

**UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS**



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

**BLOQUE DE ACTIVIDADES COMPARTIDAS  
ENTRE LOS CURSOS PARA MOODLE 2.9.X**



**AUTORES:** LISANDRA REMEDIOS REVOL.

NEOBEL ROLANDO PAVÓN HERNÁNDEZ.

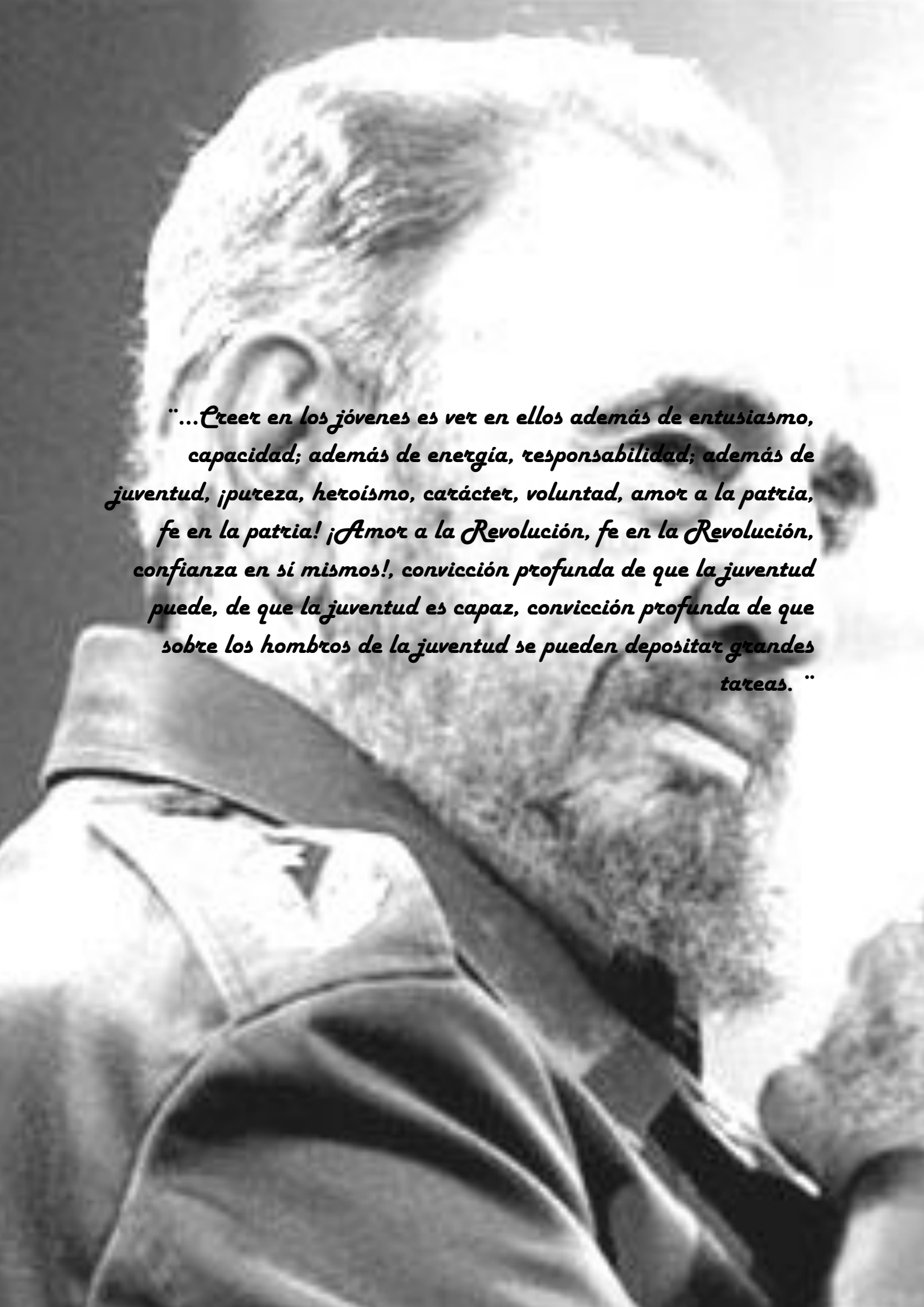
**TUTORES:** MSc, PROF AUXILIAR. LUIS AUGUSTO

ARIAS VERDECIA.

ING. ODENYS ALMORA RODRIGUEZ.

ING. YENIMA HERNÁNDEZ OROZCO.

**LA HABANA 2016**



*“... Creer en los jóvenes es ver en ellos además de entusiasmo, capacidad; además de energía, responsabilidad; además de juventud, ¡pureza, heroísmo, carácter, voluntad, amor a la patria, fe en la patria! ¡Amor a la Revolución, fe en la Revolución, confianza en sí mismos!, convicción profunda de que la juventud puede, de que la juventud es capaz, convicción profunda de que sobre los hombros de la juventud se pueden depositar grandes tareas.”*

## **DECLARACIÓN DE AUTORÍA**

Declaramos que somos los autores del trabajo “Bloque de actividades compartidas entre los cursos para Moodle 2.9.x” y delegamos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmamos la presente a los \_\_\_\_ días del mes de \_\_\_\_\_ del año \_\_\_\_\_.

### **Autores:**

\_\_\_\_\_

Lisandra Remedios Revol.

\_\_\_\_\_

Neobel Rolando Pavón Hernández.

### **Tutores:**

\_\_\_\_\_  
MSc. Luis Augusto Arias Verdecia.

\_\_\_\_\_  
Ing. Odenys Almora Rodriguez.

\_\_\_\_\_  
Ing. Yenima Hernández Orozco.

## DATOS DE CONTACTO

MSc. Luis Augusto Arias Verdecia

Correo electrónico: [ariasverde@uci.cu](mailto:ariasverde@uci.cu)

Trabaja en el Departamento de Ciencias Sociales y Humanidades de la Facultad 4.

Ing. Yenima Hernández Orozco

Correo electrónico: [yhorozco@uci.cu](mailto:yhorozco@uci.cu)

Ingeniera en Ciencias Informáticas, UCI, 2013. Trabaja en el Departamento de Componentes del centro FORTES de la Facultad 4. Tiene 6 años de experiencia en el tema de la teleformación y 3 años de graduada.

Ing. Odenys Almora Rodríguez

Correo electrónico: [odenys@uci.cu](mailto:odenys@uci.cu)

Ingeniero en Ciencias Informáticas, UCI, 2014. Trabaja en el Departamento de Componentes del centro FORTES de la Facultad 4. Tiene 6 años de experiencia en el tema de la teleformación y 2 años de graduado.

## DEDICATORIAS

---

*A mis padres Osvaldo y Sury las personas más importantes de mi vida a los que amo y dedico todos mis triunfos.*

*A mis abuelos por todo lo que significan para mí desde que tengo uso de razón y en especial a mima Justa y a mi abuela Ofelia que, aunque ya no están en este mundo siempre han sido mis guías.*

*A mi familia por apoyarme en todo momento.*

*Lisandra Remedios Revol.*

*A mi mamá Chaira porque todos mis logros han sido en gran parte gracias a ella, por todo su esfuerzo y sacrificio para lograr que hoy yo llegara aquí, por su amor incondicional, por estar siempre a mi lado y apoyarme tanto en la vida estudiantil como personal.*

*A mi papá Nobel por su apoyo y confianza en mí, por su ejemplo como padre y como hombre, por los valores que me inculcó para convertirme en el hombre que soy hoy, por motivarme y exigirme buenos resultados en los estudios.*

*A mis abuelos Aleida y Rolando por todo su amor, cariño y dedicación porque a lo largo de toda mi vida han sido como padres para mí.*

*A mi tía Yuralmis porque más que mi tía ha sido como otra madre para mí, por toda su ayuda y apoyo incondicional.*

*A mi hermano Neobel al que quiero mucho y espero que siga mi ejemplo.*

*Neobel Rolando Pavón Hernández.*

## AGRADECIMIENTOS

---

*A mis padres por su apoyo incondicional, su sacrificio, dedicación, por ayudarme a enfrentar momentos difíciles en mi vida, por confiar en mí y convertirme en la persona que soy hoy. A ellos se lo debo todo, los amo.*

*A mis abuelos por su cariño, amor, contemplación y protección desde pequeña y por dejar esos bellos recuerdos que serán eternos mientras exista.*

*A mi tía Oneyda mi segunda mamá, la que siempre me acogió como una hija y de la que siempre voy a estar agradecida por todo su apoyo, te quiero.*

*A mi tía Pami, Noeida y Sunilda por acompañar a mi mamá y ayudarla todo este tiempo, donde también me ayudaban a mí, pues posibilitaban que yo siguiera adelante con mi carrera.*

*A mi tío Orlando y a tita Yulie por su disposición en todo momento y por todas las vacaciones lindas que pasé junto a ellos, que marcaron los recuerdos más bonitos de mi vida.*

*A mi familia en general porque de una forma u otra todos me apoyaron haciendo posible el sueño de graduarme.*

*A mi novio por soportarme estos dos años donde a pesar de mi carácter fuerte, nunca faltó su comprensión, su amor y su cariño. Gracias por levantarme el ánimo y secar mis lágrimas tantas veces, cuando creía que no iba a lograrlo.*

*A mis compañeros de aula en estos cinco años los que me ayudaron y con lo que pasé inolvidables momentos a: Dinier, Vicente, Víctor, Yuniesky, Gustavo, Ricardo, Jorge, Julio, Roilán, Pedro, Humberto, Luis Daniel, Richar, Yeni, Laura, Dalianne, Cleidy, Yinelis, Yaritza, Rosmery y Susana.*

*A mi compañero de tesis por soportar mi carácter e impaciencia, por todo su apoyo y comprensión y por ayudar a que este sueño se hiciese realidad.*

*A todos los profesores que desde la primaria contribuyeron a que hoy sea una profesional, a ustedes gracias.*

*A mis tutores por la paciencia y dedicación que me han tenido y por el tiempo dedicado.*

*A Fidel y a esta Revolución que me ha dado la posibilidad de estudiar esta carrera para convertirme en una Ingeniera y cumplir el sueño de mi vida.*

*Lisandra Remedios Revol.*

# AGRADECIMIENTOS

---

*A mi mamá por ser la mejor madre del mundo para mí, por todo su apoyo esfuerzo y sacrificio, por todo lo que ha hecho por mí y por todo lo que me ha dado.*

*A mi papá por su confianza en que yo podía lograr llegar hasta aquí, por estar siempre cerca de mí, apoyarme y ayudarme en todo lo que ha podido.*

*A mis abuelos Aleida y Rolando por todo lo que han hecho por mí, por ayudar a mis padres a criarme y formarme como persona, por su amor y cariño y ser como padres para mí.*

*A mi tía Yuralmis por quererme y apoyarme como si fuera mi madre por estar siempre presente cuando la necesito.*

*A mi hermano por su cariño, por hacerme la vida más alegre con sus bromas, sus juegos y su compañía.*

*A mi tía Anita por todo su cariño y apoyo, por toda la ayuda que me ha brindado cuando la he necesitado, a Carlos por toda su ayuda tanto a mi mamá como a mí, a Ireisis por su cariño y preocupación y por cuidar a mi papá.*

*A mi compañera de tesis Lisandra por toda su ayuda y exigencia, por aguantarme todo este tiempo y por convertirse en una buena amiga y a su novio Henry por su ayuda.*

*A Víctor y Alexis porque han sido como hermanos para mí, por estar presentes en las buenas y en las malas, por sus consejos y por todo lo que hemos compartido, a Omar porque fue el primer amigo que hice aquí en la universidad, a Richard, Osvaldo, Ulises, Orestes y a los demás compañeros del grupo actual y a los de antes.*

*A Jose y Oniel por su ayuda en el desarrollo de este trabajo.*

*A los tutores por su guía y orientación en el desarrollo de la tesis.*

*A Yunierkis porque también ha sido como un hermano para mí, a Rosmelys, a Leonard, a Yisel, a Yordanis, a Daniela, Jorge (El flaco), Henry, Rosbel y todo el piquete del doble.*

*A todos los profesores que tuve por sus enseñanzas y a todas las personas que de una manera u otra me ayudaron y se preocuparon por mí durante toda la carrera, a todos Muchas Gracias.*

*Neobel Rolando Pavón Hernández.*

## RESUMEN

---

El desarrollo de las Tecnologías de la Información y las Comunicaciones en el área educativa dio paso al surgimiento de herramientas que facilitan la gestión del proceso de enseñanza-aprendizaje. La plataforma Moodle es uno de los Sistemas de Gestión del Aprendizaje más reconocidos. Ésta se utiliza en la Universidad de las Ciencias Informáticas, como apoyo al proceso de enseñanza-aprendizaje. Las personalizaciones que se solicitan por los clientes del Centro FORTES de la Facultad 4, precisan de la versión 2.9.x de la plataforma. De los componentes solicitados se encuentra el módulo Share v1.0, el mismo está desarrollado para la versión 2.3.x de Moodle con el propósito de compartir actividades entre los cursos de la plataforma. Se realizó un análisis al módulo y se detectaron deficiencias que imposibilitan su uso en Moodle 2.9.x. Se identificó que su forma de compartir las actividades era básica y no incluía aspectos que enriquecieran la interactividad. Para solventar el problema se desarrolla un bloque que permite compartir actividades entre los cursos de la plataforma Moodle en la versión 2.9.x. En él se incluyen elementos de gamificación para enriquecer la interactividad y motivar a la compartición de conocimiento entre los profesores de la plataforma. Su desarrollo está guiado por la metodología AUP versión UCI, en su elaboración se utilizan herramientas y tecnologías como: Visual Paradigm, Netbeans, CSS3, HTML, JavaScript, PHP y XML. Luego de haber desarrollado la implementación del bloque se realizaron pruebas funcionales que demostraron el adecuado funcionamiento del mismo.

**Palabras Clave:** bloque, compartir actividad, interactividad, Moodle.



# ÍNDICE

<b>Dedicatorias</b>	<b>I</b>
<b>Agradecimientos</b>	<b>II</b>
<b>Agradecimientos</b>	<b>III</b>
<b>Resumen</b>	<b>IV</b>
<b>Introducción</b>	<b>1</b>
<b>Capítulo 1: Fundamentación Teórica</b>	<b>7</b>
<b>1.1 E-learning</b>	<b>7</b>
<b>1.2 Los Sistemas de Gestión de Aprendizaje</b>	<b>8</b>
<b>1.3 Plataforma Moodle</b>	<b>9</b>
1.3.1 Bloques de la plataforma Moodle	13
1.3.2 Compartición de actividades en Moodle	14
Funcionalidad	15
Importar	15
Módulo	15
Share v1.0	15
<b>1.4 Protocolo OAI-PMH</b>	<b>16</b>
<b>1.5 Licencias Creative Commons(CC)</b>	<b>20</b>
<b>1.6 Gamificación</b>	<b>22</b>
1.6.1 Elementos de gamificación de Moodle	22
<b>1.7 Tecnologías y herramientas a utilizar en la solución</b>	<b>24</b>
1.7.1 Tecnologías del lado del servidor	24
1.7.1.1 Preprocessed Hypertext Pages	24
1.7.1.2 Metalenguaje de marcas extensible	25
1.7.1.3 Servidor web Apache	26
1.7.1.4 Sistema gestor de base de datos	26
1.7.2 Tecnologías del lado del cliente	28
1.7.2.1 Lenguaje de Marcado de Hipertexto	29
1.7.2.2 Hojas de estilo en cascada	29
1.7.2.3 JavaScript	29
1.7.2.4 JQuery	30
1.7.2.5 YUI	30
<b>1.8 Herramientas de desarrollo</b>	<b>31</b>

1.8.1	Entorno de Desarrollo Integrado (IDE) NetBeans	31
1.8.2	Lenguaje de Modelado	32
1.8.3	Herramientas CASE	32
1.8.3.1	Visual Paradigm	32
<b>1.9</b>	<b>Metodología de desarrollo de software</b>	<b>33</b>
1.9.1	Fases AUP	34
1.9.2	Disciplinas AUP	34
	<b>Conclusiones parciales</b>	<b>35</b>
	<b>Capítulo 2: Análisis y Diseño del bloque</b>	<b>36</b>
<b>2.1</b>	<b>Propuesta de solución</b>	<b>36</b>
<b>2.2</b>	<b>Modelo conceptual</b>	<b>37</b>
<b>2.3</b>	<b>Especificación de requisitos</b>	<b>39</b>
2.3.1	Requisitos funcionales	39
2.3.2	Requisitos no funcionales	41
<b>2.4</b>	<b>Historias de usuario</b>	<b>43</b>
<b>2.5</b>	<b>Modelo de análisis</b>	<b>48</b>
2.5.1	Diagrama de clases del análisis (DCA)	48
2.5.2	Diagrama de colaboración del análisis (DCO)	50
<b>2.6</b>	<b>Patrón Arquitectónico</b>	<b>52</b>
<b>2.7</b>	<b>Modelo de diseño</b>	<b>53</b>
2.7.1	Patrones de diseño	53
2.7.2	Diagrama de clases de diseño.	55
2.7.3	Diagrama de secuencia del diseño (DS)	56
<b>2.8</b>	<b>Diseño de la base de datos</b>	<b>58</b>
	<b>Conclusiones parciales</b>	<b>60</b>
	<b>Capítulo 3: Implementación y Validación del bloque</b>	<b>61</b>
<b>3.1</b>	<b>Modelo de implementación</b>	<b>61</b>
<b>3.2</b>	<b>Diagrama de despliegue</b>	<b>61</b>
<b>3.3</b>	<b>Diagrama de componentes</b>	<b>62</b>
<b>3.4</b>	<b>Pruebas de software</b>	<b>63</b>
3.4.1	Niveles de prueba	63
3.4.2	Métodos de prueba	63
3.4.3	Pruebas de caja negra	64

3.4.4	Diseños de casos de prueba	64
3.4.5	Resultados de las pruebas	70
	<b>Conclusiones parciales</b>	<b>73</b>
	<b>Conclusiones Generales</b>	<b>74</b>
	<b>Recomendaciones</b>	<b>75</b>
	<b>Referencias Bibliográficas</b>	<b>76</b>

## ÍNDICE DE FIGURAS

<b>Figura 1</b>	Modelo Conceptual	38
<b>Figura 2</b>	Estereotipo de la clase interfaz	49
<b>Figura 3</b>	Estereotipo de la clase control	49
<b>Figura 4</b>	Estereotipo de la clase entidad	49
<b>Figura 5</b>	Diagrama de clases del análisis de la HU Compartir actividad	49
<b>Figura 6</b>	Diagrama de clases del análisis de la HU Buscar AC (avanzada)	50
<b>Figura 7</b>	Diagrama de clases del análisis Utilizar AC en el curso	50
<b>Figura 8</b>	Diagrama de Colaboración de la HU Compartir actividad	51
<b>Figura 9</b>	Diagrama de Colaboración de la HU Buscar AC (avanzada)	51
<b>Figura 10</b>	Diagrama de Colaboración de la HU Utilizar AC en el curso	52
<b>Figura 11</b>	Diagrama de clase que representa al patrón experto	54
<b>Figura 12</b>	Diagrama de clase que representa al patrón controlador	54
<b>Figura 13</b>	Diagrama de clase que representa el patrón bajo acoplamiento	55
<b>Figura 14</b>	Diagrama de clases de diseño	56
<b>Figura 15</b>	Diagrama de Secuencia de la HU Compartir actividad	57
<b>Figura 16</b>	Diagrama de Secuencia de la HU Buscar AC (avanzada)	57
<b>Figura 17</b>	Diagrama de Secuencia de la HU Utilizar AC en el curso	58
<b>Figura 18</b>	Diagrama Entidad-Relación	59
<b>Figura 19</b>	Diagrama de despliegue	61
<b>Figura 20</b>	Diagrama de componentes	62
<b>Figura 21</b>	Resultados de las pruebas de caja negra	73

## ÍNDICE DE TABLAS

<b>Tabla 1</b> Comparación de soluciones similares. ....	15
<b>Tabla 2</b> Modelo propuesto para una historia de usuario.....	43
<b>Tabla 3</b> Descripción de la HU Compartir actividad. ....	44
<b>Tabla 4</b> Descripción de la HU Buscar AC (básica). ....	45
<b>Tabla 5</b> Descripción de la HU Buscar AC (avanzada) . ....	46
<b>Tabla 6</b> Descripción de la HU Utilizar AC en el curso.....	47
<b>Tabla 7</b> Descripción de la HU Agregar comentario en el foro desde el repositorio. ....	48
<b>Tabla 8</b> Caso de Prueba Compartir actividad.....	65
<b>Tabla 9</b> Caso de Prueba Buscar AC (avanzada).....	67
<b>Tabla 10</b> Caso de Prueba Utilizar AC en curso. ....	69
<b>Tabla 11</b> Resultados de las pruebas de caja negra por iteraciones.....	71

# INTRODUCCIÓN

---

En la actualidad las Tecnologías de la Información y las Comunicaciones (TIC) constituyen una vía fundamental en la implementación y evolución de actividades formativas. Su adopción en el área educativa ha permitido el desarrollo del proceso de enseñanza-aprendizaje, tanto en instituciones educacionales como en empresas.

Existen varias modalidades educativas que han surgido en correspondencia con las TIC entre las que se encuentra el modelo conocido como e-formación o *e-learning*. Éste se define como un: “conjunto de tecnologías, aplicaciones y servicios orientados a facilitar la enseñanza y el aprendizaje a través de Internet/Intranet, que facilitan el acceso a la información y la comunicación con otros participantes.” (1)

Para dar soporte a esta modalidad educativa surgen los Sistemas de Gestión de Aprendizaje (LMS por sus siglas en inglés); herramientas que permiten automatizar contenidos de formación y controlar los procesos de enseñanza-aprendizaje. Entre los LMS más utilizados se encuentra la plataforma Moodle (del inglés *Modular Object-Oriented Dynamic Learning Environment*). (2)

Moodle es una aplicación web gratuita, diseñada para ayudar a los educadores a crear cursos en línea y establecer un control detallado de las acciones del estudiante a través de los entornos de aprendizaje virtuales. Cuenta con una comunidad de usuarios que tienen como objetivo el desarrollo colaborativo de módulos y personalizaciones de esta plataforma. Posee una estructura modular, permitiendo gran flexibilidad para añadir y eliminar funcionalidades. Contiene módulos (entre ellos los recursos y las actividades), filtros y bloques.

Los bloques en la plataforma Moodle permiten agregar funciones extra para fortalecer las capacidades del sistema, se encuentran situados a los lados del sitio web y sus diversas funcionalidades pueden estar ligadas a otros bloques o trabajar de forma independiente. Contribuyen a perfeccionar el aprendizaje y las actividades formativas.

En Cuba, la aplicación de las TIC en la educación ha propiciado transformaciones manifestado en la creación de nuevos modelos en la formación y el uso de herramientas informáticas y telemáticas.

Una muestra del avance de la informatización de la Educación Superior en Cuba es la creación de la Universidad de las Ciencias Informáticas (UCI), 2002. Ésta cuenta con disponibilidad

para satisfacer las necesidades de la formación de los estudiantes, mediante el uso de herramientas disponibles en la plataforma Moodle.

