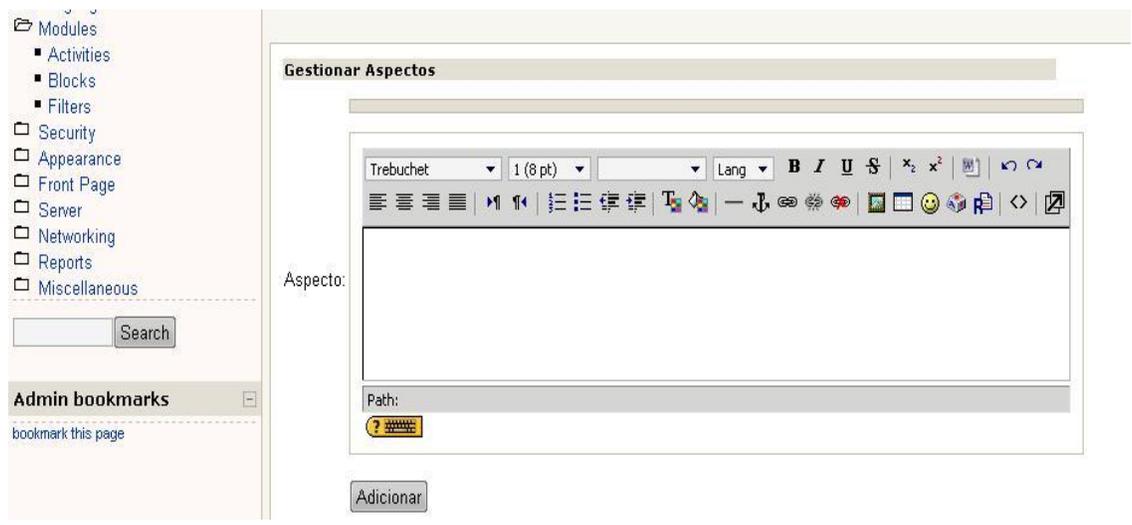
Flujo Normal de Eventos	
Sección “”	
Acción del Actor	Respuesta del Sistema
	Se implementa las funciones estandarizadas por MOODLE para la integración del Módulo de Actividad “Evaluación” a la plataforma así como la utilización de las instancias:

	modtesis_add_instance (). modtesis_delete_instance (). modtesis_update_instance (). modtesis_grades ().
Prototipo de Interfaz	
Flujos Alternos	
Acción del Actor	Respuesta del Sistema

Caso de Uso:	Gestionar Aspectos de la Actividad de Auto y/o Co-evaluación.
Actores:	Administrador del Sistema.
Resumen:	El actor gestiona los aspectos.
Precondiciones:	
Referencias	
Prioridad	
Flujo Normal de Eventos	
Sección "Insertar"	
Acción del Actor	Respuesta del Sistema
Selecciona Configurar Aspecto. Introduce la información solicitada. Adiciona el nuevo Aspecto.	1.1 El sistema pide llenar los siguientes campos: 1.1.2 Aspecto. 3.1 Verifica que información sea válida: 3.1.1 El campo no esté vacío. 3.1.2 El Aspecto ya exista. 3.2 El sistema inserta el nuevo Aspecto.

3.2.1 El sistema muestra todos los Aspectos existentes.

Prototipo de Interfaz



Flujos Alternos

Acción del Actor	Respuesta del Sistema

Sección “Modificar”

Acción del Actor	Respuesta del Sistema
Selecciona un Aspecto a editar. Introduce la información solicitada. Guarda los cambios realizados.	1.1 El sistema pide llenar los siguientes campos: 1.1.2 Aspecto. 3.1 Verifica que información sea válida: 3.1.1 El campo no esté vacío. 3.1.2 El Aspecto ya exista. 3.2 El sistema actualiza los cambios realizados.

Prototipo de Interfaz

Gestionar Aspectos

Trebuchet 1 (8 pt) Lang B I U S x₂ x²

Aspecto:

Path:

Adicionar

- 113 Verifica que los campos no estén vacíos ✖
- 115 El sistema inserta los cambios realizados ✖
- 117 nuevo aspecto ✖
- 114 El sistema pide la confirmación de la acción de eliminar. [editar](#)
- 116 frffadasdjas cjsdcsdcn kscsdcnlasmas ✖

moodle_tesis: You are logged in as Admin User

moodle

moodle_tesis ▶

Editar Aspecto:

Trebuchet 1 (8 pt) Lang B I U S x₂ x²

ASPECTO # 4

Path:

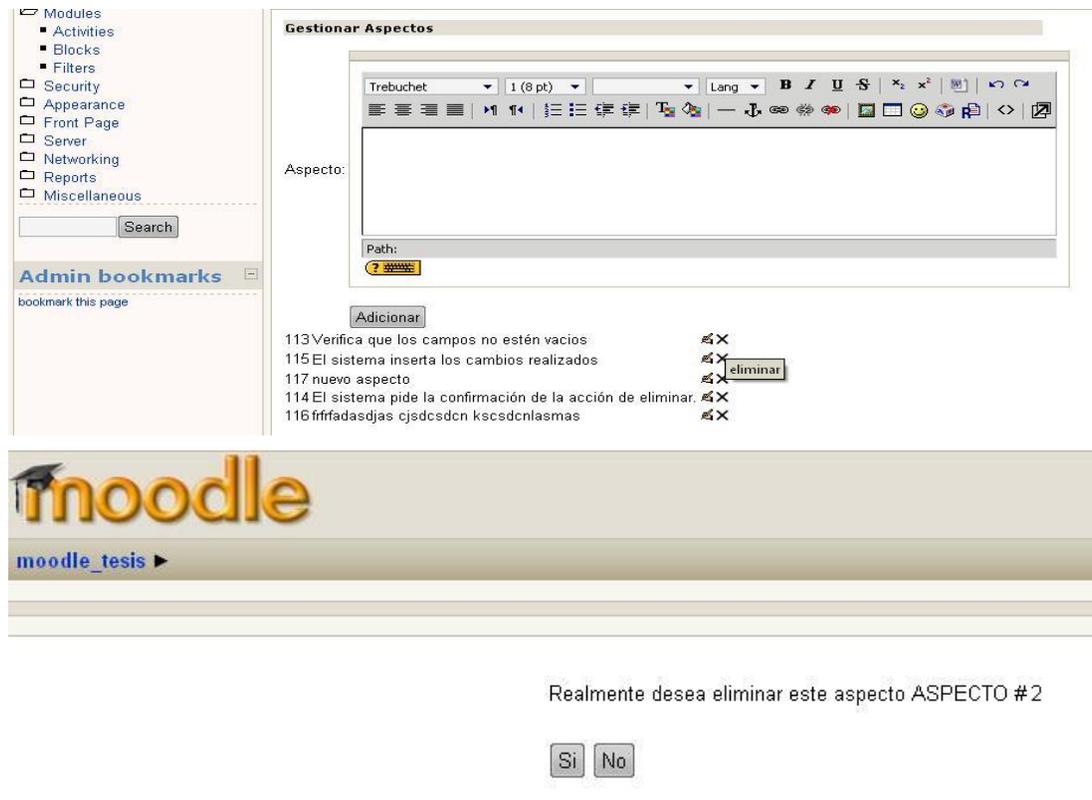
Editar Cancelar

Flujos Alternos

Acción del Actor	Respuesta del Sistema
1.1.2. a El usuario cancela la acción de Editar.	1.1.2.b El sistema no efectúa cambios .
Sección “Eliminar”	
Acción del Actor	Respuesta del Sistema
Selecciona un aspecto a eliminar. Confirma eliminar aspecto.	1.1 El sistema pide la confirmación de la acción de eliminar.

- 1.1.1 Sí
- 1.1.2 No
- 2.1 El sistema elimina el aspecto seleccionado.

Prototipo de Interfaz



Flujos Alternos

Acción del Actor	Respuesta del Sistema
2.1.a El actor no confirma la acción de eliminar el aspecto seleccionado	2.1.b El sistema no realiza ningún cambio.

Sección "Mostrar"

Acción del Actor	Respuesta del Sistema
	El sistema muestra una lista de los Aspectos existentes.

Prototipo de Interfaz

Gestionar Aspectos

Aspecto:

Path:

Adicionar

- 113 Verifica que los campos no estén vacíos
- 115 El sistema inserta los cambios realizados
- 117 nuevo aspecto
- 114 El sistema pide la confirmación de la acción de eliminar.
- 116 frifadasdjas cjsdcscdn kscsdcnlasmas

Flujos Alternos

Acción del Actor	Respuesta del Sistema

Caso de Uso:	Mostrar Resultados en el Libro de Calificación.
Actores:	Profesor.
Resumen:	Los resultados de la Actividad de Auto y/o Co-evaluación son mostrados en el libro de calificación.
Precondiciones:	El caso de uso Realizar Auto y/o Co-evaluación haya terminado.
Referencias	
Prioridad	

Flujo Normal de Eventos

Sección “”

Acción del Actor	Respuesta del Sistema
1. Accede al Libro de Calificación.	La plataforma muestra los resultados de la Actividad de Auto y/o Co-evaluación en el Libro de Calificación.

Prototipo de Interfaz

The screenshot shows the Moodle 'Grades' interface. At the top, there's the Moodle logo and navigation links. Below that, there are buttons for 'View Grades' and 'Set Preferences'. A download section offers options for ODS, Excel, and text formats. The main part of the page is a table titled 'Grades' with columns for 'Student', 'Sam', 'Carla2', 'Yeni', 'Dimmy', 'Alley', 'Ane', 'Chulu', 'Tranis', 'Yer', 'Actividad de Co-evaluacion', 'Total Stats', and 'Student'. The table lists three students: Hinojosa, Yusciani; Hurtado, Rangel; and Reventco, Alexander. All scores are 0%.

Student	Sam	Carla2	Yeni	Dimmy	Alley	Ane	Chulu	Tranis	Yer	Actividad de Co-evaluacion	Total Stats	Student
Sort by Lastname	Raw %	Raw %	Raw %	Raw %	Raw %	Raw %	Raw %	Raw %	Raw %	Raw %	50 Percent	Sort by Lastname
Sort by Firstname	Raw %	Raw %	Raw %	Raw %	Raw %	Raw %	Raw %	Raw %	Raw %	Raw %	Percent	Sort by Firstname
Hinojosa, Yusciani	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	Hinojosa, Yusciani
Hurtado, Rangel	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	Hurtado, Rangel
Reventco, Alexander	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	Reventco, Alexander

Caso de Uso:	Gestionar Actividad de Auto y/o Co-evaluación.
Actores:	Profesor
Resumen:	El Profesor elige crear una nueva Actividad de Auto y/o Co-evaluación, Introduce el nombre de la actividad, los aspectos a evaluar y la asigna a un grupo de estudiantes.
Precondiciones:	Estar autenticado en la plataforma.
Referencias	
Prioridad	

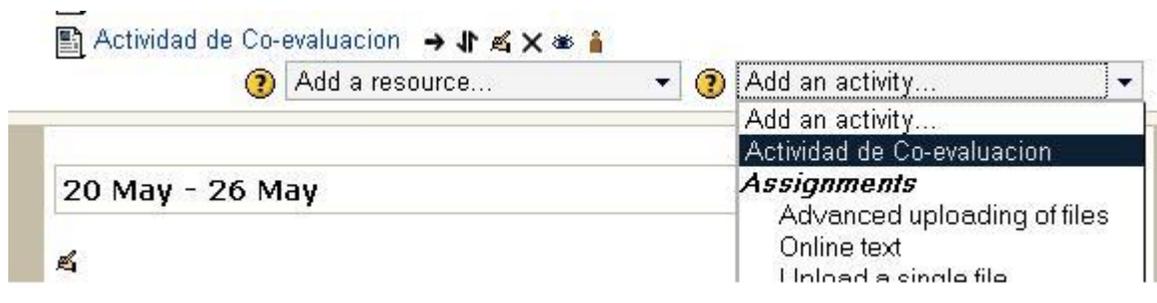
Flujo Normal de Eventos

Sección "Insertar Actividad"

Acción del Actor	Respuesta del Sistema
<p>El Profesor elige crear una nueva Actividad de Auto y/o Co-evaluación.</p> <p>2. El profesor Introduce los datos solicitados</p> <p>Asigna la actividad al grupo de estudiantes.</p> <p>Guarda y envía los resultados.</p>	<p>1.1 La plataforma solicita los datos necesarios para crear una nueva actividad:</p> <p>Nombre de la Actividad.</p> <p>Descripción.</p> <p>Aspectos a asignar.</p> <p>Tipo de Actividad:</p> <p>Auto-evaluación.</p> <p>Co-evaluación.</p>

Ambas.
Grupo.
Verifica que los campos no estén vacíos y la información sea válida.
4.2 La plataforma guarda la actividad asignada.

Prototipo de Interfaz



Flujos Alternos

Acción del Actor

4.1. a Guarda los resultados sin enviar.
 4.1.a.1 Se repite el flujo normal de eventos a partir del #2.

Respuesta del Negocio

4.1.b La plataforma guarda los resultados

Prototipo de Interfaz

Poscondiciones

Flujo Normal de Eventos

Sección “Eliminar Actividad”

Acción del Actor

El Profesor elige eliminar una Actividad de Auto y/o Co-evaluación.
 El profesor confirma eliminar la actividad.

Respuesta del Sistema

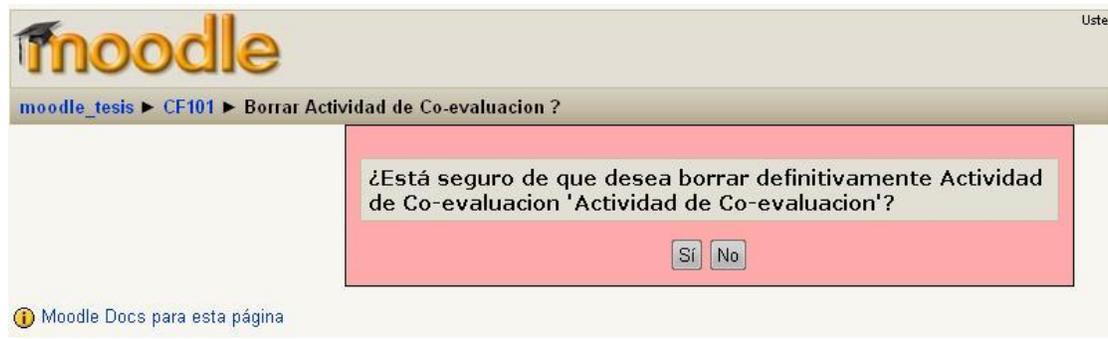
1.1 La plataforma pide la confirmación de la acción de eliminar.
 1.1.1 Sí.

	<p>1.1.2 No. 2.1. La plataforma elimina la actividad.</p>
--	---

Prototipo de



Interfaz



Flujos Alternos

Acción del Actor	Respuesta del Negocio
1.1.2.a. El profesor no confirma Eliminar la Actividad.	1.1.2.b No efectúa la acción de Eliminar Actividad.

Prototipo de Interfaz

Poscondiciones

Flujo Normal de Eventos

Sección "Modificar Actividad"

Acción del Actor	Respuesta del Sistema
El Profesor elige modificar una Actividad de Auto y/o Co-evaluación.	1.1 La plataforma solicita los datos necesarios para modificar una actividad'
2. El profesor modifica los datos solicitados	1.1.2 Nombre de la Actividad.

Guarda y envía los resultados.

Descripción.

Aspectos a asignar.

Tipo de Actividad:

Auto-evaluación.

Co-evaluación.

Ambas.

Grupo.

3.1 Verifica que los campos no estén vacíos y la información sea válida.

3.2 La plataforma guarda los cambios de la Actividad a.

Prototipo de Interfaz



Actualizando Actividad de Co-evaluación en semana 0 ?

Ajustes generales

Nombre de la Actividad* Actividad de Co-evaluación

Descripción* ?

Trebuchet 1 (8 pt) Lang B I U S x₂ x₂ [Rich Text Editor Icons]

descripción.....

Ruta: body

Seleccionar Aspectos

ASPECTO # 2

ASPECTO # 3

2

Tipo de Actividad Auto-evaluación

Asignar Grupo

Grupo* 2502

Ajustes comunes del módulo

Modo de grupo ? No hay grupos

Visible Mostrar

Guardar cambios y mostrar Guardar cambios y regresar al curso Cancelar

Flujos Alternos

Acción del Actor	Respuesta del Negocio
4.1.a Guarda los resultados sin enviar. 4.1.a.1 Se repite el flujo normal de eventos a partir del #2.	4.1.b La plataforma guarda los resultados

Prototipo de Interfaz

Poscondiciones	
-----------------------	--

Anexo # 2. Diagramas de Interacción.

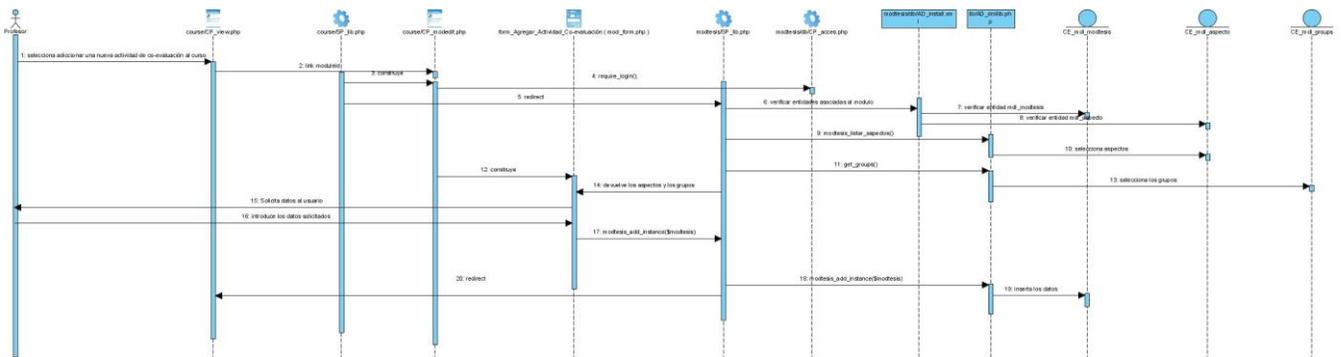


Figura 13 DI CU Gestionar Actividad de Auto y/o Co-evaluación (Insertar).

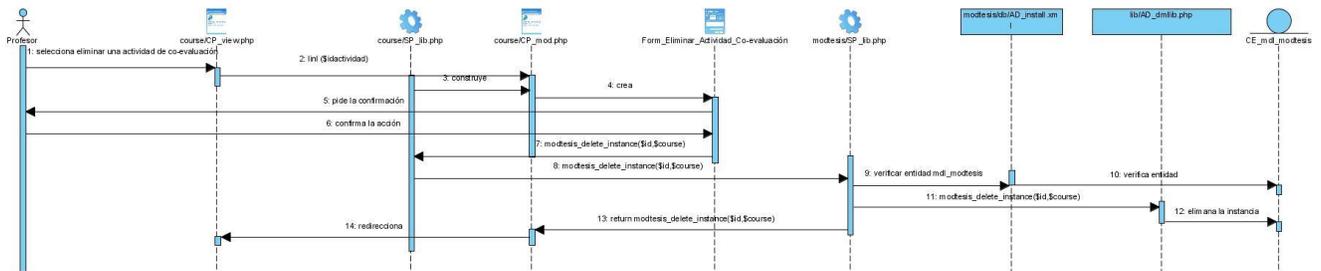


Figura 14 DI CU Gestionar Actividad de Auto y/o Co-evaluación (Eliminar).