En la UCI se inicia el trabajo con la plataforma Moodle, como parte de su programa de teleformación, a partir del año 2005. Durante el curso 2009-2010 en la Facultad 4, se crea el Centro de Tecnologías para la Formación (FORTES). Éste cuenta con una Línea de Producción de Software denominada Desarrollo de Componentes para Moodle, la cual está integrada por un grupo de especialistas que se enfocan en el desarrollo de componentes que dan solución a las necesidades identificadas por los usuarios de la plataforma.

Actualmente la plataforma se encuentra en la versión 2.9.x. Las personalizaciones que se solicitan por los clientes del Centro FORTES de la Facultad 4, precisan de la versión estable de la misma. De los componentes solicitados se encuentra el módulo Share v1.0, el mismo está desarrollado para la versión 2.3.x con el propósito de compartir actividades entre los cursos de la plataforma Moodle. Con el objetivo de verificar el funcionamiento del módulo y la forma en que comparte las actividades, se le realizó un análisis detectándose las siguientes deficiencias:

- Los cambios de la API (*Application Programming Interface*) entre las versiones 2.3.x y 2.9.x afectan el uso del módulo Share v1.0 siendo incompatible con la versión 2.9.x de Moodle.
- Modifica el core de Moodle para realizar alguna de sus funcionalidades.
- No incluye información estadística sobre el uso de las actividades compartidas de los cursos, por lo que no permite apreciar el nivel de aceptación que tiene la actividad compartida.
- No brinda la funcionalidad que permita respetar el derecho del autor de la actividad compartida.
- No cuenta con un repositorio que permita la disponibilidad de las actividades compartidas debido a que si esta es eliminada no podrá ser reutilizada.
- No permite mostrar el contenido de las actividades en forma de metadatos lo que imposibilita compartir esta información en otros repositorios o bibliotecas virtuales relacionados con el tema.

- No incorpora técnicas de gamificación que contribuyan a aumentar la calidad y uso de las actividades que se comparten.
- No cuenta con información detallada de la actividad compartida que facilite la elección.
- No cuenta con un foro que permita a los profesores valorar las actividades compartidas de los cursos y establecer un ambiente de colaboración.
- No se encuentra registrado la evidencia del desarrollo de las recomendaciones identificadas por los autores del módulo Share v1.0 para la versión 2.3.x de Moodle.
- No cumple con la filosofía de la plataforma, debido a que este constituye un módulo de tipo actividad que es la parte activa y colaborativa donde el estudiante interactúa con el profesor y su comportamiento es el de un bloque; herramienta que facilita el proceso de enseñanza-aprendizaje.

El estudio realizado evidencia insuficiencias al compartir las actividades con el módulo Share v1.0, debido a que no se incluyen aspectos que enriquezcan la interactividad. Se detectó, además, deficiencias que imposibilitan el uso del módulo Share v1.0 en la versión 2.9.x Moodle. Esto provoca que actualmente los profesores no puedan reutilizar las actividades compartidas por los diferentes profesores de la plataforma Moodle en la versión 2.9.x, ni compartir las creadas por él.

Tomando como referente lo expuesto, se plantea el siguiente **problema de investigación**:

¿Cómo contribuir a compartir actividades entre los cursos de la plataforma Moodle en la versión 2.9.x?

Se declara como **objeto de estudio** proceso de compartir actividades en plataformas educativas.

Enmarcándose en el **campo de acción** proceso de compartir actividades en la plataforma Moodle.

Se plantea la siguiente **idea a defender**: El desarrollo de un bloque que permita compartir actividades entre los cursos de plataforma Moodle en la versión 2.9.x, facilitará a los profesores compartir el conocimiento implícito en las actividades creadas.

Se declara como **objetivo general**: Desarrollar un bloque que permita compartir actividades entre los cursos de la plataforma Moodle en la versión 2.9.x.

### **Objetivos específicos:**

- Construir el marco teórico referencial sobre las formas de compartir actividades entre los cursos de la plataforma Moodle.
- Desarrollar el análisis y diseño de un bloque para la versión 2.9.x de la plataforma Moodle que permita compartir actividades entre los cursos.
- Implementar un bloque para la versión 2.9.x de la plataforma Moodle que permita compartir actividades entre los cursos.
- Validar un bloque para la versión 2.9.x de la plataforma Moodle que permita compartir actividades entre los cursos.

### **Tareas a cumplir:**

- Análisis de los aspectos teóricos relacionados con el modo de compartir actividades entre los cursos de la plataforma Moodle.
- Investigación sobre los estándares para la descripción de contenidos y metadatos.
- Identificación de las tecnologías y herramientas a utilizar en el proceso de desarrollo de un bloque que permita compartir actividades entre los cursos, en la versión 2.9.x de la plataforma Moodle.
- Elaboración de los artefactos correspondientes a la fase de análisis y diseño.
- Elaboración de los artefactos correspondientes a la fase de implementación.
- Implementación de un bloque que permita compartir actividades entre los cursos en la versión 2.9.x de la plataforma Moodle.
- Selección y ejecución de los métodos de prueba para un bloque que permita compartir actividades entre los cursos en la versión 2.9.x de la plataforma Moodle.

Para la realización de la presente investigación se hizo necesaria la utilización de los siguientes métodos de investigación científicos:

### **Métodos teóricos:**



**Analítico-Sintético:** Se utilizó con el objetivo de analizar los elementos bibliográficos y definiciones relacionadas con el término de *e-learning*, LMS y todo lo referente a los elementos de gamificación que se emplean en la plataforma Moodle. También para identificar las tecnologías a utilizar en el desarrollo de la solución.

**Análisis Histórico-Lógico:** Este método permitió analizar la trayectoria de soluciones similares, entre ellas el módulo Share v1.0 para la versión 2.3.x de Moodle e identificar las deficiencias a solventar en la propuesta a desarrollar para la versión actual de la plataforma.

**Modelación:** Se utilizó para reflejar la estructura, relaciones internas y características de la solución a través de la realización de diagramas.

**Método empírico:**

**Encuesta:** Este método permitió analizar las opiniones realizadas por los profesores (clientes) que utilizan la plataforma Moodle y corroborar la necesidad que existe de crear un bloque que permita compartir las actividades entre los cursos de la plataforma Moodle en la versión 2.9.x.

## **Estructura capitular**

### **Capítulo 1: Fundamentación Teórica**

Consiste en un análisis de los principales conceptos relacionados con la investigación. Se efectúa un estudio del estado del arte donde se tiene en cuenta el funcionamiento del módulo Share v1.0 para la versión 2.3.x de Moodle, así como de los cambios que se efectuaron en la plataforma Moodle, desde la versión 2.3.x a la 2.9.x. Son descritas, además, las tecnologías, herramientas, estándares y metodología para llevar a cabo el desarrollo de la solución propuesta.

### **Capítulo 2: Análisis y Diseño del bloque**

Se describe el proceso de desarrollo de la propuesta de solución especificando sus características principales. Se expone el modelo conceptual y los requisitos funcionales y no funcionales de la aplicación a implementar. Se muestran los principales procesos haciendo uso de las historias de usuario. Y se exponen los artefactos que se generan a partir de la metodología de desarrollo de software utilizada.

### **Capítulo 3: Implementación y Validación del bloque**

Se describen los elementos necesarios para la implementación, partiendo del resultado obtenido del diseño. Se muestra la distribución del sistema en nodos mediante el diagrama

de despliegue y la organización de los componentes con sus relaciones lógicas a través del diagrama de componentes, quedando conformado el modelo de implementación del bloque. Se realizan las pruebas de caja negra utilizando la técnica de la Partición de Equivalencia y se reflejan los resultados obtenidos.

# CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

---

La plataforma Moodle, una de las más difundidas a nivel mundial, presenta gran diversidad en lo que respecta a actividades y módulos que favorecen el aprendizaje en línea. Sus características han propiciado que empresas y centros educacionales la utilicen en la capacitación de su personal, mejorando cada una de las funcionalidades que ofrece al adaptarlas a su contexto de trabajo.

En el presente capítulo se realiza una descripción detallada de los aspectos más relevantes relacionados con la plataforma Moodle. Se realiza un estudio sobre el estado del arte relacionado con la compartición de actividades en la plataforma Moodle, así como un análisis de las herramientas, metodología y tecnologías utilizadas para el desarrollo de la investigación.

## 1.1 E-learning

La modalidad educativa *e-learning* o aprendizaje electrónico, ha alcanzado gran auge con el desarrollo de Internet. Su concepción ha transformado la educación abriendo puertas al aprendizaje individual y organizacional. Es por ello que ocupa un lugar reconocido dentro de las organizaciones empresariales y educativas.

De las definiciones más aceptadas de *e-learning* se encuentra la de “enseñanza a distancia caracterizada por una separación física entre profesorado y alumnado sin excluir encuentros físicos puntuales, entre los que predomina una comunicación de doble vía asíncrona donde se usa preferentemente Internet como medio de comunicación y de distribución del conocimiento, de tal manera que el estudiante es el centro de una formación independiente y flexible, al tener que gestionar su propio aprendizaje, generalmente con ayuda de tutores externos”. (3)

*E-learning* ha contribuido a que la formación llegue a un mayor número de personas. Entre sus características más destacadas se encuentran (4):

- Desaparecen las barreras espacio-temporales. Los estudiantes pueden realizar un curso en su casa o lugar de trabajo, estando accesibles los contenidos a cualquier hora del día. Pudiendo de esta forma optimizar al máximo el tiempo dedicado a la formación.

- La diversidad de métodos y recursos empleados, facilita la adaptación a las características y necesidades de los estudiantes.
- El alumno es el centro de los procesos de enseñanza-aprendizaje y participa de manera activa en la construcción de sus conocimientos, teniendo capacidad para decidir el itinerario formativo más acorde con sus intereses.
- El profesor, pasa de ser un mero transmisor de contenidos a un tutor que orienta, guía, ayuda y facilita los procesos formativos.
- Las novedades y recursos relacionados con el tema de estudio se pueden introducir de manera rápida en los contenidos, de forma que las enseñanzas estén totalmente actualizadas.
- Comunicación constante entre los participantes, gracias a las herramientas que incorporan las plataformas *e-learning* (foros, chat, correo-e).

Este tipo de enseñanza permite la interacción del usuario con el material mediante la utilización de diversas herramientas informáticas. Además, proporciona importantes experiencias de aprendizaje, debido a que permite la reutilización de contenidos provenientes de diversas fuentes. Una solución *e-learning* se puede conformar, por un entorno de software diseñado para automatizar y gestionar el desarrollo de actividades formativas, también llamado Sistema de Gestión de Aprendizaje o LMS.

## 1.2 Los Sistemas de Gestión de Aprendizaje

Las plataformas educativas o LMS constituyen un sistema de gestión de aprendizaje en línea, que permite administrar, distribuir, monitorear, evaluar y apoyar las diferentes actividades previamente diseñadas y programadas dentro de un proceso de formación completamente virtual o de formación semi-presencial. (5)

Marcadas características de las plataformas educativas favorecen significativamente la adquisición de aprendizaje en los estudiantes y un ejemplo de éstas son (2):

**Flexibilidad:** La plataforma puede ser adaptada tanto a los planes de estudio de la institución, como a los contenidos y estilo pedagógico de la organización. Permite organizar cursos con gran facilidad y rapidez.

**Interactividad:** La persona se convierte en el protagonista de su propio aprendizaje a través del autoservicio y los servicios autoguiados.

**Estandarización:** Esta característica permite utilizar cursos realizados por terceros, personalizando el contenido y reutilizando el conocimiento.

**Escalabilidad:** Estos recursos pueden funcionar con una cantidad variable de usuarios según las necesidades de la organización.

**Funcionalidad:** Prestaciones y características que hacen que cada plataforma sea adecuada (funcional) según los requerimientos y necesidades de los usuarios.

**Usabilidad:** Facilidad con que las personas pueden utilizar la plataforma con el fin de alcanzar un objetivo concreto.

**Ubicuidad:** Capacidad de una plataforma para generar tranquilidad al usuario y provocarle la certeza de que todo lo que necesita lo va a encontrar en dicho entorno virtual.

**Integración:** Las plataformas LMS deben poder integrarse con otras aplicaciones empresariales utilizadas por recursos humanos y contabilidad, lo que permite medir el impacto, eficacia y, sobre todo, el coste de las actividades de formación.

Las plataformas educativas permiten estimular la idea de cooperación y de interacción, como aspectos centrales del proceso de enseñanza– aprendizaje, mediante el uso de herramientas colaborativas. Un ejemplo de estas herramientas, es la plataforma de teleformación Moodle, la cual se caracteriza por un sistema integrado único, robusto y seguro para crear ambientes de aprendizaje personalizados.

### 1.3 Plataforma Moodle

Moodle es una plataforma virtual de aprendizaje dentro de los sistemas de gestión de procesos de enseñanza – aprendizaje a través de la creación de cursos en línea, pues permite el levantamiento de un centro capaz de gestionar distintos cursos a la vez a través de la red. Su misión es desarrollar nuevas teorías educativas utilizando los recursos didácticos disponibles en ella. (6)

Esta plataforma comenzó a desarrollarse en 1999, de la mano del australiano Martin Dougiamas de Perth quien basó su diseño en las ideas del constructivismo en pedagogía, que afirman que el conocimiento se construye en la mente del estudiante en lugar de ser transmitido sin cambios a partir de libros o enseñanzas y en el aprendizaje colaborativo. (7)

Una de las fortalezas de Moodle es que es Software Libre, su creador inicial, al momento de publicarlo en internet, decidió utilizar la Licencia Pública GNU (GPL) por lo que puede ser utilizado sin pagar licencias. La institución que lo instale está autorizada a copiar, usar y modificar Moodle. (8)

Constituye una herramienta útil en el desarrollo de la educación debido a que soporta disímiles contenidos como son: textos, animaciones, vídeos, imágenes, entre otros. Garantiza realizar exámenes *online* y evaluar las tareas de los estudiantes. Dispone de recursos didácticos constantemente actualizados, en distintos formatos, que contribuyen a fomentar el auto-aprendizaje y el aprendizaje colaborativo.

Es fácil de instalar en casi cualquier plataforma con un servidor Web que soporte PHP sólo requiere que exista una base de datos (esencialmente MySQL o PostgreSQL).

La comunidad de desarrollo de Moodle ha pasado por varias actualizaciones con respecto a las versiones de la plataforma. El trayecto de la versión 2.3 a la 2.9 ha reflejado variadas novedades. A continuación, se muestran algunas de las principales novedades que contribuyen al desarrollo de la presente investigación (9):

- Los ficheros *backuplib.php*, *lib.php*, *restorelib.php* y *the bb directory*, fueron eliminados por lo que cambia la manera de realizar los *backups*.
- Se agregaron nuevas funcionalidades en el fichero *accesslib* para el tratamiento de los permisos.
- Se agregaron nuevas funcionalidades para *output*, por lo que cambia la manera de crear las vistas de las páginas.
- Se realizaron cambios en el uso de funciones de javascript y del framework YUI para el tratamiento de notificaciones y otros elementos visuales.
- Se empezó a utilizar el render mediante plantillas php o javascript.
- Interfaces *sql\_internal\_reader*, *sql\_select\_reader* y *lector sql\_select* han sido eliminados y se han creado *sql\_internal\_table\_reader* y *sql\_reader* que utilizan iteradores para hacer más eficiente la memoria.

La plataforma Moodle posee una estructura modular. Contiene módulos (entre ellos los recursos y las actividades), filtros y bloques. Estos cuatro elementos son considerados como las funcionalidades principales para la gestión y desarrollo de un curso en la plataforma como expresara Antonio José Moreno Guerrero en su artículo **Moodle**, publicado en el sitio **Observatorio Tecnológico**. (10)

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. (11)

**Módulos de comunicación:** 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:** 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:** 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.

Una actividad es algo que un estudiante hará, que interactúa con otros estudiantes o con el profesor, basado en los recursos que ha utilizado. Existen diferentes tipos de actividades en Moodle 2.x estándar, y se pueden encontrar cuando se activa la edición y se elige el enlace para añadir una actividad (12). Dentro de los principales tipos de actividades se encuentran:

**Tarea:** Permite a los maestros calificar y hacer comentarios sobre archivos subidos y tareas creadas en línea y fuera de línea.

**Chat:** Herramienta de comunicación que permite a los usuarios mantener conversaciones en tiempo real.

**Consulta:** Permite al profesor realizar una pregunta a los alumnos, junto con una lista de opciones o respuestas, de las cuales sólo pueden escoger una.

**Base de Datos:** Es una actividad en la que los alumnos deben incorporar datos mediante un formulario diseñado por el profesor. Las entradas pueden contener texto, imágenes, ficheros

y otros formatos de información que posteriormente podrán compartirse con el resto de compañeros.

**Herramienta externa:** Permite a los participantes interactuar con recursos y actividades de enseñanza compatibles con LTI (*Learning Tools Interoperability*) en otros sitios web.

**Cuestionario:** Actividad autoevaluable, en la cual, la nota se calcula automáticamente, sirve al alumno como autoevaluación y el profesor puede usarlo para realizar un examen al alumno.

**Foro:** Pizarra de mensajes online donde profesores y alumnos pueden colocar nuevos mensajes o responder a otros más antiguos creando así, hilos de conversación.

**Glosario:** Permite a los participantes crear y mantener una lista de definiciones, a semejanza de un diccionario.

**Lección:** Permite al profesor crear una secuencia de páginas con contenido.

**SCORM:** Es un bloque de material web empaquetado de manera que sigue el estándar SCORM de objetos de aprendizaje. Estos paquetes pueden incluir páginas web, gráficos, programas Javascript, presentaciones Flash y cualquier otro elemento que funcione en un navegador web. El módulo SCORM permite cargar fácilmente cualquier paquete SCORM estándar y convertirlo en parte de un curso.

**Encuesta:** Permite al profesor realizar un sondeo de opinión a los alumnos, y analizar las respuestas. Se puede configurar para que las respuestas sean anónimas.

**Wiki:** Conjunto de documentos Web creados gracias a la colaboración de un grupo de usuarios. Básicamente, es una Web que puede ser creada entre los participantes de un curso sin necesidad de que tengan conocimientos de HTML.

**Taller:** Permite al profesor proponer un trabajo a realizar por los estudiantes de forma individual con la característica de que el trabajo de un estudiante puede ser evaluado por otros estudiantes (coevaluación). La evaluación debe hacerse mediante un conjunto de criterios y rúbricas.

Los filtros, como su propio nombre indica, son aplicaciones que analizan el texto que se introduce en las actividades y en los recursos, modificando el resultado final. Los filtros permiten la transformación automática del texto introducido en resultados, a menudo más complejos. Los bloques tienen un carácter específico y permiten agregar funciones extra para fortalecer las capacidades del sistema. (13)



### 1.3.1 Bloques de la plataforma Moodle

La plataforma Moodle posee bloques estándares y no estándares. Los bloques no estándares son opcionales, los cuales deben ser descargados e instalados previamente, mientras que los estándares vienen incluidos por defecto en la plataforma.

Los bloques estándares encierran un conjunto de funcionalidades que robustecen el sistema. Se encuentran en ambos lados de la pantalla, dejando el centro para visualizar los contenidos del sitio. Con los permisos de administración, estos pueden ser ocultados, suprimidos y desplazados lateral o verticalmente.

Existen varios tipos de bloques estándares con funcionalidades muy diversas, algunos trabajan de forma independiente, otros colaboran entre sí. A continuación, se describen los más relevantes (14):

**Bloque “Actividad reciente”:** Muestra los últimos movimientos de un usuario dentro del curso. Entre otras cosas, el alumno podrá observar sus últimas participaciones en los foros de la asignatura o las actividades programadas.

**Bloque “Actividades”:** Permite al usuario un acceso rápido a los diferentes tipos de actividades incluidas en un curso.

**Bloque “Administración”:** Mediante este bloque el usuario puede acceder a diferentes herramientas administrativas en función de su perfil. Desde estas opciones el alumno puede examinar sus calificaciones, cambiar su contraseña o darse de baja de un curso (el profesor se encargará de decidir que opciones estarán disponibles para sus alumnos). Por otra parte, el profesor puede configurar el aspecto de un curso (incorporando bloques, por ejemplo), añadir nuevas actividades o hacer copias de seguridad. Cuanto mayor sea nuestro nivel de privilegios, de más opciones se podrá disponer.

**Bloque “Calendario”:** El calendario de Moodle refleja todas las fechas que puede resultar de interés en el transcurso de un curso (apertura de una nueva unidad didáctica, fecha de entrega de una actividad, entre otros). Esta herramienta también, permite al usuario programar sus propios eventos a modo de agenda personal. El Calendario diferenciará entre cuatro tipos de eventos:

- Eventos generales para todos los cursos.
- Eventos para miembros de un curso.
- Eventos para miembros de un grupo dentro de un curso.

-Eventos programados por el alumno para sí mismo.

**Bloque “Buscar”:** Permite buscar en los foros de un curso información que pueda resultar de utilidad. La información se buscará en función de cadenas de texto introducidas por el usuario.

**Bloque “Cursos”:** Muestra al usuario todos los cursos en los cuales participa ordenados por categorías. Esto le permitirá desplazarse cómodamente entre sus cursos.

**Bloque “Eventos Próximos”:** Recuerda la proximidad de un evento, ya sea un evento general, a nivel de curso o programado por el propio usuario. Los eventos se programan en el bloque Calendario, donde también, se puede configurar el número de eventos que se mostrarán en este bloque o la fecha a partir de la cual un evento puede ser considerado como próximo. (Ver bloque Calendario).

**Bloque “Novedades”:** Muestra un resumen de los últimos anuncios publicados en el “Foro de Novedades” del curso. Este listado de anuncios da la posibilidad de acceder a contenido completo del anuncio o de añadir nuevas entradas. El número de anuncios que aparecerán listados puede configurarse en el formulario de “Configuración del curso”.

**Bloque “Personas”:** El bloque “Personas” agrupa opciones relacionadas con la información recogida por Moodle sobre los participantes de un curso, bien sean alumnos o profesores. Por ejemplo, se puede visualizar la información del resto de alumnos, a qué grupo pertenecen o modificar los datos de la ficha personal.

**Bloque “Usuarios en línea”:** Muestra los usuarios conectados en línea en un curso. Se detectan nuevos usuarios pasado un tiempo (en minutos) configurable por el administrador.

Los bloques apoyan el proceso de enseñanza-aprendizaje, desde el punto de vista de la estructura de la plataforma y se encuentran ubicados en la subcarpeta "blocks" de la misma. Estos profundizan las ventajas de la tecnología e-learning en la educación.