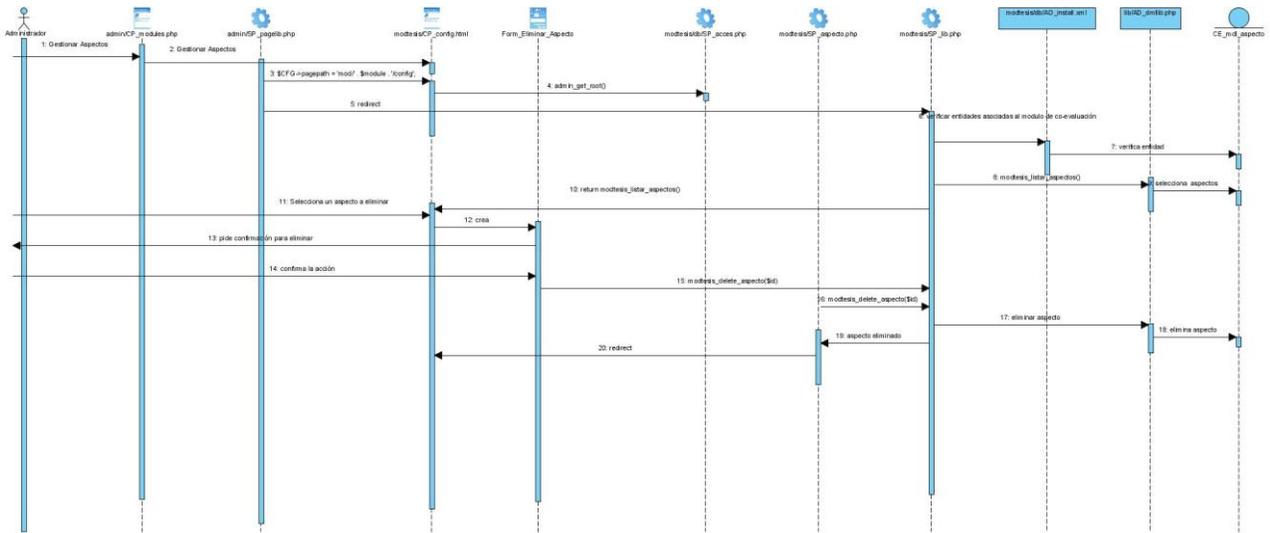


Figura 15 DI CU Gestionar Aspectos (Eliminar).

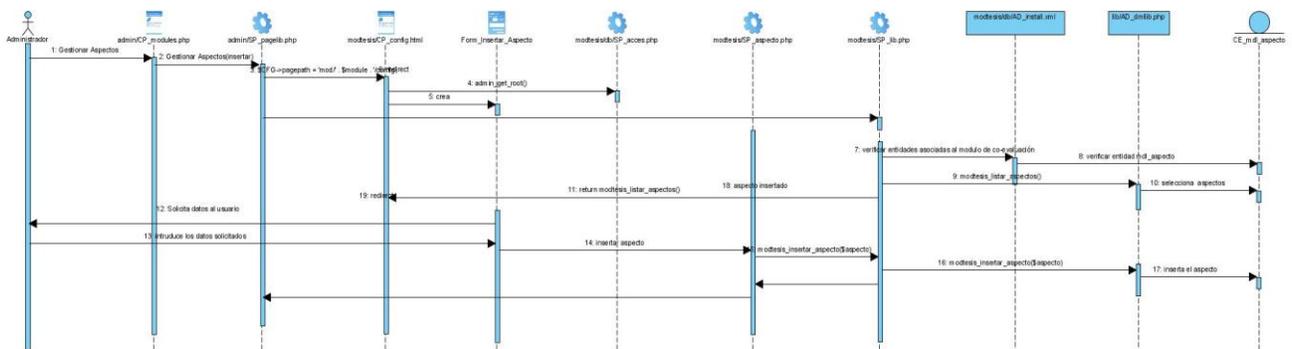


Figura 16 DI CU Gestionar Aspectos (Insertar).

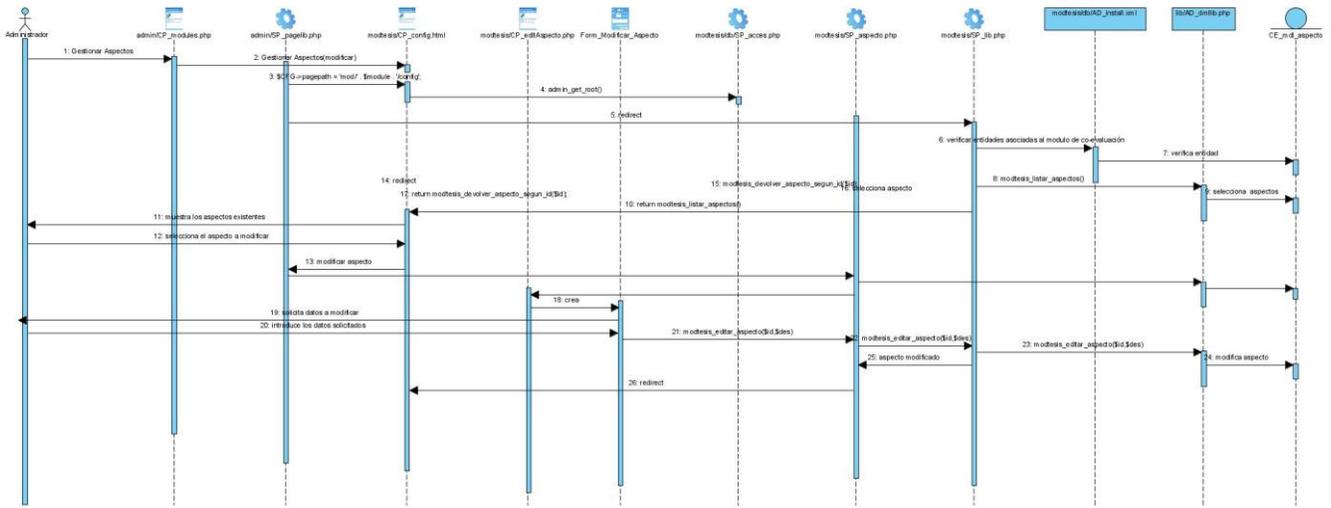


Figura 17 DI CU Gestionar Aspectos (Modificar).