### **1.3.2 Compartición de actividades en Moodle**

La necesidad de obtener información y conocimiento, demanda que actualmente existan personas que lo compartan. La compartición es la acción y el efecto de compartir algo. Compartir según el Diccionario Manual de la Lengua Española es: “usar o tener una cosa en común con otros, dividir en partes una cosa para repartirla entre varios, comunicar a otra u otras personas ideas o sentimientos”. (15)

**Tabla 1** Comparación de soluciones similares.

<b>Soluciones Similares</b>	<b>Características</b>	<b>Compatible con Moodle 2.9.x</b>	<b>Comparte contenido entre cursos de un profesor</b>	<b>Comparte contenido entre cursos de diferentes profesores</b>	<b>Durabilidad en el tiempo del contenido</b>	<b>Publicar contenido en forma de metadatos</b>	<b>Establecer ambiente de colaboración</b>
<b>Funcionalidad Importar</b>	Permite importar y copiar el contenido de un curso completo, o algunos recursos y actividades seleccionados por un profesor a otro curso de él mismo.	Sí	Sí	No	No	No	No
<b>Carrito para Compartir (Sharing Cart)</b>	Duplica las actividades de un curso hacia una biblioteca personal, la que luego, hace copia de estas actividades ,entre varios cursos de un mismo profesor.	Sí	Sí	No	Sí	No	No
<b>Módulo Share v1.0</b>	Módulo que permite compartir actividades entre los cursos de la plataforma Moodle en la versión 2.3.x. Realiza un proceso de búsqueda con el empleo de diferentes filtros sobre las actividades compartidas.	No	Sí	Sí	No	No	No

En la presente investigación el término compartir estará asociado a la acción de ofrecer el conocimiento implícito en las actividades entre los profesores de la plataforma, donde estas, además, podrán ser reutilizadas.

Para llevar a cabo el desarrollo del bloque se realizó un estudio previo, a las soluciones existentes en la plataforma Moodle que permiten la compartición de actividades. Se establecieron diferentes criterios comparativos asociados a las necesidades actuales de los profesores, arrojándose a los resultados expuestos en la tabla 1.

Realizado un análisis a las distintas soluciones similares (ver Tabla 1) se comprueba que estas solo cumplen parcialmente las necesidades existentes. De las soluciones expuestas, el módulo Share v1.0, es el único que permite compartir actividades entre cursos de diferentes profesores, pero presenta significativas deficiencias que imposibilitan su uso en la versión 2.9.x de Moodle. Una de ellas es que modifica el core de Moodle durante su instalación al insertarle código, para poder realizar algunas de sus funcionalidades. No cumple con la filosofía de la plataforma, debido a que este constituye un módulo de tipo actividad (parte activa y colaborativa donde el estudiante interactúa con el profesor) y su comportamiento es el de un bloque; herramienta que facilita el proceso de enseñanza-aprendizaje.

Por las razones expuestas se decide el desarrollo de un bloque compatible con la versión 2.9.x de Moodle, que permita compartir actividades entre cursos de diferentes profesores y que garantice la durabilidad en el tiempo de sus contenidos. Además, que propicie establecer un ambiente de colaboración que contribuya a elevar los niveles de calidad, en la elaboración de actividades y publicar los metadatos de estas hacia repositorios externos.

Con el objetivo de publicar los metadatos de las actividades compartidas hacia repositorios externos, se realiza un análisis de los protocolos para la publicación y recolección de datos.

En la presente investigación el término repositorios externos hace referencia a repositorios institucionales o bibliotecas virtuales. Éstos últimos constituyen sistemas de gestión de información, que preservan y organizan materiales científicos y académicos, como apoyo a la investigación, el aprendizaje y el acceso a la información. (16)

#### **1.4 Protocolo OAI-PMH**

En la búsqueda y recuperación de información existen un variado número de mecanismos para la generación y obtención de metadatos en Internet, el más utilizado es el protocolo OAI-PMH (*Open*

*Archives Initiative Protocol for Metadata Harvesting*), pues sirve como un depósito para almacenar y obtener cualquier tipo de información que se encuentre en cualquier formato electrónico.

Es una sencilla interfaz que permite el acceso a los metadatos de contenidos en formato XML proveniente de distintas fuentes, plataformas y repositorios. Tiene sus raíces en un esfuerzo por realizar el acceso a archivos electrónicos como medio para aumentar la disponibilidad de la comunicación académica.

Este protocolo genera y promueve estándares de interoperabilidad que facilitan la difusión, intercambio y accesibilidad a documentos de diferente naturaleza. Permite almacenar en un solo lugar los metadatos y es allí en donde se realizan las diferentes consultas. No define la creación de los metadatos, ni da los parámetros para realizar una consulta, únicamente se ocupa de la gestión de información. (17)

La interoperabilidad es “la capacidad de un sistema de información de comunicarse y compartir datos, información, documentos y objetos digitales de forma efectiva (con una mínima o nula pérdida de su valor y funcionalidad), con uno o varios sistemas de información (siendo generalmente estos sistemas completamente heterogéneos, distribuidos y geográficamente distantes), mediante una interconexión libre, automática y transparente, sin dejar de utilizar en ningún momento la interfaz del sistema propio”. (18)

OAI-PMH está basado en un modelo cliente/servidor que transmite preguntas y respuestas entre un proveedor de datos, y un proveedor de servicios. Los primeros son los archivos que proporcionan la información y los segundos son los recolectores o servicios que toman los datos, con el objetivo de incorporarles algún valor añadido y presentarlos a los usuarios finales.

Existen tres archivos disponibles en la actualidad que proporcionan información (19):

- **@rchiveSIC** <<http://archivesic.ccsd.cnrs.fr/>> es un proyecto de colaboración entre varias instituciones francesas (universidades y centros de investigación como el CNRS). En estos momentos almacena unos 80 documentos, la mayor parte de ellos en francés. Incluye documentos de áreas relacionadas como Museología.
- **DLIST** (*Digital Library of Information Science and Technology*) <<http://dlist.sir.arizona.edu/>>. Es un archivo creado por la *School of Information Resources and Library Science* y *Arizona Health Sciences Library* (University of Arizona). Almacena más de 100 documentos. Su objetivo es recoger todo tipo de documentos científicos en documentación, pero con dos áreas temáticas de mayor énfasis: materiales educativos y bibliometría. Solamente aceptan documentos en inglés.

- **E-LIS** (*Eprints in Library and Information Science*) <http://eprints.rclis.org>. Es un esfuerzo internacional para crear un archivo multinacional y multilingüe de documentos científicos en las áreas de Biblioteconomía y Documentación. Ha sido financiado parcialmente por el Ministerio de Educación.

También existen registros de los servicios creados utilizando los datos proporcionados por los archivos mencionados (19):

- **ARC** <http://arc.cs.odu.edu/> Es un servicio experimental creado con el objetivo de investigar temas relacionados con la recolección de metadatos siguiendo el protocolo OAI-PMH y cómo hacerlos disponibles a los usuarios. Más que un servicio en sí mismo es un software que podría ser utilizado por instituciones que quieran crear sus propios servicios. El código fuente está disponible en la red de forma gratuita. Ha sido desarrollado por el *Digital Library Research Group de la Old Dominion University*.
- **OAister** <http://oaister.umdl.umich.edu/> Es un proyecto financiado por la fundación Mellon con el objetivo de crear una amplia colección de recursos digitales gratuitos, útiles y que previamente eran de muy difícil acceso y ponerla al alcance de cualquier usuario de la forma más sencilla posible. Todos los recursos tienen el texto completo disponible en la red de forma que siempre se pueda llegar a los contenidos. Recoge datos de todos los archivos conocidos. En total 122 archivos con más de un millón de registros.
- **Perseus** <http://www.perseus.tufts.edu/> Es una biblioteca digital especializada en Humanidades. Su servicio OAI también recupera datos de todos los servicios conocidos.
- **Cyclades** <http://www.ercim.org/cyclades> Es un proyecto financiado por la Unión Europea. Su objetivo no está directamente relacionado con OAI ya que es crear un marco de colaboración entre los investigadores de los centros que participan en el proyecto. Intenta fomentar la colaboración entre los mismos, emitir recomendaciones y crear servicios personalizados.

El protocolo OAI-PMH soporta múltiples formatos para expresar los metadatos, no obstante, requiere que todos los servidores ofrezcan los registros utilizando *Dublin Core* no calificado, codificado en XML.

*Dublin Core* es un modelo de metadatos elaborado y auspiciado por la DCMI (*Dublin Core Metadata Initiative*), una organización dedicada a fomentar la adopción extensa de los estándares interoperables de los metadatos y a promover el desarrollo de los vocabularios especializados de

metadatos para describir recursos. Este sistema de definiciones fue diseñado específicamente para proporcionar un vocabulario de características "base", capaces de proporcionar la información descriptiva básica sobre cualquier recurso, sin que importe el formato de origen, el área de especialización o el origen cultural. (20)

OAI-PMH es en esencia una plataforma simple para acceder a la información bibliográfica disponible en un archivo o repositorio. De esta forma, cualquier institución puede realizar una implementación del protocolo para poner a disposición su propia base de datos(BD) o catálogo, y a la vez consultar otras.

Sin embargo, esta implementación requiere ciertos requisitos, ya sea como proveedor de datos o como proveedor de servicios, que pueden dificultar la utilización de este protocolo (21):

- Metadatos (almacenados en BD).
- Servidor web.
- Programación de interfaz.
- Identificador de archivo/URL Base.
- Formato de metadatos.
- Jerarquía lógica de sets.
- Control del flujo.
- Generador de errores (Respuestas XML).
- Acceso a BD para extraer los metadatos de acuerdo al formato de metadatos requerido.
- Generador XML para crear las respuestas.

La utilidad de los buscadores se circunscribía a páginas estáticas HTML, actualmente se ha incorporado a la búsqueda en el contenido de documentos en PDF, Word y otro tipo de formatos.

Esta novedad se incorpora con el uso de la recolección de metadatos (metadata harvesting) a través del protocolo OAI-PMH posibilitando su implantación en aquellas instituciones que quieran no sólo crear contenidos para la web sino también implantar aplicaciones que favorezcan su localización.

Las características de flexibilidad del OAI-PMH para adaptarse a cualquier ámbito, pudiendo ofrecer información sobre todo tipo de recurso, tanto físico como digital, permiten la diseminación de

información a través de la generación de metabuscadores mediante su uso y por tanto la formación de repositorios de información y de bibliotecas digitales.

¿Qué relación existe con otros protocolos como el Z39.50?

El marco diseñado por OAI es intencionalmente simple con el propósito de proporcionar una mínima complicación para las instituciones que deseen implementarlo. Los protocolos como el Z39.50 tienen una funcionalidad más completa, por ejemplo, tratan cuestiones como el manejo de sesiones, gestión de conjuntos de resultados y permiten la especificación de predicados para filtrar los resultados obtenidos. Además, esta funcionalidad acarrea un incremento en la complejidad de la implementación.

Por lo tanto, se incluye en la propuesta de solución el protocolo OAI-PMH con el objetivo de publicar el contenido de las actividades en forma de metadatos en repositorios externos y garantizar almacenar toda la información del bloque Actividades compartidas.

Para regular el uso de los contenidos o la información y permitir el empleo de recursos a terceros surgen las licencias de derechos de autor, éstas son conocidas como *Copyright*, *Copyleft* y *Creative Commons*.

Las licencias de derechos de autor tienen como propósito fortalecer a los creadores para que sean quienes definan los términos en que sus obras pueden ser usadas, además de qué derechos desean entregar y en qué condiciones lo harán. A diferencia de *Copyright*, que se utiliza para prohibir y restringir los usos de una obra por parte del autor, *Copyleft* garantiza a cada persona que recibe una copia de una obra poder a su vez usar, modificar y redistribuir el propio trabajo y las versiones, permitiendo en ocasiones el uso comercial. (22)

Las licencias *Creative Commons* se crearon con el objetivo de facilitar la divulgación de obras de propiedad intelectual de carácter libre.

Para compartir recursos es necesario tener en cuenta el derecho de autor; Moodle se rige por las licencias *Creative Commons*.

## **1.5 Licencias *Creative Commons*(CC)**

Las licencias *Creative Commons* son mecanismos legales que permiten a los autores de documentos, vídeos, imágenes, música, entre otros, compartir los contenidos que crean a través de la web, especificando de qué forma se puede usar su obra. Así, en lugar de tener “Todos los derechos reservados”, los contenidos pasan a tener “Algunos derechos reservados”, lo cual abre la posibilidad para que los usuarios usen el contenido acogiéndose a los términos especificados en la



licencia (23). Una de sus ventajas es que el autor autoriza el uso de su obra, pero la obra continúa estando protegida.

Las licencias Creative Commons garantizan cuatro libertades; atribución de la obra, no comercial, sin derivados y compartir igual. Las combinaciones de éstas dan lugar a once tipos de licencia distintas, aunque las más habituales son las seis siguientes (24):

**Atribución:** Esta licencia permite a otros distribuir, remezclar, retocar, y crear a partir de su obra, incluso con fines comerciales, siempre y cuando se de crédito por la creación original. Es la más flexible de las licencias ofrecidas. Se recomienda para la máxima difusión y utilización de los materiales licenciados.

**Atribución-SinDerivadas:** Esta licencia permite la redistribución, comercial o no comercial, siempre y cuando la obra circule íntegra y sin cambios, ofreciendo crédito.

**Atribución-NoComercial-SinDerivadas:** Esta licencia es la más restrictiva de las seis licencias principales, permitiendo a otros solo descargar su obra y compartirla siempre y cuando le den su crédito, pero no permite cambiarla de forma alguna ni usarla comercialmente.

**Atribución-NoComercial:** Esta licencia permite a otros distribuir, remezclar, retocar, y crear a partir de su obra de manera no comercial y, a pesar de que sus nuevas obras deben siempre mencionar y mantenerse sin fines comerciales, no están obligados a licenciar sus obras derivadas bajo las mismas condiciones.

**Atribución-NoComercial-CompartirIgual:** Esta licencia permite a otros distribuir, remezclar, retocar, y crear a partir de su obra de modo no comercial, siempre y cuando le den crédito y licencien sus nuevas creaciones bajo las mismas condiciones.

**Atribución-CompartirIgual:** Esta licencia permite a otros remezclar, retocar, y crear a partir de su obra, incluso con fines comerciales, siempre y cuando le den crédito y licencien sus nuevas creaciones bajo las mismas condiciones. Suele ser comparada con las licencias "copyleft" de software libre y de código abierto. Todas las nuevas obras basadas en la suya portarán la misma licencia, así que cualquiera obra derivada permitirán también uso comercial. Esta es la licencia que usa Wikipedia, y se recomienda para materiales que se beneficiarían de incorporar contenido de Wikipedia y proyectos con licencias similares.

El modo más propicio de acceder a determina información y de compartirla con otros; reservando algunos derechos, es con el empleo de las licencias *Creative Commons*. Por lo que en la presente

investigación se hace uso de éstas, pero de forma informativa, con el fin de permitirle a los profesores compartir sus actividades, especificando la forma en que desee que éstas sean usadas.

Para potenciar y motivar el perfeccionamiento de las actividades que se comparten se hace necesario, el empleo en la propuesta de solución de la modalidad gamificación.

## **1.6 Gamificación**

En la actualidad se han diversificado los métodos de enseñanza y las estrategias educativas que se utilizan dentro de las escuelas y otros centros educativos. Una de estas estrategias de enseñanza-aprendizaje es la gamificación.

La gamificación llamado también ludificación consiste en el uso de mecánicas, elementos y técnicas de diseño de juegos en contextos que no son juegos para involucrar a los usuarios, haciéndoles más partícipes de su aprendizaje. Aplica los tres principios básicos en que se basa cualquier tipo de juego: la mecánica del juego, la dinámica y los componentes del juego.

Moodle utiliza una serie de elementos de gamificación para inducir el aprendizaje. A continuación, se hace mención de estos (25):

### **1.6.1 Elementos de gamificación de Moodle**

Las mecánicas de juego permiten involucrar al usuario de manera dedicada y divertida en las actividades que se pretenden realizar. Dentro de éstas podemos encontrar el ganar puntos, subir de nivel, obtener premios, destacar en posiciones de clasificación o triunfar en desafíos y misiones.

Moodle hace empleo del concepto de mecánicas de juegos para definir qué cosas debe hacer un estudiante en un recurso o en una actividad para que se considere como completado. De este modo en cada actividad o recurso debe aparecer una sección nueva al final, en la que se definen las acciones que debe realizar el estudiante para que se considere que lo ha completado. Son distintas para cada actividad. Por ejemplo, en un cuestionario sería: verlo, contestarlo u obtener una puntuación mínima. Además, aparece una opción nueva en la administración del curso llamada "Finalización de curso". Esta opción permite definir qué actividades debe realizar el estudiante para completar el curso.

Otro elemento que emplea Moodle asociado a las mecánicas de juego es el Foro con calificaciones. Los mensajes de los foros pueden ser puntuados por los estudiantes, solamente hay que cambiar los permisos desde el mismo foro, en el bloque Administración->Administración del foro->Permisos. En concreto hay que añadir el rol de estudiante en el permiso "Calificar mensajes".

Para definir la calificación de un mensaje hay varias opciones, algunas serían:

- Si se califica con una escala de un elemento (“Me gusta” o similar), puede ser el número total de calificaciones.
- Si se califica con una escala numérica, puede ser la media de las puntuaciones.
- Con una escala cualitativa (por ejemplo, mal-regular-bien-muy bien), también puede hacerse la media.

Estas opciones se definen en los ajustes de cada foro, en la sección “Calificaciones”.

Las dinámicas del juego es otro elemento de la gamificación, las mismas establecen las motivaciones personales de cada individuo para participar en la actividad gamificada; por ejemplo, las recompensas, el reconocimiento o prestigio, los logros, la competencia y el compañerismo.

Los bloques de resultados de cuestionarios son un ejemplo de dinámicas de juego empleadas en Moodle. Éste muestra las calificaciones más altas de un cuestionario. Permite publicar resultados individuales o en grupo (si se han utilizado grupos en la actividad). En el caso de los resultados individuales, pueden aparecer de forma nominal o anónima.

Otro de los elementos que contiene la gamificación son las mecánicas de juegos, para comprenderlas es necesario conocer los elementos que lo conforman. Algunos de ellos son: los avatares, los puntos acumulados por el participante, el tablero o área de juego, las insignias, las medallas y el status social entre el grupo de jugadores.

Moodle hace uso de esta mediante la creación de badges o insignias y otorgarlas a los estudiantes de un curso para reconocer su trabajo. La concesión de estas “medallas” puede ser manual, cuando es el profesor directamente el que decide a quien se entrega, o automática. En este segundo caso, el estudiante consigue la insignia cuando “completa” una o varias actividades. Las insignias conseguidas quedan en la parte pública del perfil de cada estudiante. Además, si tuviera una mochila externa, puede exportar las medallas logradas y traer a su perfil medallas conseguidas en otros lugares.

Para motivar a que los profesores compartan sus actividades y éstas sean utilizadas en sus cursos, se propone incorporar en la propuesta de solución, técnicas de gamificación asociadas a las mecánicas de juego.

Una de las técnicas a emplear, es “subir de nivel”, expresada en el uso de un ranking de actividades compartidas en el repositorio. Se realiza con el objetivo de reconocer las actividades compartidas más usadas y calificadas positivamente por los usuarios de la plataforma.

Entre los elementos de gamificación que usa Moodle, se propone aplicar la forma de los Foro con calificaciones, a través del empleo de las opciones (Me gusta o No me gusta) para calificar una actividad compartida. Esta forma de gamificación contribuye a que el profesor pueda emitir un criterio sobre la actividad de manera rápida y concisa.

## **1.7 Tecnologías y herramientas a utilizar en la solución**

Una vez analizadas las tecnologías y herramientas que se tuvieron en cuenta en el desarrollo del módulo Share v1.0 para la versión 2.3.x de Moodle y las definidas a utilizar en el desarrollo de contribuciones para la plataforma Moodle en su sitio oficial (<http://moodle.org/>), se determina emplear las siguientes en el desarrollo de la presente investigación: XML, PHP, Apache, PostgreSQL, CSS, Java Script y HTML. Además, se utilizan las librerías YUI y JQuery de JavaScript para solucionar determinados escenarios que surgen durante la implementación.

### **1.7.1 Tecnologías del lado del servidor**

La programación del lado del servidor es una tecnología que consiste en el procesamiento de una petición de un usuario mediante la interpretación de un script en el servidor web para generar páginas HTML dinámicamente como respuesta. Un ejemplo eficaz de lenguaje que sea reconocido, ejecutado e interpretado por el propio servidor y que se envían al cliente en un formato comprensible para él, es PHP. (26)

#### **1.7.1.1 *Preprocessed Hypertext Pages***

*Preprocessed Hypertext Pages* (PHP) es un lenguaje interpretado de propósito general ampliamente usado y que está diseñado especialmente para desarrollo web y puede ser embebido dentro de código HTML.

Este lenguaje posee propósito general y su naturaleza de código abierto (open-source) lo muestra como una alternativa de fácil acceso para todos. Es sencillo para su aprendizaje, soporta la orientación a objetos y es capaz de conectarse con la mayoría de los manejadores de bases de datos que se utilizan en la actualidad. (27)

Se ejecuta en un servidor web, tomando el código en PHP como su entrada y creando páginas web como salida. Puede ser desplegado en la mayoría de los servidores web y en casi todos los sistemas operativos y plataformas sin costo alguno. Posibilita la generación de archivos PDF, Flash, así como imágenes en diferentes formatos. Permite la conexión a diferentes tipos de servidores de bases de datos tales como MySQL, Postgres, Oracle, ODBC, DB2, Microsoft SQL Server, Firebird y SQLite.

PHP también tiene la capacidad de ser ejecutado en la mayoría de los sistemas operativos, tales como UNIX (y de ese tipo, como Linux o Mac OS X) y Windows. Además, puede interactuar con los servidores de web más populares ya que existe en versión CGI, módulo para Apache, e ISAPI (28). La versión a utilizar será la 5.4.19.

La unión de XML y PHP puede funcionar de una forma muy similar a como consigue trabajar PHP con una base de datos. En la presente investigación se hace uso de la unión de estas dos tecnologías.

#### **1.7.1.2 Metalenguaje de marcas extensible**

XML proviene de *Extensible Markup Language* (Lenguaje de Marcas Extensible). Se trata de un metalenguaje (un lenguaje que se utiliza para decir algo acerca de otro) extensible de etiquetas que fue desarrollado por el *World Wide Web Consortium* (W3C), una sociedad mercantil internacional que elabora recomendaciones para la *World Wide Web*.

El XML es una adaptación del SGML (*Standard Generalized Markup Language*), un lenguaje que permite la organización y el etiquetado de documentos. Esto quiere decir que el XML no es un lenguaje en sí mismo, sino un sistema que permite definir lenguajes de acuerdo a las necesidades.

El XHTML, el MathML y el SVG son algunos de los lenguajes que el XML tiene la capacidad de definir. Las bases de datos, los documentos de texto, las hojas de cálculo y las páginas web son algunos de los campos de aplicación del XML. El metalenguaje aparece como un estándar que estructura el intercambio de información entre las diferentes plataformas.

Los expertos señalan varias ventajas que derivan de la utilización del XML, como por ejemplo que: es extensible (se pueden añadir nuevas etiquetas tras el diseño del documento), su analizador es estándar (no requiere de cambios para cada versión del metalenguaje), facilita el análisis y el procesamiento de los documentos XML creados por terceros. La validez de los documentos (i.e. que su estructura sintáctica se encuentre desarrollada correctamente) depende la relación especificada entre los distintos elementos a partir de una definición o documento externo.

XML presenta una serie de ventajas muy atractivas para los desarrolladores, especialmente porque permite relacionar aplicaciones de diferentes lenguajes y plataformas; sin embargo, esto mismo puede ser visto como un arma de doble filo, dado que no incentiva la búsqueda de compatibilidad. La universalidad que persigue XML puede no llegar jamás si en lugar de aprovecharlo para resolver problemas, se generan nuevos sabiendo que tendrán una solución. (29)

En el desarrollo de cualquier aplicación basada en tecnologías como XML y PHP se hace necesaria la utilización de un servidor web para su desarrollo. Apache constituye uno de los más usados en la actualidad debido a sus flexibilidad, rapidez y eficiencia.

#### **1.7.1.3 Servidor web Apache**

El servidor Apache es un servidor web HTTP de código abierto para plataformas Unix (BSD, GNU/Linux, entre otros), Windows, Macintosh y otras, que implementa el protocolo HTTP/1.1 y la noción de sitio virtual. Fue desarrollado dentro del proyecto HTTP Server (httpd) de la *Apache Software Foundation*. Apache es altamente configurable, admite bases de datos de autenticación y negociado de contenido, aunque carece de una interfaz gráfica que ayude en su configuración. (30)

Apache ha incorporado en su soporte a una amplia gama de lenguajes de programación web, como Perl, PHP y Python. También incluye soporte "SSL" y "TLS". Estos son los protocolos para enviar datos encriptados a través de Internet, de gran importancia para aquellas aplicaciones que requieren privacidad.

Una de sus principales ventajas es su portabilidad, debido a que se puede instalar en una amplia variedad de servidores y sistemas operativos. Apache es capaz de ejecutarse en todas las versiones del sistema operativo UNIX. Linux es compatible, así como los sistemas operativos Windows NT y MacOS. La versión a utilizar en la presente investigación es la 2.4.4.

En general, Apache es uno de los sistemas de servidores más adaptables y disponibles en la actualidad, por su ejecución en una amplia gama de entornos técnicos. (31)

#### **1.7.1.4 Sistema gestor de base de datos**

Un Sistema Gestor de Base de Datos es un tipo de software muy específico, dedicado a servir de interfaz entre la base de datos, el usuario y las aplicaciones que la utilizan, y controlar la entrada y salida de datos de la base de datos, manteniendo la integridad de la misma. La importancia de su uso es que proporcionan a los usuarios la capacidad de almacenar datos en la base de datos, acceder a ellos y actualizarlos. (32)

En el mundo del software libre, siempre que se necesita algo, tarde o temprano se implementa. Un ejemplo claro es MySQL, gestor muy usado en la web o PostgreSQL.

#### **1.7.1.4.1 PostgreSQL**

PostgreSQL está ampliamente considerado como el sistema de bases de datos de código abierto más avanzado del mundo. Es un Sistema de Gestión de Bases de Datos Objeto-Relacionales (ORDBMS), debido a que incluye características de la orientación a objetos, como puede ser la herencia, tipos de datos, funciones, restricciones, disparadores, reglas e integridad transaccional. (33)

#### **Características de PostgreSQL**

PostgreSQL proporciona un gran número de características que normalmente sólo se encontraban en las bases de datos comerciales tales como DB2 u Oracle. La siguiente es una breve lista de algunas de esas características, a partir de PostgreSQL 7.1.x.

- Aproxima los datos a un modelo objeto-relacional, y es capaz de manejar complejas rutinas y reglas. Ejemplos de su avanzada funcionalidad son consultas SQL declarativas, control de concurrencia multi-versión, soporte multi-usuario, transactions, optimización de consultas, herencia, y arrays.
- Soporta operadores, funcionales, métodos de acceso y tipos de datos definidos por el usuario.
- PostgreSQL soporta integridad referencial, la cual es utilizada para garantizar la validez de los datos de la base de datos.
- Tiene soporte para lenguajes procedurales internos, incluyendo un lenguaje nativo denominado PL/pgSQL. Este lenguaje es comparable al lenguaje procedural de Oracle, PL/SQL. Otra ventaja de PostgreSQL es su habilidad para usar Perl, Python, o TCL como lenguaje procedural embebido.
- PostgreSQL usa una arquitectura proceso-por-usuario cliente/servidor. Esta es similar al método del Apache 1.3.x para manejar procesos. Hay un proceso maestro que se ramifica para proporcionar conexiones adicionales para cada cliente que intente conectar a PostgreSQL.

#### 1.7.1.4.2 MySQL

Es un sistema de gestión de base de datos relacional, multihilo y multiusuario con más de seis millones de instalaciones. Se ofrece bajo la GNU GPL para cualquier uso compatible con esta licencia, pero para aquellas empresas que quieran incorporarlo en productos privativos deben comprar a la empresa una licencia específica que les permita este uso. (34)

Entre las características más significativas de MySQL se encuentran:

- MySQL aprovecha la potencia de sistemas multiprocesador, gracias a su implementación multihilo.
- Funciona sobre múltiples plataformas (AIX, BSD, GNU/Linux, MacOS X, NetBSD, OpenBSD).
- El código MySQL se prueba con Purify (un detector de memoria perdida comercial) así como con Valgrind, una herramienta GPL.
- Soporte completo para UNICODE.
- Múltiples motores de almacenamiento.
- Gestión de usuarios y *passwords*, manteniendo un muy buen nivel de seguridad en los datos.

El sistema gestor de bases de datos seleccionado para el desarrollo de la aplicación es PostgreSQL 9.4.1 debido a que es considerado entre los ORDBMS de licencia libre como uno de los más completos por poseer todas las funcionalidades de un gestor de bases de datos privativo. Además, en la UCI existe una comunidad de PostgreSQL cubano en la que se trabaja en el desarrollo y explotación de este gestor. Entre otras de las ventajas determinantes es su documentación bien detallada y organizada.

#### 1.7.2 Tecnologías del lado del cliente

Las tecnologías del lado del cliente, tal como su nombre lo indica, son el modelo de aplicación distribuida en el que las tareas se reparten entre los proveedores de recursos o servicios, llamados servidores, y los demandantes, llamados clientes. Estas tecnologías emplean disímiles lenguajes tales como HTML y JavaScript para facilitar las posibilidades del cliente en la creación de aplicaciones dinámicas de Internet al aprovechar el poder de procesamiento local de las computadoras y los dispositivos. (35)



A continuación, se mencionan las tecnologías del lado del cliente a emplear en la presente investigación:

### **1.7.2.1 Lenguaje de Marcado de Hipertexto**

El Lenguaje de Marcado de Hipertexto (*HyperText Markup Language, HTML*) es el predominante para elaborar las páginas web. Se trata de un formato no propietario basado en SGML6; se puede crear y procesar una amplia gama de herramientas, desde simples editores de texto a sofisticadas herramientas de creación en modo WYSIWYG7. Básicamente HTML permite estructurar la página web utilizando etiquetas predefinidas que marcan el inicio y el final de cada elemento además de insertar hipertextos (enlaces) que comunican la página web con otras páginas. (36)

Actualmente, HTML se usa para estructurar y proporcionar una descripción semántica a las páginas web, que son las funciones para las que se creó en un principio. Por tanto, no determina ni la apariencia ni el comportamiento interactivo de las páginas, sino solamente su estructura y el significado de sus elementos (37). Debido a las novedosas herramientas que incorpora para el desarrollo web, se considera la selección de HTML5 como el lenguaje para realizar el maquetado de la propuesta de solución.

### **1.7.2.2 Hojas de estilo en cascada**

El denominado Hojas de Estilo en Cascada (*Cascading Style Sheets*) es un estándar para la aplicación de estilos que se utiliza para definir el aspecto de todos los contenidos como es el color, tamaño y tipo de letra de los párrafos de texto la separación entre titulares y párrafos; y la tabulación con la que se muestran los elementos de una lista. Es un mecanismo simple que permite definir una serie de estilos para luego aplicarlos a una página o, incluso, a un grupo determinado. Su objetivo es separar la estructura de un documento de su presentación, haciéndose imprescindible para crear páginas web complejas. (36)

La selección de CSS como lenguaje para escribir el estilo visual de la propuesta de solución se hizo teniendo en cuenta que el mismo posee un amplio uso por parte de maquetadores, diseñadores y desarrolladores web. La versión a utilizar será la 3.0.

### **1.7.2.3 JavaScript**

El lenguaje de programación JavaScript es un lenguaje interpretado basado en guiones que son integrados directamente en el código HTML. Es utilizado en el desarrollo web: el código es transferido al cliente para que este lo interprete al cargar la página. (38)

JavaScript nace producto a la necesidad de facilitarle a los autores de sitio web crear páginas que permitan el intercambio con los usuarios. Esto fue debido a que se necesitaban crear webs de mayor complejidad y HTML solo permitía elaborar páginas estáticas donde se podía mostrar textos con estilos, pero no se podía interactuar con los usuarios. (39)

Entre las características de este lenguaje se puede mencionar que es un lenguaje basado en acciones que posee menos restricciones. Gran parte de su programación está centrada en describir objetos, escribir funciones que respondan a movimientos del mouse, aperturas, utilización de teclas, cargas de páginas, entre otros.

Es un lenguaje de programación interpretado, por lo que no es necesario compilar los programas para ejecutarlos. Permite interactuar con el navegador de manera eficaz, proporcionando a las páginas web dinamismo y vida. No es un lenguaje orientado a objetos, debido a que no dispone de herencia. Es más bien un lenguaje basado en prototipos, script u orientado a documento. Gracias a su compatibilidad con la mayoría de los navegadores modernos, es el lenguaje de programación del lado del cliente más utilizado. (40)

#### **1.7.2.4 JQuery**

Es un *framework* o librería JavaScript, implementa una serie de clases (Programación Orientada a Objetos) que permite simplificar la manera de interactuar con los documentos HTML, manipular el árbol DOM (*Document Object Manager*), manejar eventos, desarrollar animaciones y agregar interacción con la tecnología AJAX a páginas Web. Es un producto bien documentado y con un gran equipo de desarrolladores a cargo de la mejora y actualización del *framework*.

Al igual que otras librerías, ofrece una serie de funcionalidades basadas en JavaScript que de otra manera requerirían de mucho más código, es decir, con las funciones propias de esta biblioteca se logran grandes resultados en menos tiempo y espacio (41). La versión a utilizar es la 1.11.2.

#### **1.7.2.5 YUI**

El *framework* o librería YUI (*Yahoo! User Interface*) es un conjunto de utilidades y controles escritos en JavaScript que se utilizan para crear aplicaciones web dinámicas complejas. Incluye varias utilidades relacionadas con CSS, por lo que también se considera un *framework* CSS. La arquitectura de YUI es modular que permite enlazar únicamente los archivos que necesita cada página y no todos los archivos de la librería, lo que mejora el rendimiento de la aplicación.

YUI se ejecuta correctamente en todos los navegadores modernos e incluso en algún navegador obsoleto. Es extremadamente modular con diferentes componentes, funciones y plugins. Se caracteriza por ser altamente extensible, rápido, modular.

Desde la versión 2.4.x de Moodle se incluyó SimpleYUI para exponer varias características de forma nativa sin necesidad de especificar las funciones a utilizar (42). La versión a utilizar es la 3.17.2.

## **1.8 Herramientas de desarrollo**

El desarrollo de software es un proceso complejo que requiere de herramientas competentes para lograr resultados satisfactorios. En el presente epígrafe se hace referencia a las herramientas empleadas para el desarrollo de la propuesta de solución y las características por las que fueron seleccionadas. Se determinó el uso del Entorno de Desarrollo Integrado(IDE) NetBeans versión 8.0 y la herramienta CASE Visual Paradigm versión 8.0.

### **1.8.1 Entorno de Desarrollo Integrado (IDE) NetBeans**

NetBeans IDE es un entorno de desarrollo, una herramienta para que los programadores puedan escribir, compilar, depurar y ejecutar programas. Está escrito en Java, pero puede servir para cualquier otro lenguaje de programación. Es un producto libre y gratuito sin restricciones de uso.

Todas las funciones del IDE son provistas por módulos. Cada módulo provee una función bien definida, tales como el soporte de Java, edición, o soporte para el sistema de control de versiones. NetBeans contiene todos los módulos necesarios para el desarrollo de aplicaciones Java en una sola descarga, permitiéndole al usuario comenzar a trabajar inmediatamente. Los desarrolladores pueden crear rápidamente sitios web, aplicaciones de escritorio y aplicaciones móviles utilizando la plataforma Java, así como JavaFX, PHP, JavaScript y Ajax, Ruby, Ruby on Rails, Groovy, Grails y C / C + +. (43)

Realiza un trazo de los valores tomados por las variables en todo el proceso de ejecución. Es posible colocar puntos de parada (*breakpoints*) en los scripts y realizar las acciones típicas de depuración.

Por las características antes expuestas y por ser una herramienta que ofrece una amplia documentación y formación de recursos, se considera ideal para llevar a cabo la implementación de la aplicación. Permite la integración con la librería JQuery de JavaScript y con el lenguaje PHP; lenguaje sobre el cual se implementan funcionalidades en la plataforma Moodle.

### **1.8.2 Lenguaje de Modelado**

El Lenguaje de Modelado (UML) “*es un lenguaje de modelado visual que se usa para especificar, visualizar, construir y documentar artefactos de un sistema de software*”. UML cuenta con un conjunto de notaciones y diagramas para modelar sistemas orientados a objetos y puede usarse en las diferentes etapas del ciclo de vida del desarrollo del software. La especificación de UML no define un proceso estándar, pero está ideado para ser útil en un proceso de desarrollo iterativo. Pretende dar apoyo a la mayoría de los procesos de desarrollo orientados a objetos (44).

La capacidad de UML para modelar cualquier sistema de software y el hecho de que haya sido adoptado por el Grupo de Gestión de Objetos(OMG) como un estándar desde noviembre de 1997, permitieron determinarlo como el lenguaje de modelado para desarrollar la propuesta de solución. La versión a utilizar es la 2.5.

### **1.8.3 Herramientas CASE**

Las herramientas CASE (*Computer Aided Software Engineering*), traducido al español como Ingeniería de Software Asistida por Computadora, son diversas aplicaciones informáticas destinadas a aumentar la productividad en el desarrollo de software reduciendo el coste de las mismas en términos de tiempo y de dinero. Estas herramientas ayudan en todos los aspectos del ciclo de vida de desarrollo del software en tareas como el diseño de proyectos, cálculo de costes, implementación de parte del código automáticamente con el diseño dado, compilación automática, documentación o detección de errores entre otras. A continuación se ofrece una breve descripción de la herramienta CASE que se va a utilizar (45):

#### **1.8.3.1 Visual Paradigm**

Visual Paradigm es una herramienta CASE que soporta el modelado mediante UML. Se caracteriza principalmente por su robustez, usabilidad y portabilidad. Proporciona asistencia a los analistas, ingenieros de software y desarrolladores, durante todo el ciclo de vida de desarrollo de un Software: análisis y diseño orientados a objetos, construcción, pruebas y despliegue. (46)

Esta herramienta permite realizar ingeniería tanto directa como inversa, a partir de un modelo relacional en SQLServer, MySQL, PostgreSQL, es capaz de desplegar todas las clases asociadas a las tablas (siguiendo el patrón de diseño Una Clase-Una Tabla). Soporta múltiples usuarios trabajando sobre el mismo proyecto, permite control de versiones y genera la documentación automáticamente en formatos como web o PDF. Es libre, multiplataforma y se tiene su licencia en nuestra universidad.

A modo de resumen la herramienta presenta diversas características tales como (46):

- Tiene soporte para UML versión 2.1.
- Permite el modelado colaborativo con CVS y Subversion.
- Realiza la ingeniería inversa. De código a modelo y de código a diagrama.
- Realiza la generación de código. De modelo a código y de diagrama a código.
- Posee el editor de detalles de Casos de Uso. Entorno todo-en-uno para la especificación de los detalles de los Casos de Uso, incluyendo la especificación del modelo general y de las descripciones de los Casos de Uso.
- Realiza los diagramas de flujo de datos.
- Permite la generación de bases de datos que es la transformación de diagramas de Entidad-Relación en tablas de base de datos.
- Permite la ingeniería inversa de bases de datos desde los Sistemas Gestores de Bases de Datos (DBMS) existentes hacia los diagramas de Entidad-Relación.
- Admite la importación y exportación de ficheros XML.

Con el fin de gestionar el proceso de desarrollo del software de forma más eficiente surgen las metodologías de desarrollo de software.

## 1.9 Metodología de desarrollo de software

El Proceso Unificado Ágil de Scott Ambler o *Agile Unified Process* (AUP) en inglés es una versión simplificada del Proceso Unificado de Rational (RUP). Este describe de una manera simple y fácil de entender la forma de desarrollar aplicaciones de software de negocio usando técnicas ágiles y conceptos que aún se mantienen válidos en RUP.

AUP aplica técnicas ágiles incluyendo Desarrollo Dirigido por Pruebas, Modelado Ágil, Gestión de Cambios Ágil, y Refactorización de Base de Datos para mejorar la productividad. (47)

Al igual que en RUP, en AUP se establecen cuatro fases que transcurren de manera consecutiva.

### 1.9.1 Fases AUP

**Inicio:** El objetivo de esta fase es obtener una comprensión común cliente-equipo de desarrollo del alcance del nuevo sistema y definir una o varias arquitecturas candidatas para el mismo.

**Elaboración:** El objetivo es que el equipo de desarrollo profundice en la comprensión de los requisitos del sistema y en validar la arquitectura.

**Construcción:** Durante la fase de construcción el sistema es desarrollado y probado al completo en el ambiente de desarrollo.

**Transición:** El sistema se lleva a los entornos de preproducción donde se somete a pruebas de validación y aceptación y finalmente se despliega en los sistemas de producción.

La UCI, siguiendo el paradigma que toda metodología debe ser adaptada a las características de cada proyecto (equipo de desarrollo, recursos), decide hacer una variación de la metodología AUP, de forma tal que se adapte al ciclo de vida definido para la actividad productiva de la Universidad. Ésta plantea que de las 4 fases que propone AUP (Inicio, Elaboración, Construcción, Transición) se tome para el ciclo de vida de los proyectos de la UCI mantener la fase de Inicio, pero modificando el objetivo de la misma, unificando las restantes 3 fases de AUP en una sola, a la que llama Ejecución, la cual posteriormente se incorpora a la fase de Cierre.

### 1.9.2 Disciplinas AUP

AUP propone 7 disciplinas (Modelo, Implementación, Prueba, Despliegue, Gestión de configuración, Gestión de proyecto y Entorno), se decide para el ciclo de vida de los proyectos de la UCI tener 8 disciplinas, pero a un nivel más atómico que el definido en AUP. Los flujos de trabajos: Modelado de negocio, Requisitos y Análisis y diseño están unidos en la disciplina Modelo, en la variación para la UCI se consideran a cada uno de ellos disciplinas. Se mantiene la disciplina Implementación, en el caso de Prueba se desagrega en 3 disciplinas: Pruebas Internas, de Liberación y Aceptación y la disciplina Despliegue se considera opcional. Las restantes 3 disciplinas de AUP asociadas a la parte de gestión para la variación UCI se cubren con las áreas de procesos que define CMMI-DEV v1.3 para el nivel 2, serían CM (Gestión de la configuración), PP (Planeación de proyecto) y PMC (Monitoreo y control de proyecto). (48)

En consecuencia, a que el equipo de trabajo es pequeño y la duración del proyecto solo enmarca pocos meses de esfuerzo de desarrollo, se decide utilizar la metodología ágil AUP, que además es la definida por el proyecto Moodle. Ésta posibilita trabajar con precisión y calidad, perfeccionado el

software en cualquier momento del desarrollo. La documentación que propone contribuirá a lograr un mejor entendimiento del sistema por parte del equipo de desarrollo.

## **Conclusiones parciales**

Luego de realizar un análisis sobre los medios empleados por la plataforma Moodle para lograr la compartición de actividades, se concluye que:

- Las soluciones similares estudiadas solo cumplen parcialmente las necesidades asociadas al proceso de compartir actividades en la plataforma Moodle, por lo que se determina el desarrollo de un bloque que cumpla con las necesidades existentes.
- Para almacenar la información del bloque y mostrar el contenido de las actividades en forma de metadatos, se incluye en la propuesta de solución el uso del protocolo OAI-PMH.
- Para la elaboración de la propuesta de solución fueron seleccionadas las siguientes tecnologías: XML, PHP, Apache, PostgreSQL, CSS, Java Script y HTML.
- Por las ventajas que brinda NetBeans 8.0 a los desarrolladores al presentar un soporte completo para PHP 5.4 y su condición de software libre, se selecciona como IDE para el desarrollo de la propuesta de solución.

## CAPÍTULO 2: ANÁLISIS Y DISEÑO DEL BLOQUE

---

En el presente capítulo se procede a realizar la descripción de la propuesta de solución. Se elabora el modelo conceptual con la correspondiente descripción de cada uno de sus elementos. Se describen las características principales del sistema mediante los requisitos funcionales y no funcionales, los cuales son modelados en las historias de usuario. Por último, se obtienen los artefactos generados durante la fase de análisis y diseño de la propuesta de solución.

### 2.1 Propuesta de solución

Para dar solución al problema planteado se determina el desarrollo de un bloque para la plataforma Moodle en la versión 2.9.x, que permita compartir actividades creadas anteriormente en los cursos, por los profesores que interactúen con la plataforma.

El bloque puede ser configurado, permitiéndole al usuario seleccionar los tipos de actividades que se desee compartir en la plataforma y para repositorios externos. En él se visualizarán notificaciones cuándo: una actividad ha sido modificada, presenta nuevos comentarios o ha subido de ranking. Presentará un repositorio donde se almacenarán las actividades compartidas por los diferentes usuarios y una sección que contendrá un listado de actividades pendientes a calificar.

Incorpora al final del menú de edición de cada actividad el botón “Compartir actividad” que conduce a un formulario donde se deben insertar los datos de la actividad a compartir. El formulario muestra por defecto, que la actividad se comparta para el profesor, pero brinda la posibilidad de que la misma sea compartida para todos los profesores de la plataforma y/o para repositorios externos.

Se hace uso de diferentes técnicas de gamificación para motivar al usuario a la creación y compartición de actividades. Una de ellas es “subir de nivel”; esta se evidencia con la creación de un ranking para las actividades compartidas en el repositorio. La posición del ranking se asigna en correspondencia a la cantidad de veces que una actividad es utilizada en un curso y calificadas positivamente por diferentes usuarios de la plataforma.

Otra de las técnicas empleadas es el “foro debate”, donde el usuario puede realizar un comentario a cada actividad compartida en el repositorio y establecer un foro debate en la página del foro que abre cada actividad. Se incluye además la técnica de calificación, en la que cada actividad compartida puede ser calificada a través de la selección de las opciones “Me gusta” y “No me gusta”.



Todos los usuarios podrán previsualizar y utilizar en su curso las actividades compartidas, pero tendrán acceso a eliminar, editar y poner visible las actividades, solo aquellos usuarios que las hayan creado.

El sistema brindará la posibilidad de mostrar en el repositorio el listado de actividades compartidas por diferentes usuarios, a través de la realización de una búsqueda. Ésta se puede ejecutar de manera básica, donde el usuario introduce el nombre de la actividad y palabras claves. Pero si desea agilizar el proceso (búsqueda avanzada), podrá aplicar diferentes filtros sobre la actividad compartida. El resultado de la búsqueda, se puede ordenar teniendo en cuenta diferentes criterios. El usuario tiene la opción de escoger la cantidad de actividades por página que desee visualizar en el repositorio, aunque por defecto se muestran cinco resultados.