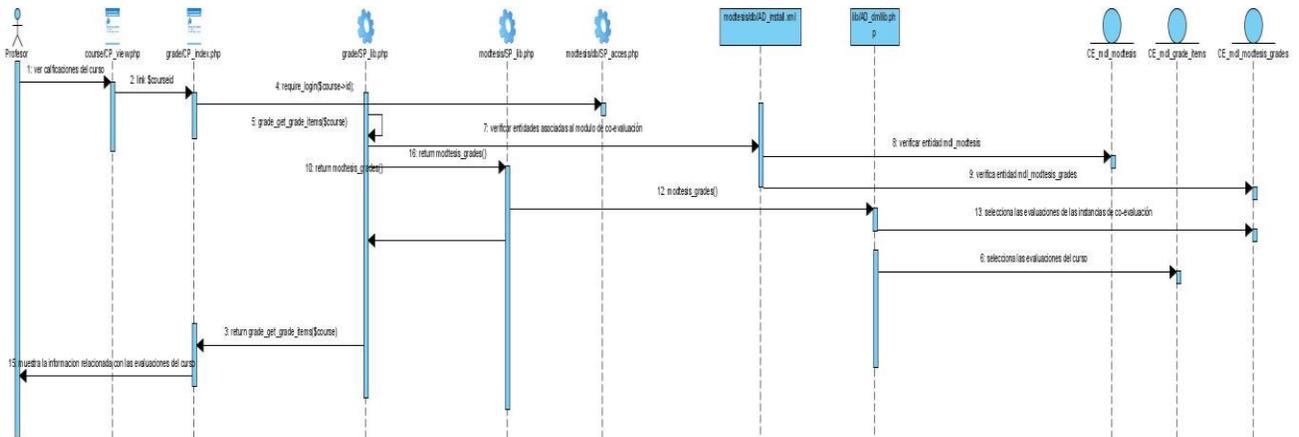


Figura 18 DI CU Mostrar Resultados en el Libro de Calificación.

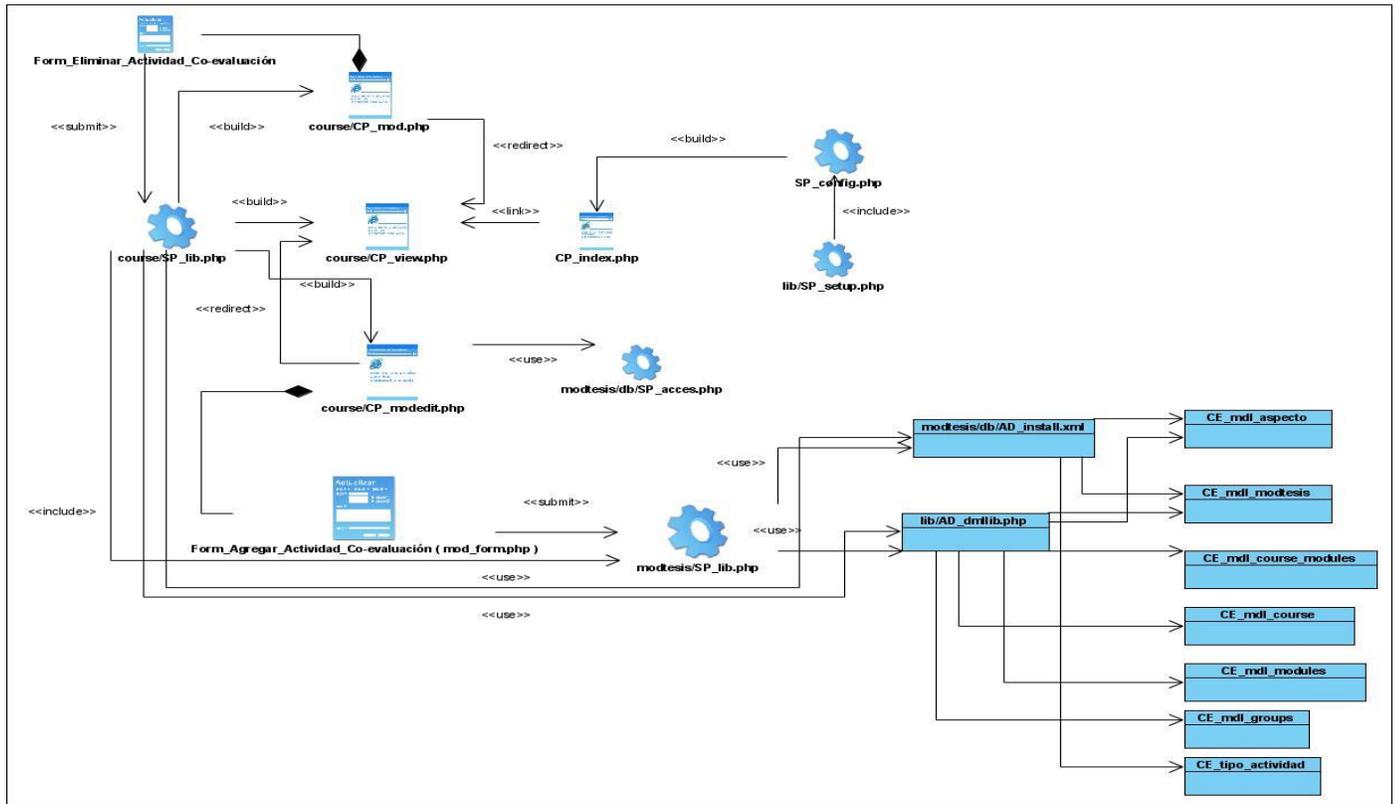


Figura 20 CU Gestionar Actividad de Auto y/o Co-evaluación.