Se podrá exportar las actividades que se encuentran compartidas en el repositorio e importar en repositorio, el repositorio adquirido de otra plataforma.

## 2.2 Modelo conceptual

Un Modelo conceptual es una representación visual de las clases conceptuales u objetos del mundo real en un dominio de interés. También se les denomina modelos de dominio y modelos de objetos de análisis. No es una descripción del diseño del software. (49)

### Definición de las clases del modelo conceptual

**Moodle:** Sistema gestor de aprendizaje donde el usuario accede para obtener conocimientos y compartir actividades entre cursos, creadas en la plataforma.

**Bloque Actividades compartidas:** Bloque que permitirá a profesores compartir actividades entre los cursos de la plataforma Moodle en la versión 2.9.x.

**Curso:** Conjunto de contenidos referentes a una materia, son creados y organizados por los administradores del sitio y dirigidos por los usuarios que se establezcan como profesores del curso, de manera tal que puedan satisfacer las necesidades existentes.

**Actividad:** Es considerada como la parte activa del curso porque es donde el usuario da solución a sus tareas mediante la participación en foros, chats, cuestionarios, entre otras.

**Actividad compartida:** Constituye la actividad que será visible en el repositorio para ser utilizada por los profesores en la plataforma como actividad compartida.

**Actividad pendiente a calificar:** Representación de una actividad compartida que aún no han sido calificada.

**Foro:** Espacio que permite a los profesores valorar las actividades compartidas de los cursos y establecer un ambiente de colaboración.

**Comentario:** Es una apreciación escrita que realiza el profesor sobre una actividad compartida.

**Notificaciones:** Conjunto de mensajes que son enviados a los creadores de las actividades compartidas, especificando los cambios que han sido aplicados a las actividades que compartieron.

**Repositorio:** Sitio centralizado donde se almacenan todas las actividades compartidas por los profesores en la plataforma.

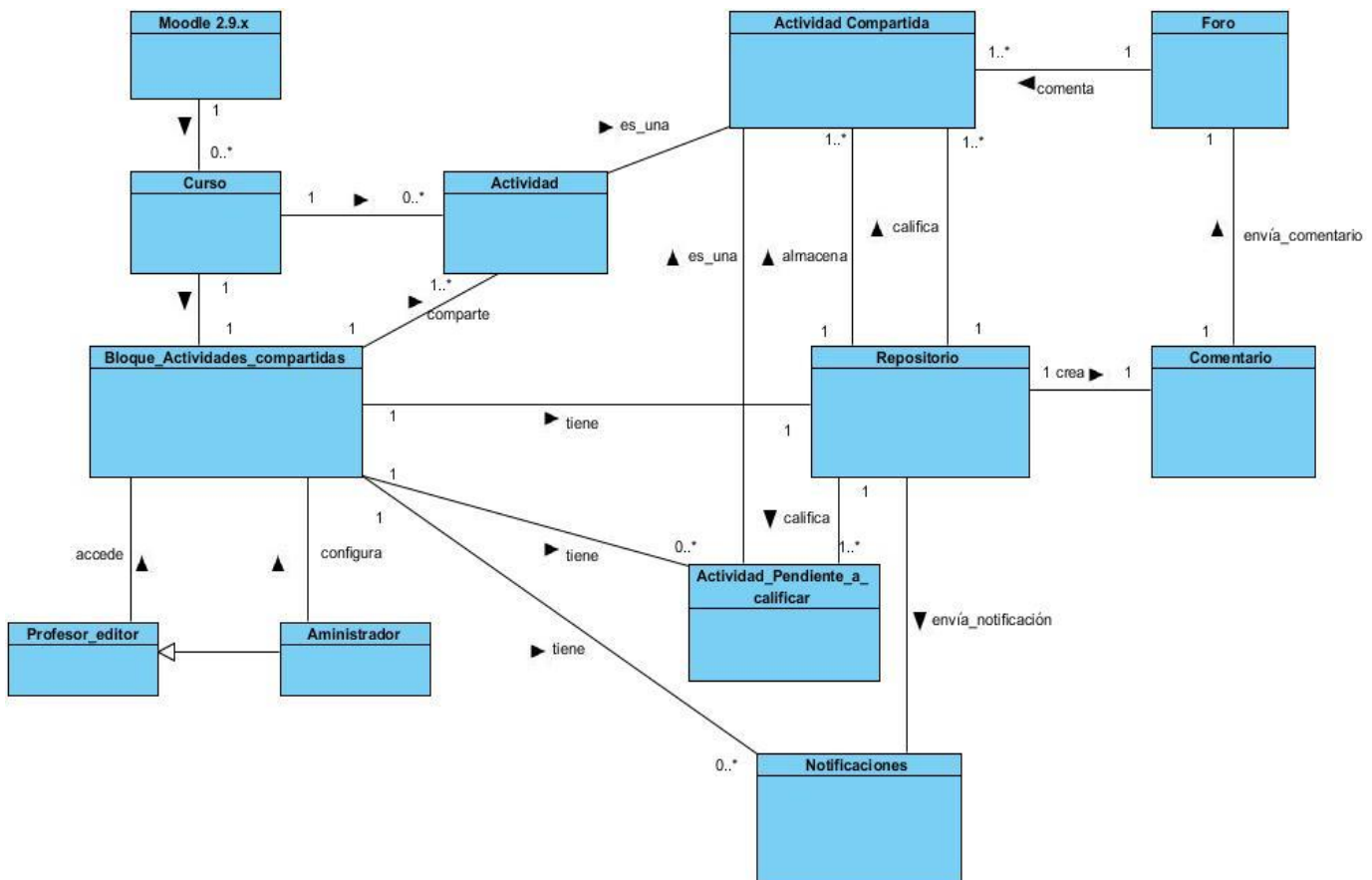


Figura 1 Modelo Conceptual.

**Profesor editor:** Es el encargado de realizar las funcionalidades que brinda el bloque Actividades compartidas excepto instalar, desinstalar y configurar.

**Administrador:** Es el encargado de instalar, desinstalar, configurar y realizar las demás funcionalidades que brinda el bloque Actividades compartidas.

## 2.3 Especificación de requisitos

El levantamiento de los requisitos constituye una de las etapas más importantes en el proceso de ingeniería de requisitos. Para lograr el éxito durante el desarrollo de esta etapa, se realiza un trabajo en conjunto entre los participantes y los desarrolladores con el objetivo de identificar el problema, proponer elementos de solución, negociar diferentes enfoques y especificar un conjunto preliminar de requisitos para la solución. (50)

### 2.3.1 Requisitos funcionales

“Los requisitos funcionales son declaraciones de los servicios que debe proporcionar el sistema, de la manera en que este debe reaccionar a entradas particulares y de cómo se debe comportar en situaciones particulares” (51). A continuación, se describen los requisitos funcionales (RF) del sistema:

**RF1:** Instalar bloque Actividades compartidas: El sistema debe permitirle al usuario instalar el bloque Actividades compartidas en la plataforma Moodle.

**RF2:** Configurar bloque Actividades compartidas: El sistema debe permitirle al usuario configurar el bloque Actividades compartidas.

**RF3:** Desinstalar bloque Actividades compartidas: El sistema debe permitirle al usuario desinstalar el bloque Actividades compartidas en la plataforma Moodle.

**RF4:** Exportar repositorio de actividades compartidas: El sistema debe permitirle al usuario exportar las actividades que se encuentran compartidas, en el repositorio, para utilizar en otra plataforma.

**RF5:** Importar repositorio de actividades compartidas: El sistema debe permitirle al usuario importar en repositorio, el repositorio adquirido de otra plataforma.

**RF6:** Mostrar repositorio de actividades compartidas: El sistema debe permitirle al usuario visualizar en el repositorio las actividades compartidas ordenadas a través del ranking y ubicar al usuario en que página se encuentra de la cantidad que se visualizan.

**RF7:** Escoger cantidad de resultados por página: El sistema debe permitirle al usuario escoger la cantidad de resultados por página que desee visualizar en el repositorio.

**RF8:** Compartir actividad: El sistema debe permitirle al usuario compartir una actividad para todos los usuarios de la plataforma.

**RF9:** Buscar actividad compartida(AC):

**RF9.1:** Buscar AC (básica): El sistema debe permitirle al usuario realizar una búsqueda básica de actividades compartidas a través de la inserción del nombre o palabras clave de la actividad.

**RF9.2:** Buscar AC (avanzada): El sistema debe permitirle al usuario realizar una búsqueda avanzada de actividades compartidas atendiendo a los siguientes criterios de búsqueda: categoría, curso, actividad, autor y licencia.

**RF10:** Ordenar búsqueda de AC: El sistema debe permitirle al usuario ordenar la búsqueda de actividades compartidas atendiendo a los siguientes criterios: categoría, curso, actividad, autor y licencia.

**RF11:** Previsualizar AC: El sistema debe permitirle al usuario previsualizar la actividad que se ha compartido en el repositorio de la forma en que se debe ver dentro de un curso.

**RF12:** Calificar AC: El sistema debe permitirle al usuario calificar las actividades compartidas en el repositorio.

**RF13:** Utilizar AC en el curso: El sistema debe permitirle al usuario utilizar una actividad compartida en el repositorio dentro de su curso.

**RF14:** Editar datos de la AC: El sistema debe permitirle al usuario editar los datos de las actividades compartidas por él en el repositorio.

**RF15:** Eliminar AC del repositorio: El sistema debe permitirle al usuario eliminar las actividades compartidas por él en el repositorio.

**RF16** Notificar AC:

**RF16.1:** Notificar AC modificadas: El sistema debe permitirle al usuario visualizar en la vista del bloque Actividades Compartidas la notificación que la actividad original se modificó.

**RF16.2:** Notificar AC con nuevos comentarios: El sistema debe permitirle al usuario visualizar en la vista del boque Actividades Compartidas la notificación que de una actividad que él compartió, existen nuevos comentarios.

**RF16.3:** Notificar AC que subieron de ranking: El sistema debe permitirle al usuario visualizar la notificación de que la actividad compartida por él ha subido de ranking.

**RF17:** Mostrar AC a repositorios externos: El sistema debe permitirle al usuario mostrar los metadatos de la actividad compartida hacia repositorios externos.

**RF18:** Agregar comentario:

**RF18.1:** Agregar comentario en el foro desde el repositorio: El sistema debe permitirle al usuario agregar un comentario en el repositorio que se publicará en la página del foro que abre la actividad.

**RF18.2:** Agregar nuevo comentario en el foro: El sistema debe permitirle al usuario agregar un nuevo comentario en el foro.

**RF19:** Responder comentario en el foro: El sistema debe permitirle al usuario responder cada comentario realizado a las actividades compartidas por él.

**RF20:** Eliminar comentario en el foro: El sistema debe permitirle al usuario eliminar cada comentario realizado por él en el foro.

**RF21:** Editar comentario en el foro: El sistema debe permitirle al usuario editar cada comentario realizado por él en el foro.

**RF22:** Visualizar comentario en el foro: El sistema debe permitirle al usuario visualizar los comentarios realizados en el foro.

### **2.3.2 Requisitos no funcionales**

Los requisitos no funcionales, como su nombre sugiere, son aquellos que no se refieren directamente a las funciones específicas que proporciona el sistema, sino a las restricciones de los servicios o funciones ofrecidas por el sistema. Normalmente solo se aplican a características o servicios individuales del sistema (51). A continuación, se muestran los requisitos no funcionales(RNF) definidos para la investigación:

✓ **Confiabledad:**

**RNF1:** Solo tendrán acceso al sistema los usuarios registrados con una contraseña y usuario válido.

**RNF2:** El sistema debe ofrecer una advertencia antes de realizar acciones irreversibles (Ejemplo: borrar cualquier dato).

**RFN3:** Seguridad de acceso y administración de usuarios: otorgamiento roles. Los niveles de acceso están determinados por los diferentes roles válidos dentro del sistema.

✓ **Disponibilidad:**

**RFN4:** La aplicación debe estar disponible en todo momento para aquellas personas con acceso a la información y los mecanismos utilizados para lograr la seguridad no debe obstaculizar a los usuarios para obtener los datos deseados en un momento dado.

✓ **Usabilidad:**

**RFN5** Utilizar íconos sugerentes para lograr que el usuario encuentre lo que busca en el menor tiempo posible.

✓ **Restricciones de diseño e implementación:**

**RNF6:** Para el análisis y el diseño del sistema debe ser empleada como herramienta CASE para el modelado Visual Paradigm en su versión 8.0.

**RNF7:** El sistema debe ser implementado haciendo uso del IDE NetBeans en su versión 8.0.

**RNF8:** Se debe utilizar para el desarrollo, como lenguaje de programación del lado del servidor PHP en su versión 5.4.

**RNF9:** Se debe utilizar para el desarrollo del lado del cliente HTML en su versión 5, CSS 3 y como lenguaje de scripting debe ser empleado Java Script haciendo uso de las librerías YUI y JQuery.

**RNF10:** Se debe utilizar como gestor de base de datos PostgreSQL en su versión 8.0 y como servidor web Apache en su versión 2.4.4.

✓ **Interfaz:**

**RNF11:** La interfaz del sistema debe ser de fácil comprensión.

**RNF12:** El sistema debe ser capaz de notificar a los usuarios de la presencia de un error.

En la presente investigación se define el escenario número cuatro, de la metodología de desarrollo de software AUP versión UCI, para modelar el sistema propuesto, debido a que el negocio a informatizar se encuentra definido y el cliente acompaña al equipo de desarrollo para convenir los detalles de los requisitos, probarlos y validarlos. Este escenario emplea las historias de usuario para encapsular los requisitos de software.

## 2.4 Historias de usuario

Las historias de usuario son tarjetas en las cuales el cliente describe brevemente las características que el sistema debe poseer, sean requisitos funcionales o no funcionales. El tratamiento de las historias de usuario es muy dinámico y flexible, en cualquier momento estas pueden romperse o reemplazarse por otras más específicas o generales, añadirse nuevas o ser modificadas.

Cada historia de usuario es lo suficientemente comprensible y delimitada para que los programadores puedan implementarla en unas semanas (para no superar el tamaño de una iteración, que es el tiempo estimado para una entrega de un componente de desarrollo de manera incremental). (52)

A continuación, se muestra en la tabla 2 el modelo que propone AUP para la elaboración de cada historia de usuario (en lo adelante HU). Aquellas de mayor importancia para el cliente, correspondientes al bloque Actividades compartidas, se muestran en las tablas 3- 7. En el anexo 1 se exponen las demás HU que complementan la descripción del comportamiento del bloque.

**Tabla 2** Modelo propuesto para una historia de usuario.

<b>Número:</b> Número de la HU.	<b>Nombre HU:</b> El nombre de la HU que la identifica.
<b>Programador:</b>	<b>Iteración Asignada:</b> Número de iteraciones.
<b>Prioridad:</b> Cuán difícil es para el desarrollador (Alta, Media , Baja).	<b>Tiempo Estimado:</b> Tiempo en días que se le asignará.
<b>Riesgo en Desarrollo:</b> Hacer referencia a los riesgos identificados en plan de riesgos.	<b>Tiempo Real:</b> Tiempo real dedicado a la realización de la HU en semanas.
<b>Descripción:</b> Descripción de la HU, detallando las operaciones del usuario y opcionalmente la respuesta del sistema.	
<b>Observaciones:</b> Información de interés, como glosarios o información de los usuarios.	
<b>Prototipo de interfaz:</b>	

**Tabla 3** Descripción de la HU Compartir actividad.

<b>Número:</b> 8	<b>Nombre HU:</b> Compartir actividad.
<b>Programador:</b> Neobel Rolando Pavón	<b>Iteración Asignada:</b> 1
<b>Prioridad:</b> Alta	<b>Tiempo Estimado:</b> 8 días
<b>Riesgo en Desarrollo:</b>	<b>Tiempo Real:</b> 8 días
<p><b>Descripción:</b> El sistema debe permitirle al usuario compartir una actividad hacia el repositorio del bloque Actividades compartidas. Mostrando por defecto los datos:</p> <ul style="list-style-type: none"> <li>• Categoría</li> <li>• Curso</li> <li>• Autor(es)</li> </ul> <p>Este último muestra por defecto los autores del curso, pero permite ser modificado por el usuario.</p> <p>El usuario debe especificar los siguientes datos obligatorios:</p> <ul style="list-style-type: none"> <li>• Nombre de la actividad</li> <li>• Descripción</li> <li>• Licencia</li> <li>• Palabras claves</li> <li>• Autor(es)</li> </ul> <p>Por defecto la actividad se comparte para el profesor solamente, si este desea que la misma sea vista por todos los profesores y/o por repositorios externos debe marcar los campos siguientes:</p> <ul style="list-style-type: none"> <li>• Entre cursos.</li> <li>• Repositorios externos.</li> </ul>	
<p><b>Observaciones:</b> El usuario debe estar autenticado como un profesor editor del curso.</p>	



## Prototipo de interfaz:

### Formulario de actividad a compartir

<Curso>  
<Categoria>

▼ General

Nombre\* ?

Autor(es)\* ?

Licencia\* ?

Palabras claves\* ?

▼ Descripción\* ?

▼ Compartir

Compartir(para él) ?

Entre cursos ?

Repositorios Externos ?

**Tabla 4** Descripción de la HU Buscar AC (básica).

<b>Número:</b> 9	<b>Nombre HU:</b> Buscar AC (básica).
<b>Programador:</b> Lisandra Remedios Revol	<b>Iteración Asignada:</b> 1
<b>Prioridad:</b> Alta	<b>Tiempo Estimado:</b> 8 días
<b>Riesgo en Desarrollo:</b>	<b>Tiempo Real:</b> 8 días
<p><b>Descripción:</b> El sistema debe permitirle al usuario realizar una búsqueda básica de actividades compartidas en el repositorio, atendiendo a los siguientes criterios:</p> <ul style="list-style-type: none"><li>• Nombre de la actividad.</li><li>• Palabras claves.</li></ul> <p>El resultado de la búsqueda se debe visualizar con las actividades ordenadas por el ranking.</p>	

**Observaciones:** En la búsqueda el campo de inserción de los datos no debe quedarse en blanco. En caso de que se quede en blanco, el sistema debe mostrar un cartel que diga: “Debe entrar un criterio de búsqueda”. El ranking de una actividad se calcula a partir de la cantidad de calificaciones de la opción “Me gusta” más la cantidad de descargas realizadas a la actividad compartida.

**Prototipo de interfaz:** Prototipo de interfaz de la HU número 6, apartado:

**Tabla 5** Descripción de la HU Buscar AC (avanzada) .

<b>Número:</b> 10	<b>Nombre HU:</b> Buscar AC (avanzada).
<b>Programador:</b> Neobel Rolando Pavón	<b>Iteración Asignada:</b> 1
<b>Prioridad:</b> Alta	<b>Tiempo Estimado:</b> 10 días
<b>Riesgo en Desarrollo:</b>	<b>Tiempo Real:</b> 8 días
<p><b>Descripción:</b> El sistema debe permitirle al usuario realizar una búsqueda avanzada de actividades compartidas en el repositorio, filtrando la búsqueda por los siguientes criterios:</p> <ul style="list-style-type: none"> <li>• Categoría</li> <li>• Curso</li> <li>• Actividad</li> <li>• Autor</li> <li>• Licencia</li> </ul> <p>Una vez escogido el o los criterios de búsqueda y su especificación, el sistema debe permitirle al usuario realizar la búsqueda mediante la selección del botón Buscar.</p> <p>El resultado de la búsqueda se debe visualizar con las actividades ordenadas por el ranking.</p>	
<b>Observaciones:</b>	
<b>Prototipo de interfaz:</b> Prototipo de interfaz de la HU número 6, apartado:	

► Búsqueda de Actividades Compartidas

Categoría  Curso  Nombre

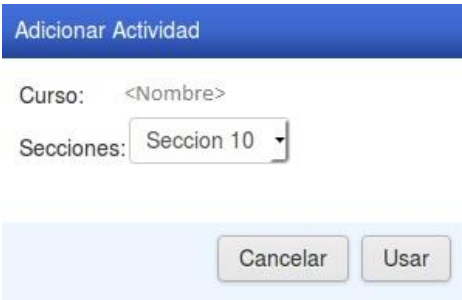
Licencia  Autor

Ordenar por

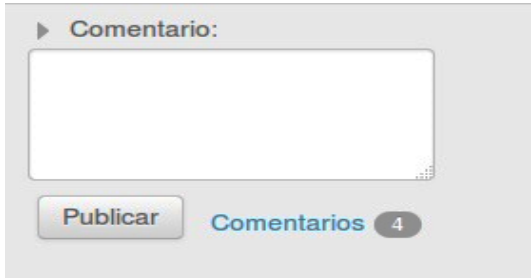
Resultados por Páginas

Buscar

**Tabla 6** Descripción de la HU Utilizar AC en el curso.

<b>Número:</b> 14	<b>Nombre HU:</b> Utilizar AC en el curso.
<b>Programador:</b> Neobel Rolando Pavón	<b>Iteración Asignada:</b> 1
<b>Prioridad:</b> Alta	<b>Tiempo Estimado:</b> 6 días
<b>Riesgo en Desarrollo:</b>	<b>Tiempo Real:</b> 6 días
<p><b>Descripción:</b> El sistema debe permitirle al profesor:</p> <ul style="list-style-type: none"> <li>Utilizar una actividad que ya está compartida en el repositorio dentro de su curso, mediante la selección del ícono: “Utilizar en curso”. Este último debe conducir a un menú que permita escoger en que sección del curso ubicar la actividad compartida.</li> <li>Usar la actividad compartida en la sección seleccionada o cancelar la acción.</li> </ul>	
<b>Observaciones:</b>	
<p><b>Prototipo de interfaz:</b> Prototipo de interfaz de la HU número 6, apartado:</p>	
<p>Prototipo de interfaz: Menú Adicionar Actividad.</p> 	

**Tabla 7** Descripción de la HU Agregar comentario en el foro desde el repositorio.

<b>Número:</b> 21	<b>Nombre HU:</b> Agregar comentario en el foro desde el repositorio.
<b>Programador:</b> Neobel Rolando Pavón	<b>Iteración Asignada:</b> 1
<b>Prioridad:</b> Media	<b>Tiempo Estimado:</b> 4 días
<b>Riesgo en Desarrollo:</b>	<b>Tiempo Real:</b> 6 días
<p><b>Descripción:</b> El sistema debe permitirle al usuario:</p> <ul style="list-style-type: none"> <li>• Escribir un comentario en la página del repositorio.</li> <li>• Publicar el comentario insertado en el repositorio, en la página del foro que abre la actividad, mediante la selección del botón Publicar.</li> <li>• Visualizar la cantidad de comentarios realizados a la actividad compartida.</li> </ul>	
<b>Observaciones:</b>	
<b>Prototipo de interfaz:</b>	
	

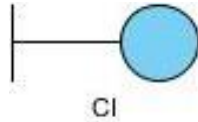
## 2.5 Modelo de análisis

El Modelo de Análisis ofrece una especificación precisa de los requisitos funcionales y de las historias de usuario. Permite obtener una visión más detallada del sistema y, los desarrolladores utilizan esta información para comprender cómo debe diseñarse e implementarse el software.