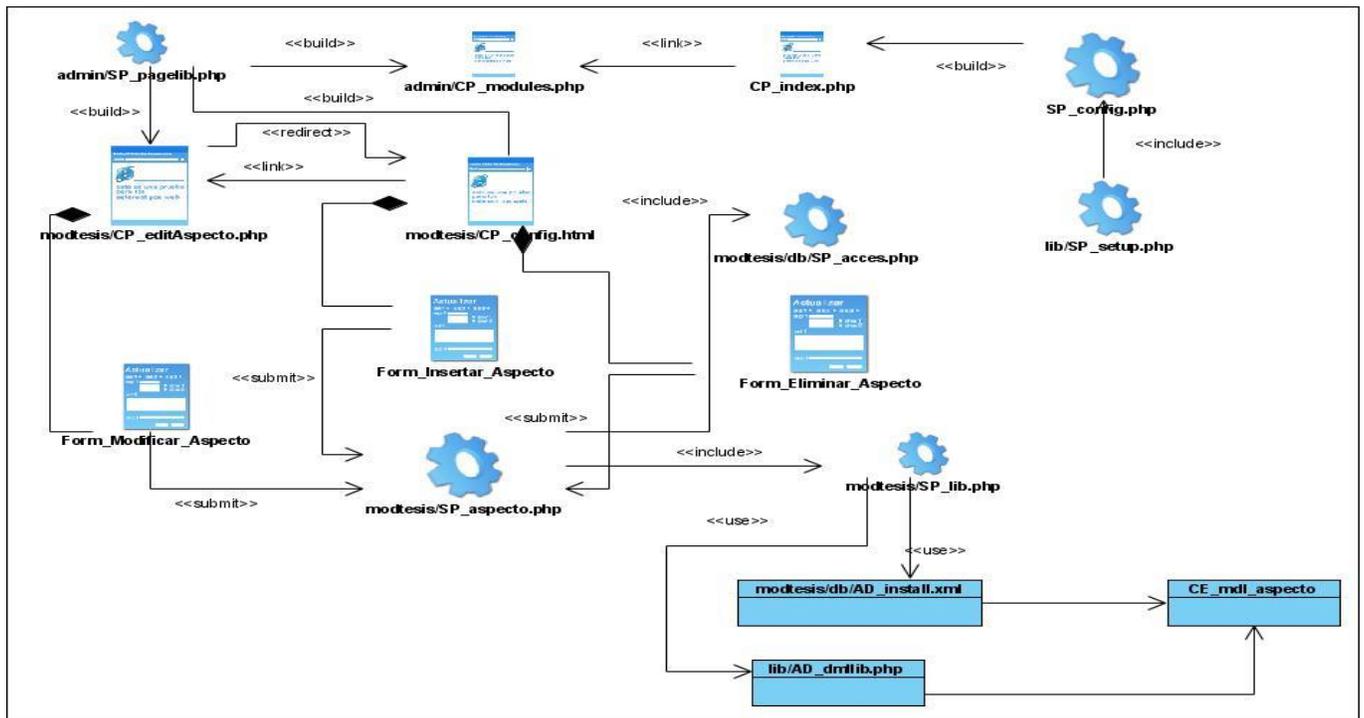


Figura 21 CU Gestionar Aspectos.

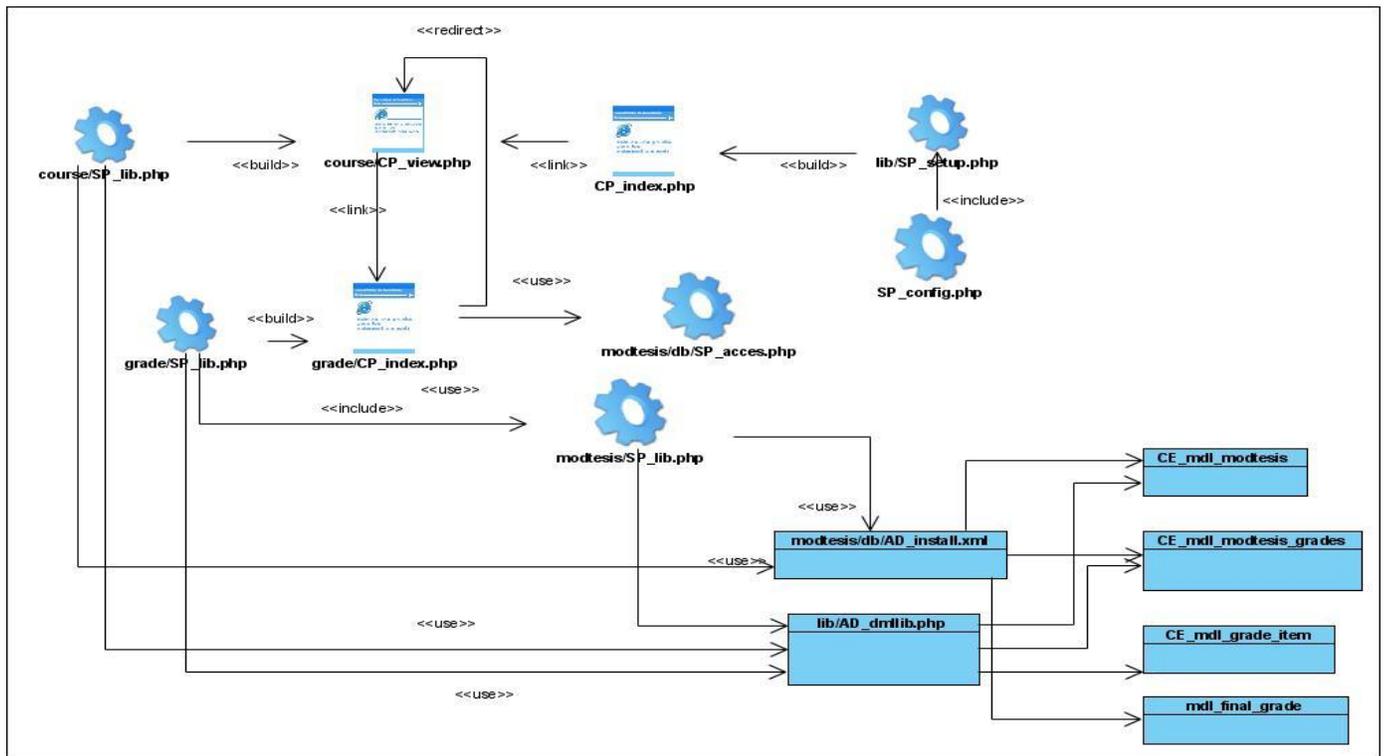


Figura 22 CU Mostrar Resultados en el Libro de Calificación.