### 2.5.1 Diagrama de clases del análisis (DCA)

Una clase de análisis representa una abstracción de una o varias clases, ajustando las mismas a uno de los tres estereotipos existentes sobre las clases utilizados por el modelo de dominio: de interfaz, de control o de entidad.

La **clase interfaz** se utiliza generalmente para modelar la interacción entre el sistema y los actores.



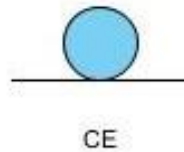
**Figura 2** Estereotipo de la clase interfaz.

La **clase de control** es empleada habitualmente para representar coordinación, secuenciación, transacciones y control de objetos.



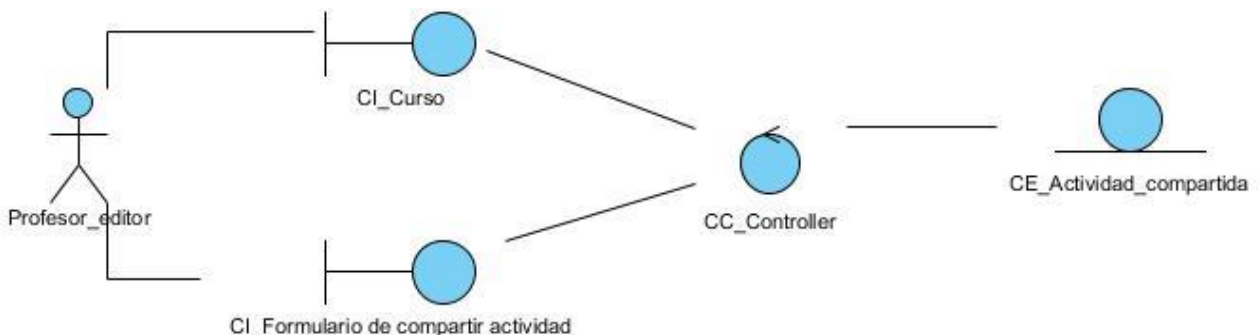
**Figura 3** Estereotipo de la clase control.

La **clase entidad** es usada para modelar la información que tiene una vida larga y a veces es persistente, asimismo muestran una estructura de datos lógica y contribuyen a comprender de qué información depende el sistema.

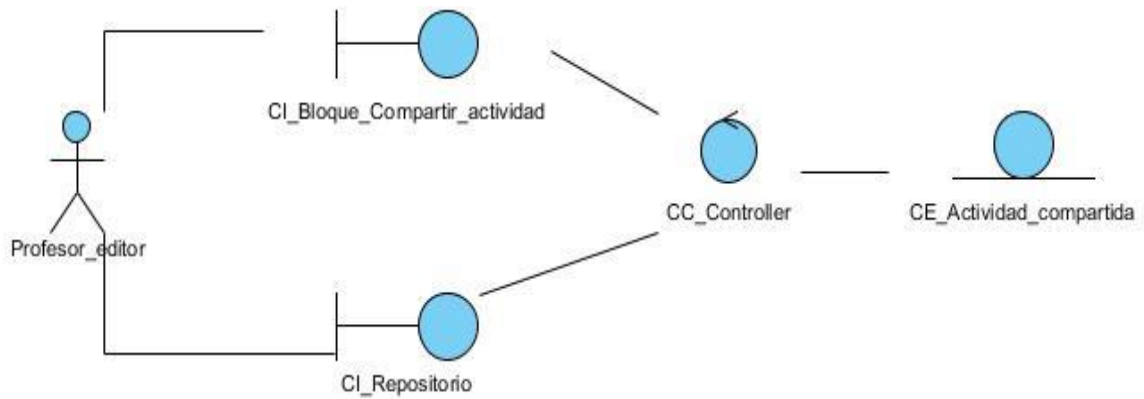


**Figura 4** Estereotipo de la clase entidad.

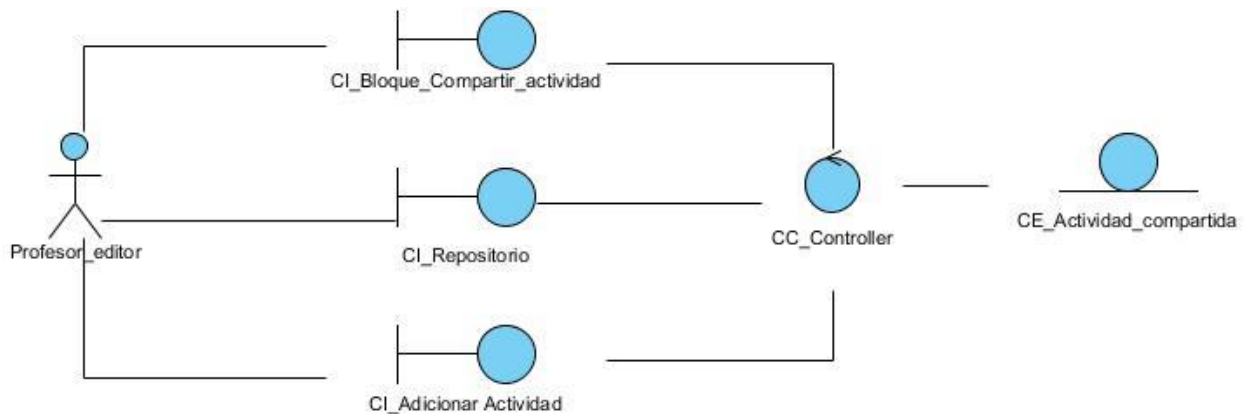
A continuación, se muestran los diagramas de clases del análisis correspondiente a las historias de usuario Compartir actividad, Buscar AC (avanzada) y Utilizar AC en el curso. Los diagramas de clases del análisis correspondiente a las restantes historias de usuario se muestran en el Anexo 2.



**Figura 5** Diagrama de clases del análisis de la HU Compartir actividad.



**Figura 6** Diagrama de clases del análisis de la HU Buscar AC (avanzada).

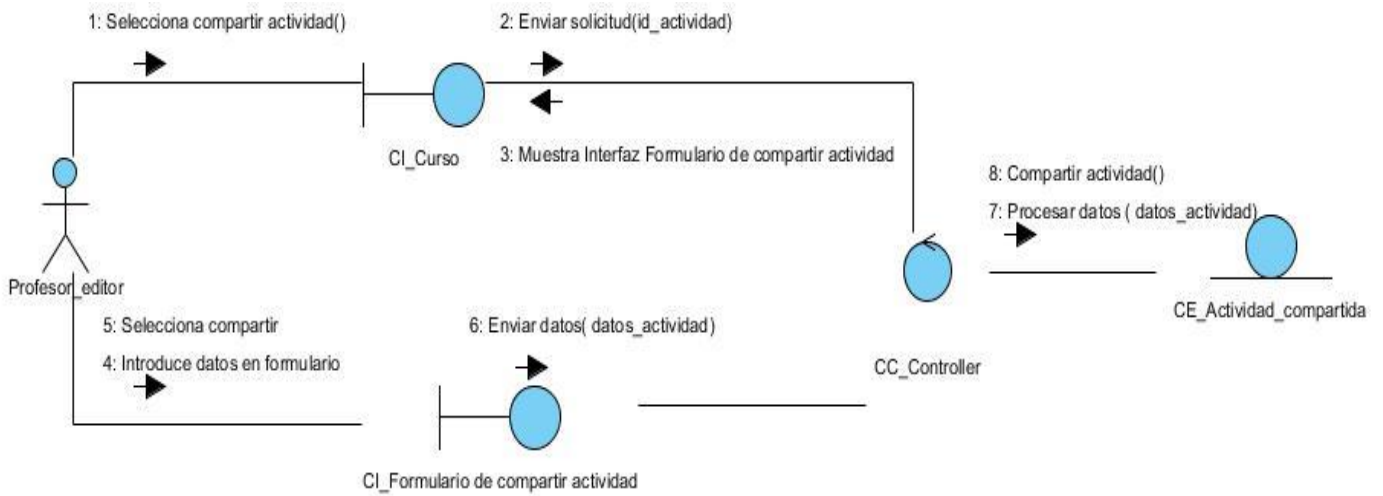


**Figura 7** Diagrama de clases del análisis Utilizar AC en el curso.

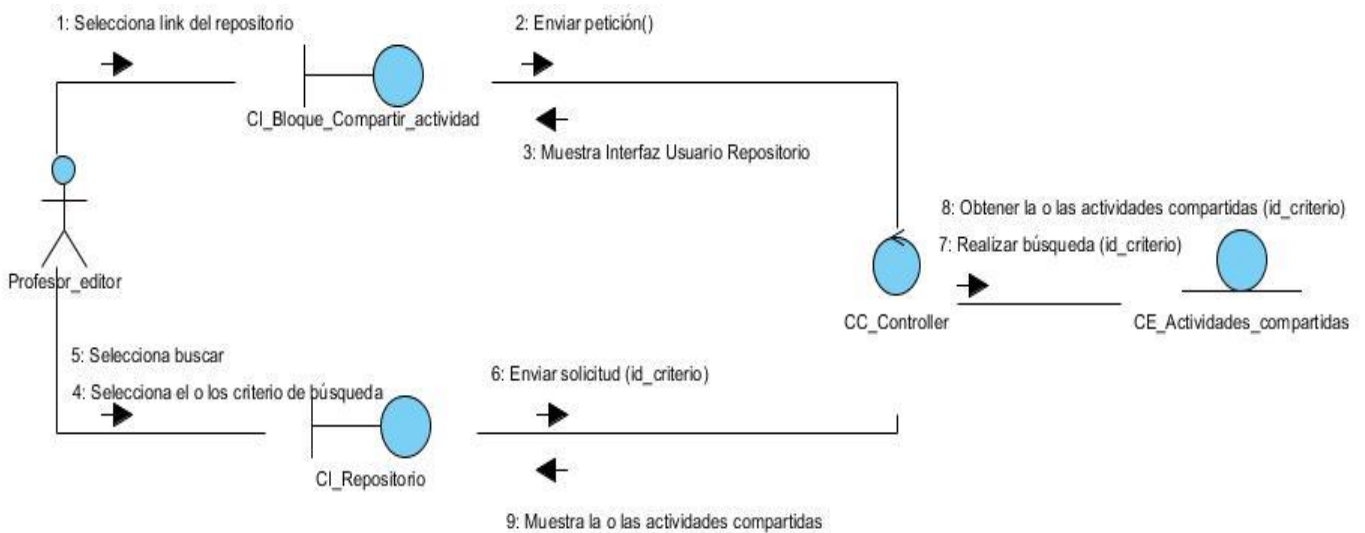
### 2.5.2 Diagrama de colaboración del análisis (DCO)

Los diagramas de colaboración del análisis son utilizados fundamentalmente para modelar las interacciones entre los objetos en el análisis. Éstos recuerdan los diagramas de clases, pero contienen instancias y enlaces en lugar de clases y asociaciones, mostrando cómo interactúan los objetos secuencialmente o en paralelo enumerando los mensajes que se envían unos a otros. (53)

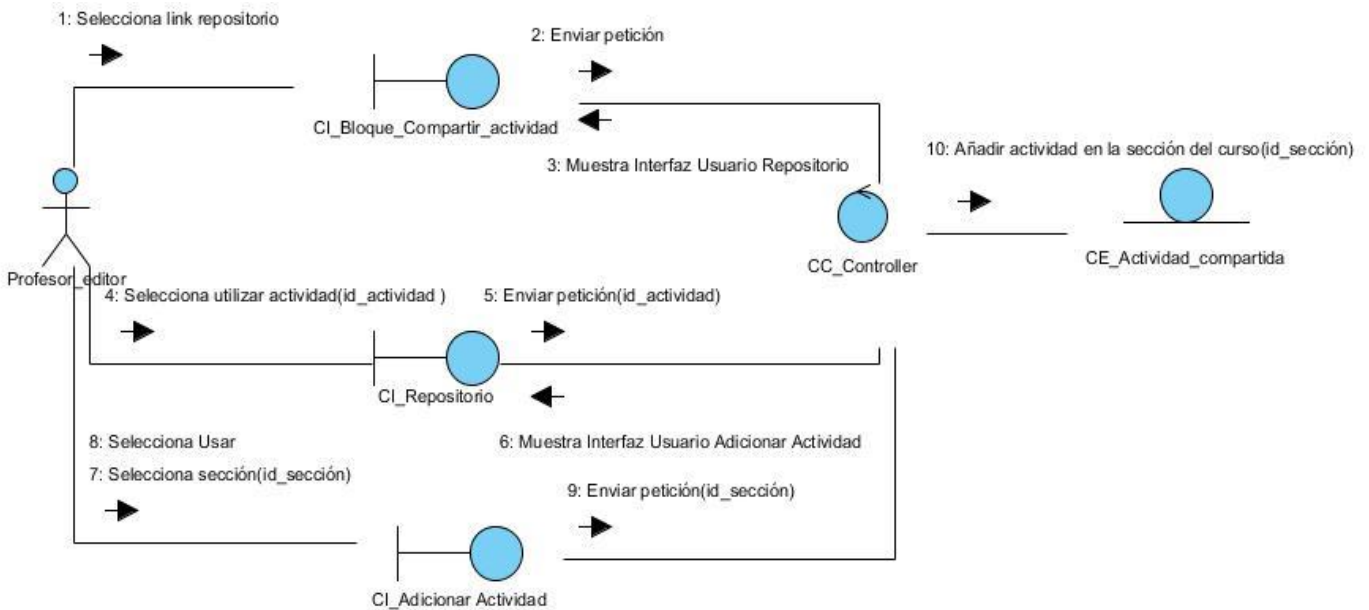
A continuación, se muestran los diagramas de colaboración correspondiente a las historias de usuario Compartir actividad, Buscar AC(avanzada) y Utilizar AC en el curso. Los diagramas de colaboración correspondiente a las restantes historias de usuario se muestran en el Anexo 3.



**Figura 8** Diagrama de Colaboración de la HU Compartir actividad.



**Figura 9** Diagrama de Colaboración de la HU Buscar AC (avanzada).



**Figura 10** Diagrama de Colaboración de la HU Utilizar AC en el curso.

## 2.6 Patrón Arquitectónico

Los patrones arquitectónicos representan un esquema de organización estructural esencial para un sistema de software, que consta de subsistemas, sus responsabilidades e interrelaciones.

Moodle no se desarrolla atendiendo a un patrón arquitectónico definido, debido a que mezcla un poco de programación orientada a objetos con programación estructurada. No utiliza ningún sistema de encapsulación de datos, ni separa la vista de la lógica de la aplicación, por lo que no emplea el patrón arquitectónico Modelo Vista Controlador.

Moodle está estructurado por una arquitectura modular regida por un núcleo (conocido como el Moodle core) del sistema, rodeado de una gran cantidad de plugins que proveen de funcionalidad específica a la plataforma. Los plugins en Moodle pueden ser de varios tipos: autenticación, bloques, matriculación, temas, actividades, etc.

Todos los módulos comparten una funcionalidad específica entre sí y el core. La distribución estándar de Moodle incluye el core del sistema y un número de módulos de todos los tipos que proveen de la funcionalidad base del sistema y con el cual se puede empezar a funcionar sin carencias ni problemas de ningún tipo.

El core de Moodle proporciona todos los mecanismos necesarios para implementar la plataforma educativa. Implementa todos aquellos mecanismos que luego los plugins utilizan para desarrollar su función. Es por lo tanto el core la parte más importante del sistema Moodle. (54)



## **2.7 Modelo de diseño**

Es un modelo de objetos que describe la realización física de los casos de uso centrándose, en cómo los requisitos funcionales y no funcionales, junto con otras restricciones relacionadas con el entorno de implementación, tienen impacto en el sistema a considerar. Por lo tanto, mientras el modelo de análisis sirve como una aproximación del modelo de diseño, el modelo de diseño funciona como esquema para la implementación.

### **2.7.1 Patrones de diseño**

Un patrón de diseño nombra, abstrae e identifica los aspectos clave de un diseño estructurado común, que lo hace útil para la creación de diseños orientados a objetos reutilizables. Definen una posible solución correcta para un problema de diseño dentro de un contexto dado, describiendo las cualidades invariantes de todas las soluciones (55). A continuación, se describen los patrones de diseño utilizados durante el desarrollo del sistema:

#### **Patrones GRASP**

Los patrones GRASP describen los principios fundamentales de la asignación de responsabilidades a objetos, expresados en formas de patrones. Se conocen como una serie de "buenas prácticas" de aplicación, recomendable en el diseño de software. Tienen gran importancia, debido a que dan solución a muchos de los problemas que se pueden presentar a la hora de programar. (56)

Dentro de los patrones GRASP utilizados en el desarrollo del sistema se encuentran los siguientes:

#### **Creador**

Este patrón plantea que se debe asignar a una clase X la responsabilidad de crear una instancia de una clase Y. La clase `block_sharing_activity` será la responsable de crear una nueva instancia de la `controller`.

#### **Experto**

Es empleado para lograr que cada clase cumpla con la responsabilidad que le corresponde asignando una responsabilidad a la clase que tiene la información necesaria para cumplirla.



Figura 11 Diagrama de clase que representa al patrón experto.

## Controlador

Sirve como intermediario entre una determinada interfaz y el algoritmo que la implementa, de tal forma que es la que recibe los datos del usuario y la que los envía a las distintas clases según el método llamado.



Figura 12 Diagrama de clase que representa al patrón controlador.

## Bajo Acoplamiento

Este patrón es una medida de la fuerza con que una clase está conectada a otras clases. Acoplamiento bajo significa que una clase no depende de muchas clases.



**Figura 13** Diagrama de clase que representa el patrón bajo acoplamiento.

### 2.7.2 Diagrama de clases de diseño.

Un diagrama de clases de diseño sirve para visualizar las relaciones entre las clases que involucran el sistema, las cuales pueden ser asociativas, de herencia, de uso y de agregación, ya que una clase es una descripción de conjunto de objetos que comparten los mismos atributos, operaciones, métodos, relaciones y semántica; mostrando un conjunto de elementos que son estáticos, como las clases y tipos junto con sus contenidos y relaciones. (57)

El presente diagrama sirve de apoyo en la comprensión de la estructura del sistema. A continuación, se muestra el diagrama de clases del diseño a desarrollar.

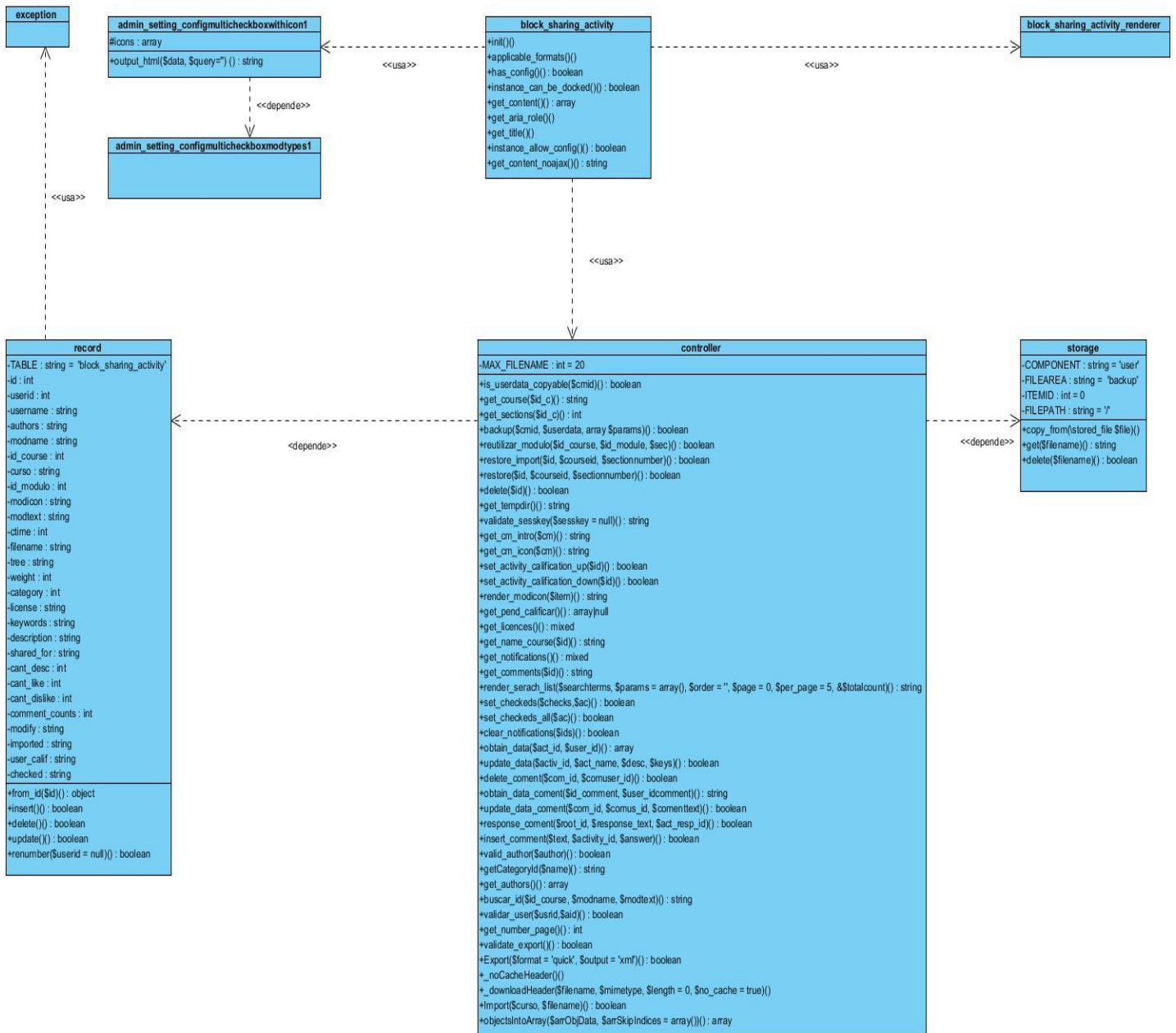


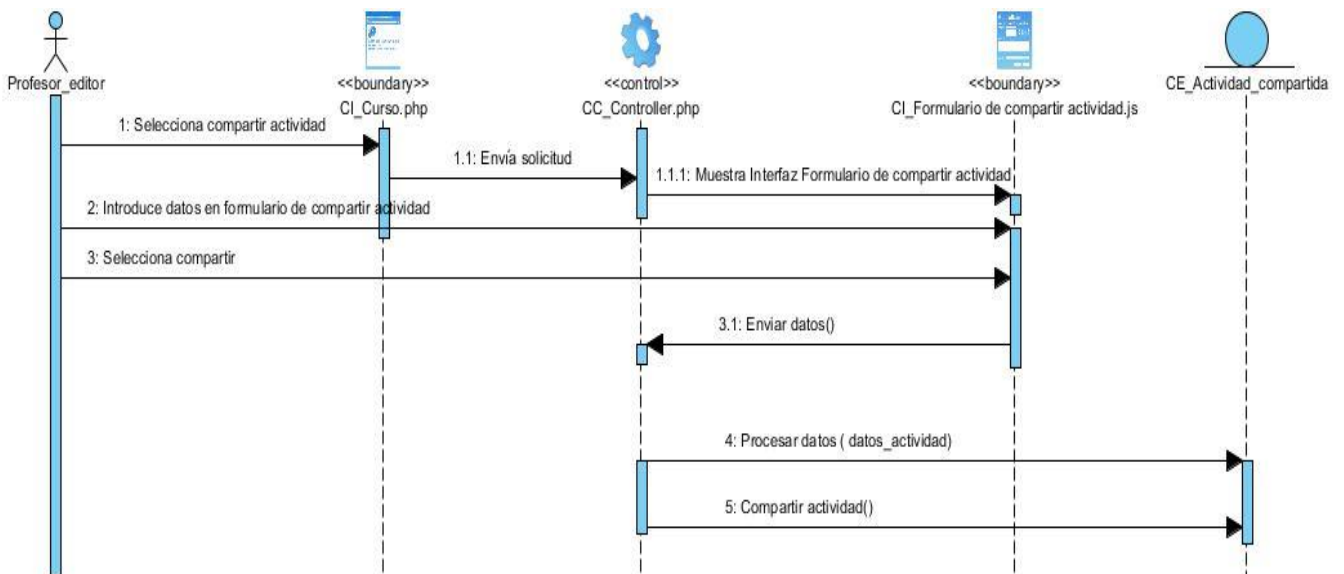
Figura 14 Diagrama de clases de diseño.

### 2.7.3 Diagrama de secuencia del diseño (DS)

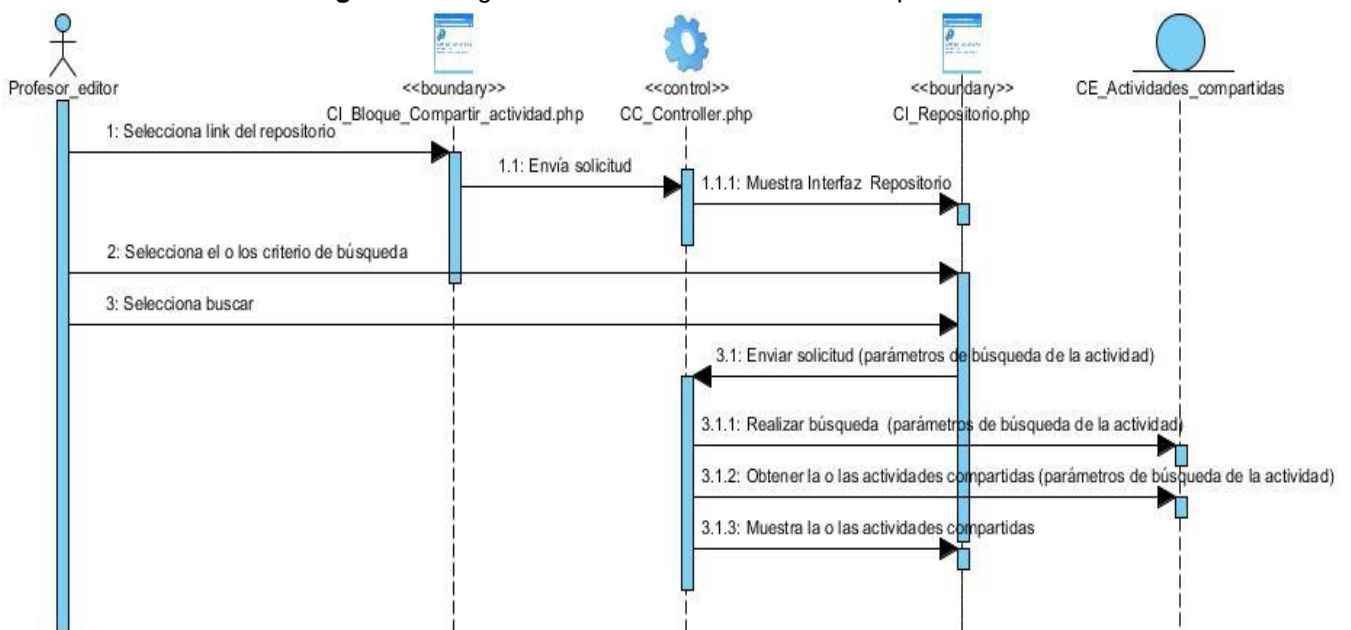
Un diagrama de secuencia muestra un conjunto de mensajes, dispuestos en una secuencia temporal, donde cada rol en la secuencia se muestra como una línea de vida que representa el rol durante cierto plazo del tiempo, con la interacción completa. Los mensajes se muestran como flechas entre las líneas de vida. Un diagrama de secuencia puede mostrar un escenario, es decir una historia individual de una transacción. (44)

Este artefacto permitirá representar la interacción de los objetos que intervienen en una funcionalidad, mediante la transferencia de mensajes o métodos; le indicarán además al equipo de desarrollo en qué momento un objeto determinado toma el control del sistema mediante su línea de vida.

A continuación, se presentan los diagramas de secuencia correspondiente a las historias de usuario Compartir actividad, Buscar AC(avanzada) y Utilizar AC en el curso. El resto de los diagramas de secuencia encuentran en el Anexo 4.



**Figura 15** Diagrama de Secuencia de la HU Compartir actividad.



**Figura 16** Diagrama de Secuencia de la HU Buscar AC (avanzada).

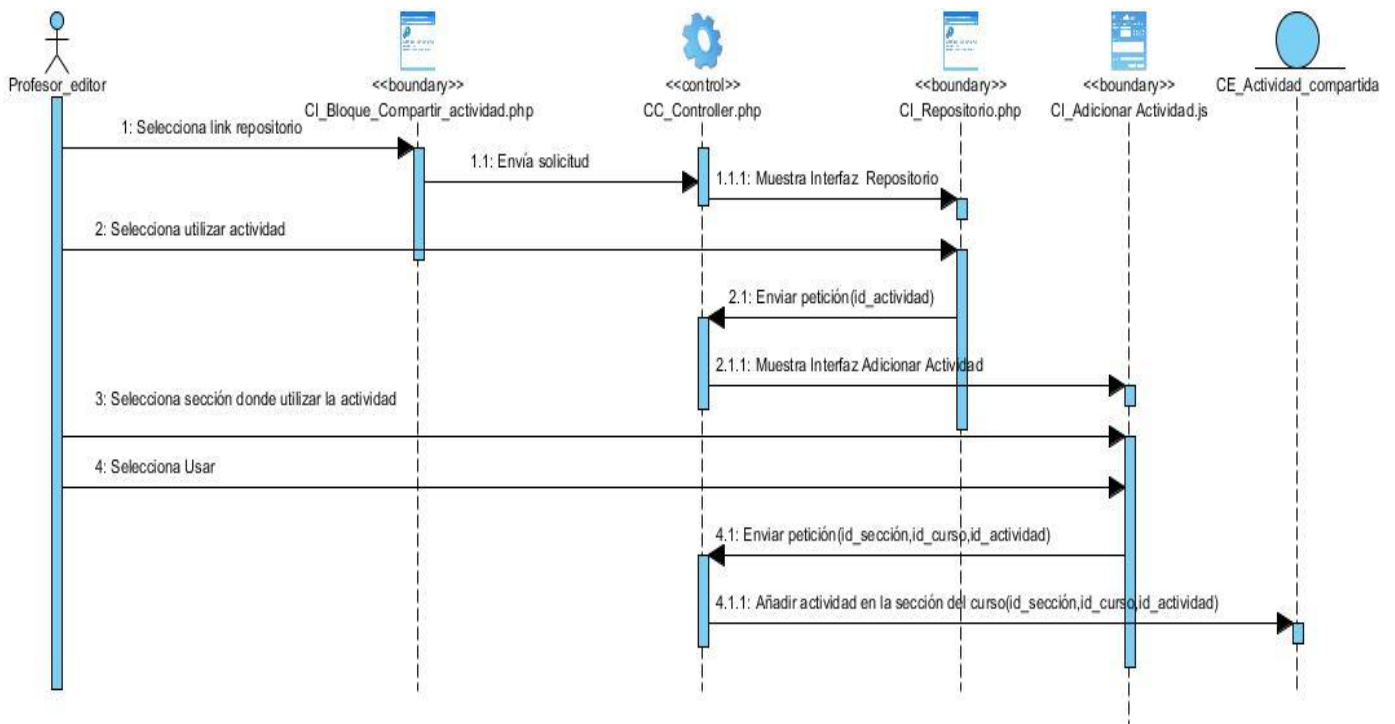


Figura 17 Diagrama de Secuencia de la HU Utilizar AC en el curso.

## 2.8 Diseño de la base de datos

El diagrama entidad-relación es uno de los modelos más usados para diseñar bases de datos, este modelo se encuentra basado en dos conceptos fundamentales: entidades, que representan objetos sobre los cuales se desea guardar información y las relaciones, que constituyen las relaciones entre las entidades. (58)

A continuación, se presenta el modelo de datos que contiene las entidades del bloque Actividades compartidas, que formarán parte de las tablas de la base de datos de Moodle y las relaciones entre ellas. Éstas serán utilizadas por las funcionalidades a desarrollar. El bloque emplea nueve entidades para su funcionamiento, las creadas por este son *block\_sharing\_activity*, *forum\_sharing\_activity* y *block\_sharing\_activity\_reused*, las demás representan entidades de la base de datos de Moodle. (ver Figura 18).

La entidad *block\_sharing\_activity* almacena toda la información referente a las actividades que se comparten para el repositorio. Por otra parte, *forum\_sharing\_activity* contiene los datos de los comentarios que se manejan en el sistema y *block\_sharing\_activity\_reused* almacena la relación de datos de las actividades que ya han sido reutilizadas por un determinado usuario. *User*, a su vez, tiene la información asociada a cada usuario que interactúa con el bloque,

mientras que *course* contiene los datos de los cursos en los que se reutiliza o comparte una actividad.

La entidad *modules* almacena los atributos de los módulos de tipo actividad de la plataforma y la tabla *license* es la que emplea el bloque para obtener los datos de la licencia bajo la cual se comparte la actividad. La entidad *role\_assignments* es la encargada de guardar la relación entre el usuario y el rol que este presenta. La entidad *role* almacena los roles que por defecto presenta la base de datos de Moodle.

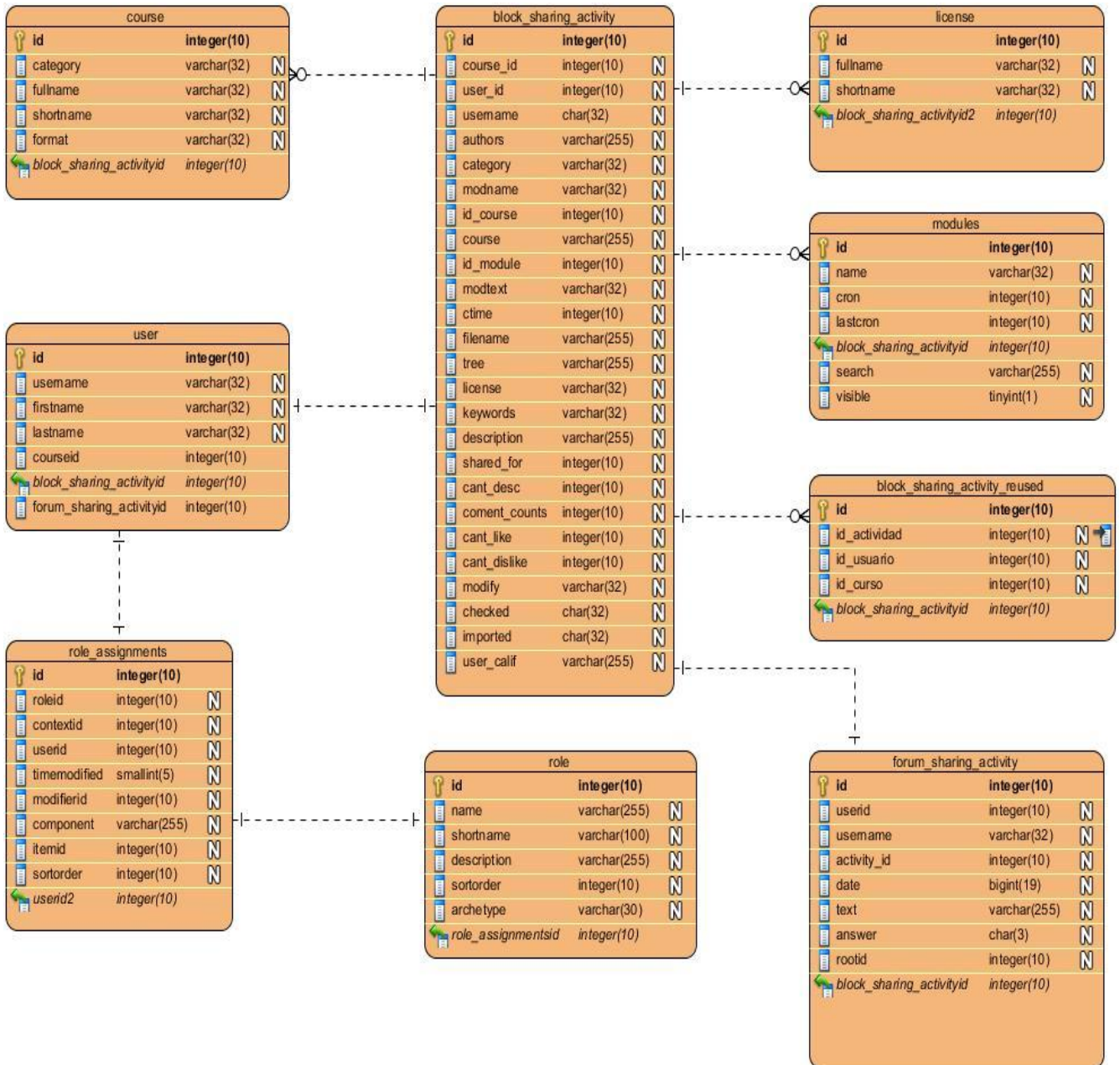


Figura 18 Diagrama Entidad-Relación.

## **Conclusiones parciales**

- El modelo conceptual elaborado a partir de los principales conceptos involucrados en el entorno donde se usará el software, constituyó la antesala a la identificación de los requisitos funcionales y no funcionales.
- Las definiciones de historias de usuario permitieron tener la base para la implementación de la aplicación.
- Con el desarrollo del flujo de análisis y diseño correspondiente a la metodología seleccionada, se logró identificar los elementos que componen el diseño del bloque a desarrollar.
- Se aplicaron los patrones necesarios para lograr un diseño flexible y eficiente.



## CAPÍTULO 3: IMPLEMENTACIÓN Y VALIDACIÓN DEL BLOQUE

---

En el presente capítulo se describen los elementos necesarios para llevar a cabo la implementación del bloque. Se muestra la distribución del sistema en nodos mediante el diagrama de despliegue. Se describe la organización de los componentes y las relaciones lógicas entre ellos a través del diagrama de componentes, con el que se conforma el modelo de implementación de la solución. Además, se abordan los tipos de pruebas realizadas a la aplicación con el fin de evidenciar su correcto funcionamiento.

### 3.1 Modelo de implementación

El flujo de trabajo de implementación describe cómo los elementos del modelo del diseño se implementan en términos de componentes y cómo estos se organizan de acuerdo a los nodos específicos en el modelo de despliegue.

Los diagramas de despliegue y componentes, que son artefactos generados en este flujo de trabajo, conforman lo que se conoce como un modelo de implementación al describir los componentes a construir, su organización y dependencia entre nodos físicos en los que funcionará la aplicación.

### 3.2 Diagrama de despliegue

El diagrama de despliegue que aparece a continuación muestra la disposición física de los distintos nodos que componen el sistema y el reparto de los componentes sobre dichos nodos, los cuales representan los objetos físicos con recursos computacionales. Las conexiones establecidas son asociaciones de comunicación entre los nodos, y se etiquetan con un estereotipo que identifica el protocolo de comunicación o la red utilizada.

El bloque se integrará a la plataforma Moodle en la versión 2.9.x, por lo que el diagrama estará compuesto por los nodos necesarios para acceder a dicha plataforma.



**Figura 19** Diagrama de despliegue.

Los usuarios podrán acceder a la plataforma a través de una computadora cliente que se conecte mediante el protocolo HTTPS (del inglés *Hypertext Transfer Protocol Secure*) a un servidor web Apache donde se montará Moodle. Este debe estar conectado a un servidor de base de datos PostgreSQL o MySQL (de acuerdo a la preferencia del administrador del sistema) mediante el protocolo ADO (del inglés *ActiveX Data Objects*).

### 3.3 Diagrama de componentes

Los diagramas de componentes muestran las dependencias lógicas entre componentes de software. Estos diagramas son utilizados para describir la vista de implementación estática de un sistema determinado. Los mismos poseen un nivel más alto de abstracción en correspondencia con el diagrama de clases y representan una o varias clases, interacciones o colaboraciones.

Los componentes físicos incluyen archivos, bibliotecas, ejecutables o paquetes. Son muy utilizados para modelar y documentar la arquitectura al permitir visualizar la organización y las dependencias entre un conjunto de componentes. Representan partes modulares del sistema que pueden desplegarse y reemplazarse. También encapsulan implementación y un conjunto de interfaces

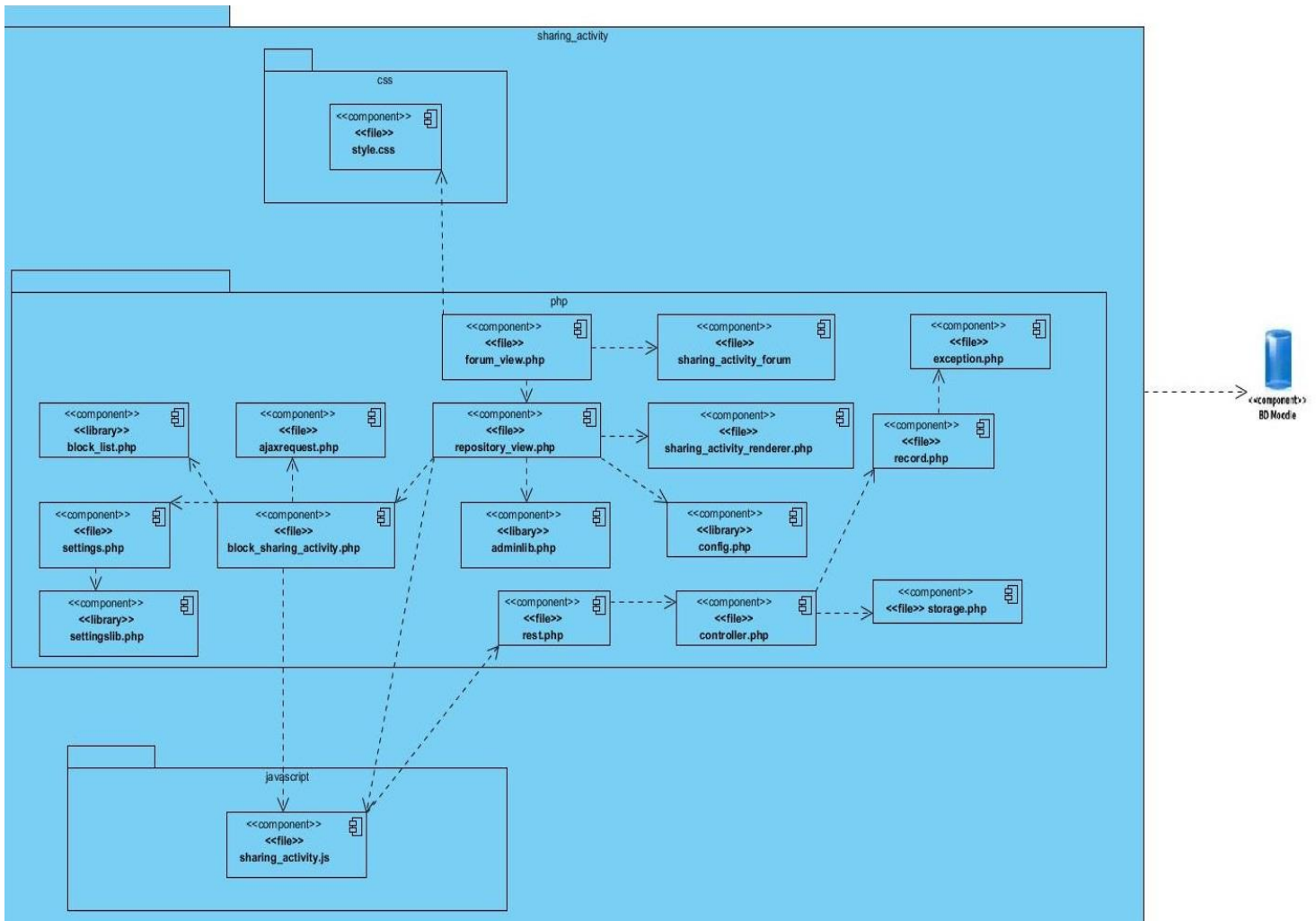


Figura 20 Diagrama de componentes.

proporcionado la realización de los mismos. La Figura 20 muestra el diagrama de componentes de la aplicación. (59)

### 3.4 Pruebas de software

Las pruebas de software son las investigaciones empíricas y técnicas cuyo fin es proporcionar información objetiva e independiente sobre posibles fallos de implementación, la calidad o usabilidad de un producto; probando el comportamiento del mismo. (60)

Con la ejecución de las pruebas de software se persigue descubrir defectos en el sistema asociados a comportamientos incorrectos o no deseables y para verificar que cumple con los requerimientos del cliente, con el fin de suplir sus necesidades. Al ejecutarse esta actividad no se obtiene un sistema totalmente libre de errores, pero si apto para ser usado por el usuario final.

#### 3.4.1 Niveles de prueba

El proceso de pruebas se realiza en varios niveles, estos se enfocan a determinados objetivos y están estrechamente relacionados con los tipos de pruebas. Con el objetivo de llevar a cabo la realización de pruebas al bloque desarrollado para Moodle en la versión 2.9.x, se aplica a nivel de pruebas las pruebas de (61):

**Sistema:** Comprueban los requisitos no funcionales.

**Aceptación:** Comprueban el comportamiento del sistema frente a los requisitos del cliente (suele participar el mismo cliente o los usuarios).

Para ejecutar la prueba de sistema se determinó aplicar el siguiente tipo de prueba:

- **Seguridad:** Intenta identificar fallas en los mecanismos de protección establecidos en el sistema y las acciones no autorizadas para los distintos roles de usuarios.

Para ejecutar la prueba de aceptación se determinó aplicar el siguiente tipo de prueba:

- **Función:** Valida las funciones, métodos, servicios e historias de usuario. Para la realización de esta prueba se utiliza el método o prueba de caja negra.

#### 3.4.2 Métodos de prueba

Existen distintas técnicas de pruebas que proporcionan criterios para generar casos de pruebas que provoquen fallos en los programas. Se agrupan en: técnicas de caja blanca o estructural y técnicas de caja negra o funcional. La primera se basa en un minucioso examen de los detalles procedimentales a evaluar, por lo que es necesario conocer la lógica del programa. La segunda se

basa en la realización de pruebas sobre las entradas y salidas de dicho programa, no es necesario conocer la lógica del programa, únicamente la funcionalidad que debe realizar.