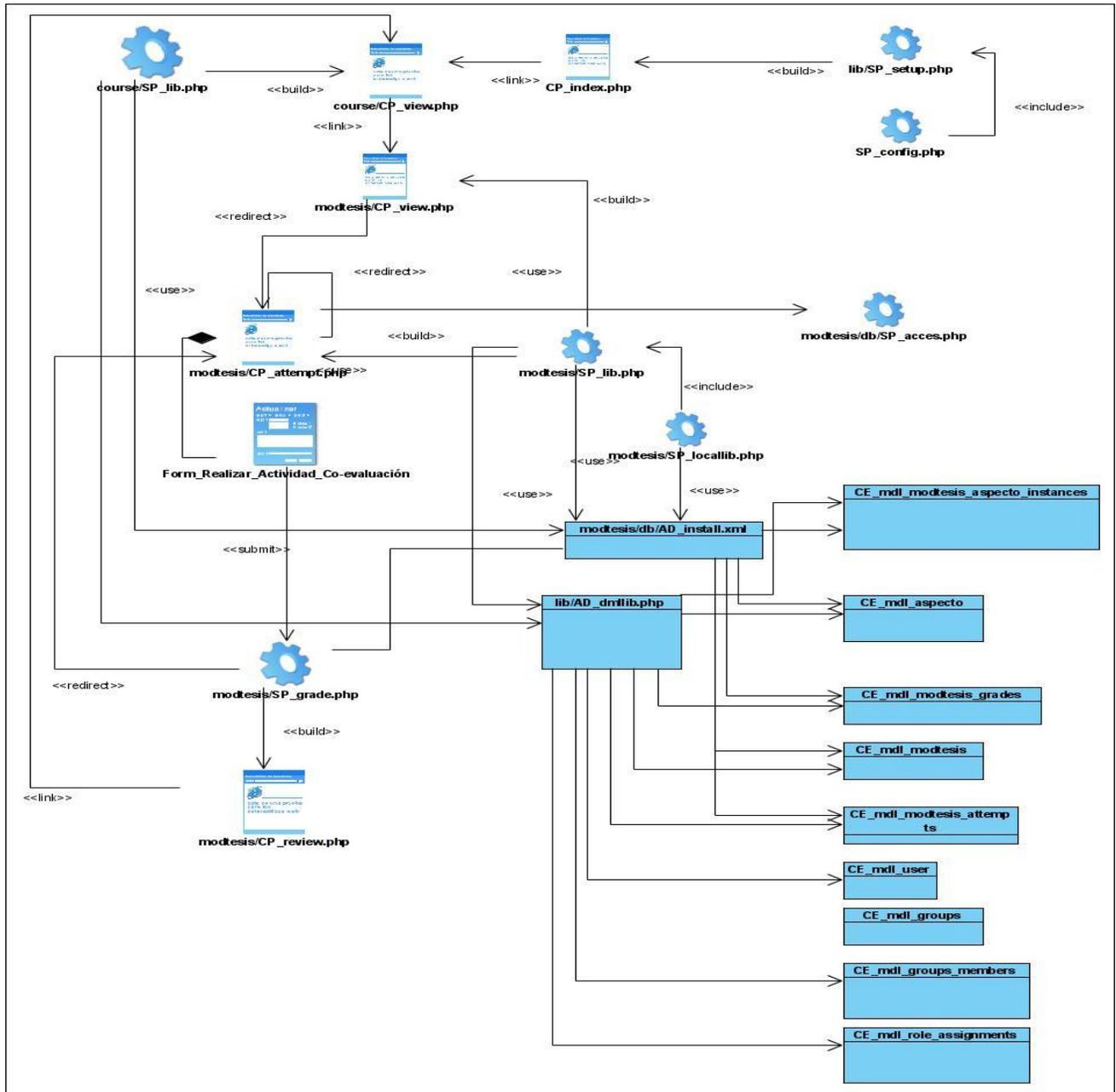


Figura 23 CU Realizar Actividad de Auto y/o Co-evaluación.

Anexo # 4 Diagramas de Componentes.

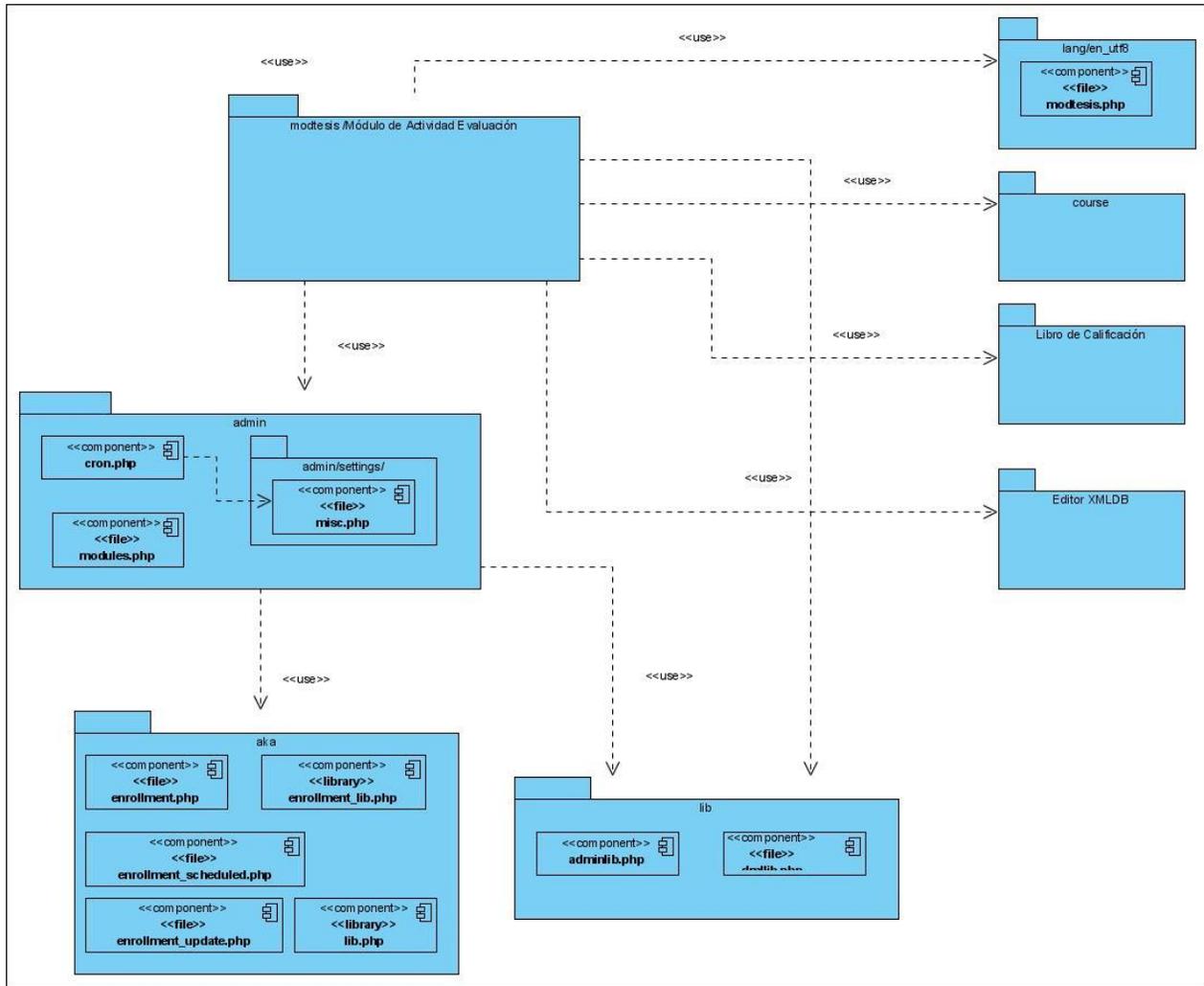


Figura 24 Diagrama de Componentes Módulo de Actividad “Evaluación”(General).

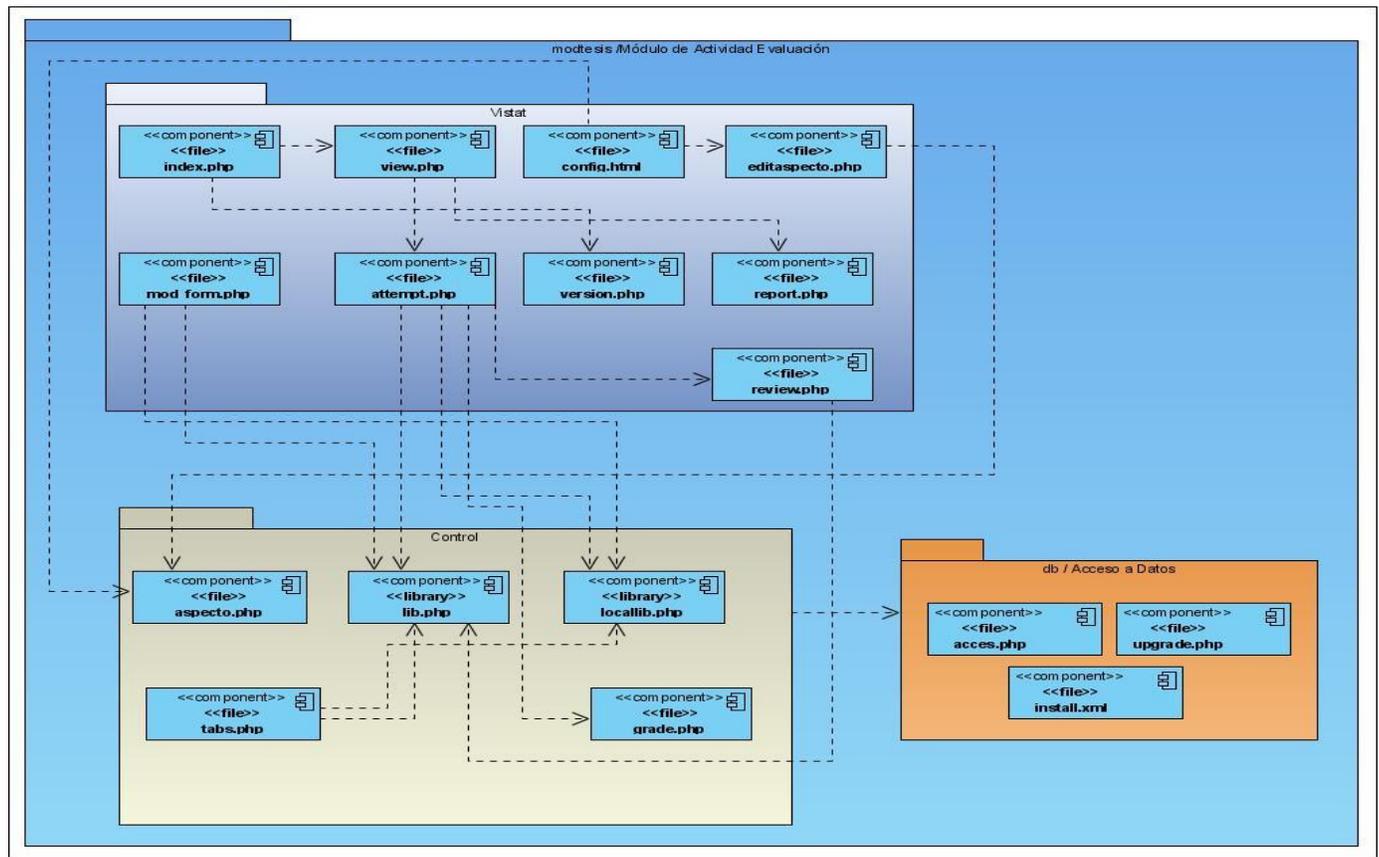


Figura 25 Diagrama de Componentes Módulo de Actividad “Evaluación” (módulo modtesis).

GLOSARIO DE TÉRMINOS

MOODLE: Entorno de Aprendizaje Dinámico Modular Orientado a Objetos.

TIC: Tecnologías de la Informática y las Comunicaciones.

UCI: Universidad de las Ciencias Informáticas.

PEA: Proceso de Enseñanza Aprendizaje.

EVA: Entorno Virtual de Aprendizaje.

LMS: Sistema de Administración de Aprendizaje.

CMS: Sistema de Gestión de Contenidos.

LCMS: Sistema de Administración de Contenidos de Aprendizaje.

OA: Objetos de Aprendizaje.

ROA: Repositorio de Objetos de Aprendizaje

XML: Lenguaje de Marcas Extensible.

SCORM: Sharable Content Object Reference Model.

GNU: Licencia Pública General.

PHP: Procesador del Hipertexto.

URL: Localizador de Recurso Uniforme.

HTML: Lenguaje de Marcas Hipertextuales.

SEPAD: Sistema de Enseñanza Personalizado A Distancia.

CEIS: Centro de Estudios de Ingeniería de Sistemas.

CUJAE: Instituto Superior Politécnico José Antonio Echeverría.

RUP: Rational Unified Process.

UML: Lenguaje Unificado de Modelación.

OMT: Object Modeling Techniques.

BSD: Berkeley Software Distribution..

ACID: Atomicidad, Consistencia, Aislamiento y Durabilidad.

PITR: point in time recovery.

MVCC: Multi-Version Concurrency Control.

SSL: Protocolo de Capa de Conexión Segura.

UUID: Identificador Universalmente Único.

SQL: Lenguaje de Consulta Estructurado.

HTTP: Protocolo de Transferencia de Hipertexto.

WWW: World Wide Web.

CSS: Hojas de Estilo en Cascada.

ASP: Active Server Pages.

SGML: Lenguaje de Mercado Generalizado.