Para la validación de la propuesta de solución se aplicará el método de caja negra, debido a que este permitirá corregir problemas de rendimiento, en la interfaz del usuario y se comprobará que la propuesta de solución realiza las funciones requeridas por el usuario.

### **3.4.3 Pruebas de caja negra**

Las pruebas de caja negra, también denominadas de comportamiento, se centran en los requisitos funcionales del software. Estas pruebas le permiten al ingeniero del software obtener un conjunto de condiciones de entrada que ejerciten completamente todos los requisitos funcionales de un programa.

Para desarrollar la prueba de caja negra existen varias técnicas, entre ellas están:

- **Técnica de la Partición de Equivalencia:** esta técnica divide el campo de entrada en clases de datos que tienden a ejercitar determinadas funciones del software.
- **Técnica del Análisis de Valores Límites:** esta técnica prueba la habilidad del programa para manejar datos que se encuentran en los límites aceptables.
- **Técnica de Grafos de Causa-Efecto:** es una técnica que permite al encargado de la prueba validar complejos conjuntos de acciones y condiciones.

Se hace uso de la técnica de la Partición de Equivalencia debido a que permite examinar los valores válidos e inválidos de las entradas existentes en el software. Divide el dominio de entrada de un programa en clases de datos a partir de las cuales se derivan las descripciones de casos de prueba. La partición equivalente se dirige a la definición de casos de prueba que descubran clases de errores, reduciendo así el número de casos de prueba a desarrollar. (62)

### **3.4.4 Diseños de casos de prueba**

En el presente epígrafe se muestran las pruebas realizadas para demostrar el funcionamiento de la aplicación y los resultados obtenidos en ellas. La intención de los casos de prueba es probar el sistema de una forma detallada, incluyendo las entradas con las que se experimentarán, las condiciones bajo las cuales se realizan y los resultados esperados.

A continuación, se presentan los casos de prueba propuestos para las HU Compartir actividad, Buscar AC (avanzada) y Utilizar AC en curso. Para consultar el resto de los casos de prueba remitirse al Anexo 5.

**Tabla 8** Caso de Prueba Compartir actividad.

<b>CP Compartir actividad</b>									
<b>Descripción general</b>									
Permitir compartir una actividad hacia el repositorio del bloque Actividades compartidas.									
<b>Condiciones previas</b>									
El usuario debe estar autenticado como un profesor editor del curso.									
<b>Escenario</b>	<b>Descripción</b>	<b>Nombre</b>	<b>Licencia</b>	<b>Autor(es)</b>	<b>Palabras claves</b>	<b>Descripción</b>	<b>Compartir</b>	<b>Respuesta del sistema</b>	<b>Flujo central</b>
EC 1.1 Opción de compartir una actividad.	Selecciona la opción de compartir una actividad.							Muestra los datos del tipo de actividad a compartir permitiendo introducir o seleccionar de manera obligatoria los valores: -Nombre de la actividad-Descripción - Licencia, -Palabras claves- Autor(es)-compartir. Permite además:-Compartir-Cancelar la operación en cualquier momento.	Página del Curso/Editar actividad/ /Ícono Compartir actividad/ Formulario de compartir actividad.

EC 1.2 Opción de Compartir.	Introduce y/o selecciona los datos de la actividad a compartir y selecciona la opción Compartir.	V	V	V	V	V	V	Valida los datos. Crea una actividad compartida. Muestra la actividad recién compartida en el listado de actividades compartidas del repositorio.	Página del Curso/Editar actividad/Ícono Compartir actividad/Formulario de compartir actividad / Compartir.
EC 1.3 Opción de Cancelar.	Selecciona la opción de cancelar.							Elimina los datos creados. Regresa a la página del curso.	Página del Curso/Editar actividad/Ícono Compartir actividad/Formulario de compartir actividad /Cancelar.
EC 1.4 Datos incompletos.	Existen datos incompletos.	I	V	V	V	V	V	Muestra un mensaje de información. Muestra un indicador sobre los campos vacíos.	Página del Curso/Editar actividad/Ícono Compartir actividad/Formulario de compartir actividad /
		V	I	V	V	V	V		
		V	V	I	V	V	V		
		V	V	V	I	V	V		
		V	V	V	V	I	V		
		V	V	V	V	V	I		
		V	V	V	V	V	V	Regresa al EC 1.1.	

									Compartir.
EC 1.5 Datos incorrectos.	Existen datos incorrectos.	I	V	V	V	V	V	Muestra un mensaje de información.	Página del Curso/Editar actividad/Ícono
		V	I	V	V	V	V		
		V	V	I	V	V	V		
		V	V	V	I	V	V	Muestra un indicador sobre los campos incorrectos.	Compartir actividad/ Formulario/ Compartir.
		V	V	V	V	I	V		
		V	V	V	V	V	I		
		V	V	V	V	V	V		
							Regresa al EC 1.1.		

**Tabla 9** Caso de Prueba Buscar AC (avanzada).

<b>CP Buscar AC (avanzada).</b>								
<b>Descripción general</b>								
Permitir realizar una búsqueda avanzada de las actividades compartidas.								
<b>Condiciones previas</b>								
Para realizar la búsqueda se debe escoger como mínimo una especificación de un criterio de búsqueda.								
Escenario	Descripción	Categoría	Curso	Actividad	Autor	Licencia	Respuesta del sistema	Flujo central
EC 1.1 Selección del criterio	Selecciona el criterio de búsqueda avanzada.						Brinda la posibilidad de escoger el o los criterios por el que se desea filtrar la búsqueda mediante las opciones: -	Página del repositorio/Sección Búsqueda de actividades compartidas.

búsqueda avanzada.							Categoría-Curso-Actividad-Autor-Licencia. Por cada criterio muestra un desplegable para seleccionar una especificación de búsqueda. Permite además realizar la búsqueda.	
EC 1.2 Especificar por criterio de búsqueda.	Selecciona el o los criterio de búsqueda avanzada y de este escoge una especificación de búsqueda.	V	N/A	N/A	N/A	N/A	Permite escoger el o los criterios de búsqueda avanzada y de ellos una de sus especificaciones de búsqueda.	Página del repositorio/Sección Búsqueda de actividades compartidas.
		V	V	N/A	N/A	N/A		
		V	V	V	N/A	N/A		
		V	V	V	V	N/A		
		V	V	V	V	V		
EC 1.3 Opción Buscar.	Selecciona la opción Buscar.	N/A	N/A	N/A	N/A	N/A	Filtra los datos.  Muestra el listado de actividades compartidas ordenadas por ranking y acorde con la especificación del criterio de búsqueda seleccionado.	Página del repositorio/Sección Búsqueda de actividades compartidas/Buscar.



**Tabla 10** Caso de Prueba Utilizar AC en curso.

<b>CP Utilizar AC en curso</b>				
<b>Objetivo</b>				
Permitir utilizar una actividad que ya está compartida en el repositorio dentro de un curso.				
<b>Condiciones previas</b>				
<b>Escenario</b>	<b>Descripción</b>	<b>Secciones</b>	<b>Respuesta del sistema</b>	<b>Flujo central</b>
EC 1.1 Opción Utilizar en curso.	Selecciona el ícono utilizar en curso.		Permite seleccionar el ícono utilizar en curso. Muestra un menú que permite seleccionar en que sección del curso ubicar la actividad compartida. Permite, además:  -Usar.  -Cancelar.	Página del repositorio  /Sección Lista de actividades compartidas/Ícono Utilizar en curso.
EC 1.2 Seleccionar sección.	Seleccionar la sección del curso donde se desee ubicar la actividad compartida.	V	Permite la sección del curso.	Página del repositorio /Sección Lista de actividades compartidas/ Ícono Utilizar en curso/Secciones.
EC 1.3 Opción Usar	Seleccionar la sección del curso donde se desee ubicar la actividad compartida y selecciono el botón Usar.	V	Valida los datos.  Añade la actividad compartida a la sección del curso seleccionada.	Página del repositorio /Sección Lista de actividades compartidas/ Ícono Utilizar en curso/ Secciones/Usar.

EC 1.4 Opción Cancelar.	Selecciona la opción cancelar.		Elimina los datos creados. Muestra la lista de actividades compartidas consultada anteriormente.	Página del repositorio /Sección Lista de actividades compartidas/ Ícono Utilizar en curso/ Secciones/Cancelar.
-------------------------	--------------------------------	--	--	--

### 3.4.5 Resultados de las pruebas

Las pruebas ejecutadas al bloque para la plataforma Moodle en la versión 2.9.x arrojaron los siguientes resultados:

#### Pruebas de seguridad

Para realizar las pruebas de seguridad se tuvieron en cuenta los roles definidos para interactuar con el bloque; Profesor editor y Administrador. A cada usuario se le asignó un rol de los antes mencionados, con el objetivo de comprobar si estos podían acceder, sin los privilegios requeridos, a las funcionalidades del bloque.

El resultado obtenido luego de aplicar las pruebas de seguridad es que el usuario con permiso de Administrador puede ejecutar todos los procesos de compartir actividad, buscar, editar, eliminar, calificar y utilizar actividad compartida, visualizar notificaciones, ordenar la búsqueda de actividades, realizar los procesos asociadas al foro, y exportar e importar el repositorio. Sin embargo, el Profesor editor puede llevar a cabo todas las acciones que realiza el Administrador, exceptuando configurar, instalar y desinstalar el bloque Actividades compartidas.

#### Prueba de función

Con el objetivo de medir la funcionalidad operativa del software se hace uso de las pruebas de caja negra, teniendo en cuenta la técnica de partición por equivalencia. Además, se utilizan los casos de prueba generados durante este flujo de trabajo con el fin de detectar la mayor cantidad de no conformidades posibles en las funcionalidades del bloque realizándose tres iteraciones de prueba.

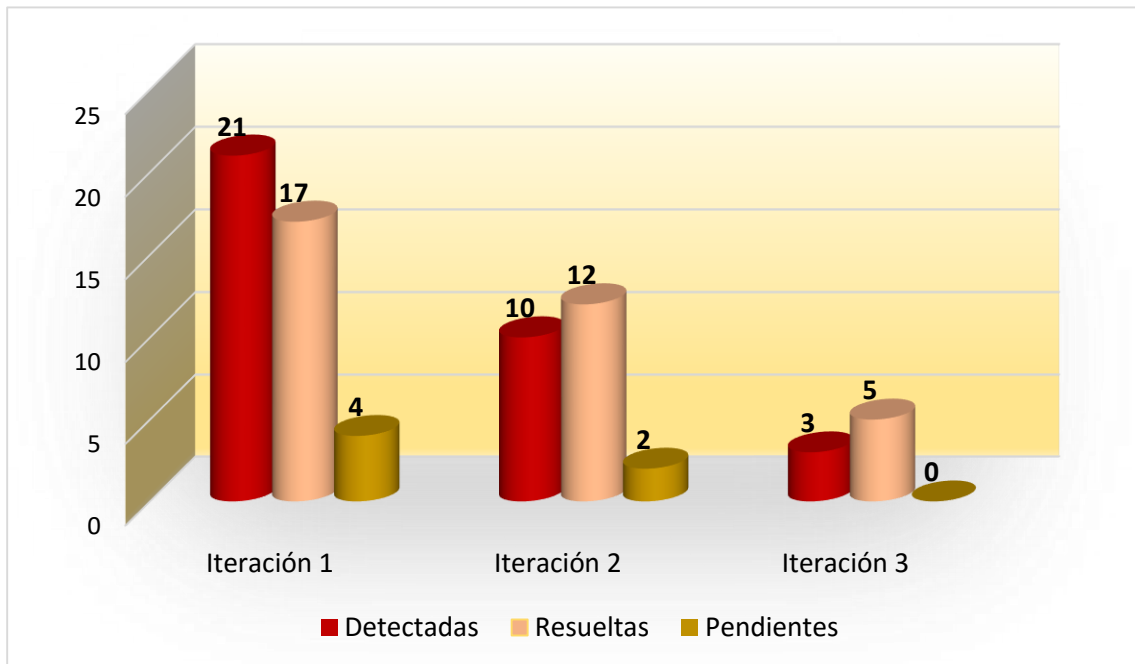
Para el seguimiento de todo el proceso de corrección de no conformidades se realiza una tabla, la misma contará con el requisito, la cantidad de no conformidades detectadas (NC), la cantidad de no conformidades significativas (S) y la cantidad de no significativas (NS) por cada historia de usuario:

**Tabla 11** Resultados de las pruebas de caja negra por iteraciones.

No	Historia de usuario	Iteración 1			Iteración 2			Iteración 3		
		NC	S	NS	NC	S	NS	NC	S	NS
1	Instalar bloque Actividades compartidas.	-	-	-	-	-	-	-	-	-
2	Configurar bloque Actividades compartidas.	1	-	1	-	-	-	-	-	-
3	Compartir actividad.	4	1	3	1	1	-	-	-	-
4	Buscar AC(básica).	2	-	2	-	-	-	-	-	-
5	Buscar AC(avanzada).	3	1	2	-	-	-	-	-	-
6	Agregar comentario en el foro desde el repositorio.	-	-	-	2	-	2	-	-	-
7	Agregar nuevo comentario en el foro.	-	-	-	-	-	-	-	-	-
8	Responder comentario en el foro.	-	-	-	1	-	1	-	-	-
9	Eliminar comentario en el foro.	-	-	-	1	-	1	-	-	-
10	Editar comentario en el foro.	-	-	-	1	1	-	-	-	-
11	Visualizar comentario en el foro.	-	-	-	-	-	-	-	-	-
12	Previsualizar AC	1	1	-	1	1	-	-	-	-
13	Calificar AC.	1	1	-	-	-	-	-	-	-
14	Notificar AC modificadas.	-	-	-	1	-	1	-	-	-
15	Notificar AC con nuevos comentarios.	-	-	-	1	-	1	-	-	-

16	Notificar AC que subieron de ranking.	-	-	-	1	-	1	-	-	-
17	Utilizar AC en el curso.	2	1	1	1	1	-	-	-	-
18	Editar datos de AC	5	1	4	-	-	-	-	-	-
19	Eliminar AC del repositorio.	1	-	1	-	-	-	-	-	-
20	Mostrar AC a repositorios externos.	1	1	-	1	1	-	1	1	-
21	Exportar repositorio de actividades compartidas.	-	-	-	-	-	-	-	-	-
22	Importar repositorio de actividades compartidas.	-	-	-	-	-	-	-	-	-
23	Desinstalar bloque Actividades compartidas.	-	-	-	-	-	-	-	-	-
24	Mostrar repositorio de actividades compartidas..	-	-	-	2	-	2	1	-	1
25	Escoger cantidad de resultados por página.	-	-	-	-	-	-	2	-	2
26	Ordenar búsqueda de AC.	-	-	-	-	-	-	1	1	-
	<b>Total</b>	<b>21</b>	<b>7</b>	<b>14</b>	<b>14</b>	<b>5</b>	<b>9</b>	<b>5</b>	<b>2</b>	<b>3</b>

A continuación, se muestra un gráfico donde se puntualiza por iteraciones el total de no conformidades identificadas, el total de no conformidades resueltas y la cantidad de no conformidades pendientes. Para consultar las no conformidades detectadas por iteraciones remitirse al Anexo 6.



**Figura 21** Resultados de las pruebas de caja negra.

## Conclusiones parciales

- Mediante los diagramas de despliegue y de componentes se mostró, tanto la distribución física como la representación lógica de los elementos del diseño, que permitieron llevar a cabo la correcta implementación de la solución propuesta.
- Las pruebas realizadas en las tres iteraciones arrojaron un total de catorce no conformidades significativas y veintiséis no significativas, las cuales quedaron totalmente solucionadas en la última iteración.

## CONCLUSIONES GENERALES

---

Con la realización de la presente investigación se brinda solución a los objetivos trazados arrojándose a los siguientes resultados:

- El análisis realizado a las soluciones existentes que permiten la compartición de actividades en la plataforma Moodle, confirmó la necesidad de un bloque que permitiera compartir actividades entre los cursos de la plataforma.
- El modelo de análisis y diseño desarrollado; correspondiente a la metodología seleccionada, permitió obtener una abstracción más cercana a la implementación del sistema propuesto.
- Mediante la puesta en práctica de las pruebas de caja negra, se pudo detectar y corregir los errores encontrados, logrando con esto verificar el perfecto funcionamiento de la solución propuesta.
- Se obtuvo el bloque Actividades compartidas para la plataforma Moodle en la versión 2.9.x, el cual permite compartir y facilitar la reutilización de actividades entre los cursos de la plataforma.

## RECOMENDACIONES

---

A partir de los resultados obtenidos con la investigación los autores proponen las siguientes recomendaciones para futuros trabajos tomando como referencia el actual:

- Ampliar el sistema de notificaciones del bloque Actividades compartidas, abarcando el sistema de mensajería hacia los tipos de notificaciones que usa Moodle.
- En próximas actualizaciones del bloque Actividades compartidas se agregue una funcionalidad que permita respetar los términos de las licencias bajo la cual se comparten las actividades.
- Agregar funcionalidades que permitan compartir las actividades entre distintas plataformas de Moodle.

## REFERENCIAS BIBLIOGRÁFICAS

---

1. *Conceptualización de una red social educativa que integre de forma colaborativa las aplicaciones e-learning de la Universidad de las Ciencias Informáticas*. Guerra, Yulaine Arias. 3, s.l. : Revista Cubana de Ciencias Informáticas, 2011, Vol. Vol.5. ISSN-e: 2227.1899.
2. viu -Universidad Internacional de Valencia. [En línea] [Citado el: 2015 de octubre de 26 .] <http://www.viu.es/caracteristicas-tipos-y-plataformas-mas-utilizadas-para-estudiar-a-distancia/>.
3. *Los entornos virtuales de aprendizaje y el e-learning*. Magaña, Jesús Aurelio Castro. 14, Yucatán : s.n., 2015. ISSN 2007-4751.
4. Portal web del Centro de Formación Permanente de la Universidad de Sevilla. [En línea] [Citado el: 1 de noviembre de 2015.] <http://www.cfp.us.es/e-learning-definicion-y-caracteristicas>.
5. Centro de Comunicación y Pedagogía. [En línea] [Citado el: 3 de noviembre de 2015.] <http://www.centrocp.com/lms-y-lcms-funcionalidades-y-beneficios/>.
6. e-historia. e-historia. *e-historia*. [En línea] [Citado el: 12 de noviembre de 2015.] <http://www.e-historia.cl/e-historia/%C2%BFque-es-moodle/>.
7. Universidad Luterana Salvadoreña. [En línea] [Citado el: 13 de noviembre de 2015.] [http://www.uls.edu.sv/pdf/manuales\\_moodle/queesmoodle.pdf](http://www.uls.edu.sv/pdf/manuales_moodle/queesmoodle.pdf).
8. Entornos educativos. [En línea] [Citado el: 27 de octubre de 2015.] <http://www.entornos.com.ar/moodle>.
9. Moodle. *Moodle*. [En línea] [Citado el: 15 de noviembre de 2015.] [https://docs.moodle.org/all/es/dev/Historia\\_de\\_las\\_versiones](https://docs.moodle.org/all/es/dev/Historia_de_las_versiones).
10. intef. *intef*. [En línea] [Citado el: 25 de diciembre de 2015.] <http://recursostic.educacion.es/observatorio/web/es/software/servidores/789-monografico-moodle..>
11. Blog de Luis Castellanos. *Blog de Luis Castellanos*. [En línea] [Citado el: 12 de diciembre de 2015.] <https://luiscastellanos.wordpress.com/tecnologia/moodle/>.
12. Moodle. *Moodle*. [En línea] [Citado el: 11 de diciembre de 2015.] <https://docs.moodle.org/all/es/Actividades>.
13. Moodle. *Moodle*. [En línea] [Citado el: 24 de marzo de 2016.] <https://docs.moodle.org/all/es/Bloques>.



14. Universidad Politécnica de Madrid. *Universidad Politécnica de Madrid*. [En línea] 2013. [Citado el: enero de 25 de 2016.] [http://serviciosgate.upm.es/docs/moodle/manual\\_moodle\\_2.4.pdf](http://serviciosgate.upm.es/docs/moodle/manual_moodle_2.4.pdf).
15. Foronda, Eladio Pascual. *Diccionario manual de la lengua española*. 2007. 84833210x,9788483321058.
16. Biblioteca Universitaria. *Biblioteca Universitaria*. [En línea] [Citado el: 7 de enero de 2016.] <http://biblioteca.unileon.es/ayuda-formacion/repositorio-institucional>.
17. e-prints in library & information science. [En línea] [Citado el: 15 de noviembre de 2015.] [http://eprints.rclis.org/10677/1/COMO\\_FUNCIONA\\_EL\\_PROTOCOLO\\_OAI\\_-\\_PMH\\_EN\\_LA\\_RECUPERACION\\_DE\\_INFORMACION.pdf](http://eprints.rclis.org/10677/1/COMO_FUNCIONA_EL_PROTOCOLO_OAI_-_PMH_EN_LA_RECUPERACION_DE_INFORMACION.pdf).
18. *Interoperabilidad en los Sistemas de Información Documental (SID): la información debe fluir*. Dueñas, Laureano Felipe Gómez. 1, 2007, Vol. Vol. 3 .
19. Vniversitat de Valencia. *Vniversitat de Valencia*. [En línea] [Citado el: 8 de diciembre de 2016.] [www.uv.es/=barrueco/cardedeu.doc](http://www.uv.es/=barrueco/cardedeu.doc).
20. Metadata Innovation. *Metadata Innovation*. [En línea] [Citado el: 11 de diciembre de 2016.] <http://dublincore.org/>.
21. *Z39.50 y OAI-PMH: Protocolos de Transferencia y Rec*. Salinas, Aldo Guajardo. 2010.
22. Las herramientas web 2.0 y la educación en la Facultad de Ciencias para la Salud. [En línea] [Citado el: 17 de noviembre de 2015.] <http://web2facsalud.blogspot.com/2010/12/licencias-y-derechos-de-autor-en.html>.
23. Hincapié, Natalia Mejía. Aprende en línea. *Aprende en línea*. [En línea] [Citado el: 12 de diciembre de 2015.] <http://aprendeenlinea.udea.edu.co/lms/sitio/mod/forum/view.php?id=82>.
24. Creative Commons. [En línea] [Citado el: 4 de diciembre de 2015.] <https://creativecommons.org/licenses/?lang=es>.
25. Universidad Politécnica de Madrid. [En línea] [Citado el: 16 de diciembre de 2015.] [http://serviciosgate.upm.es/docs/asesoramiento/Gamificar\\_Moodle.pdf](http://serviciosgate.upm.es/docs/asesoramiento/Gamificar_Moodle.pdf).
26. Desarrollo web. [En línea] [Citado el: 17 de diciembre de 2015.] <http://www.desarrolloweb.com/articulos/243.php>.
27. PHP YA. [En línea] [Citado el: 20 de diciembre de 2015.] <http://www.phpya.com.ar/temarios/descripcion.php?cod=23>.

28. PROGRAMACIÓN WEB. [En línea] [Citado el: 7 de enero de 2016.] <http://www.paginasprodigy.com.mx/evaristopacheco/pweb/clienteServidor.html>.
29. Definición de XML. [En línea] [Citado el: 9 de enero de 2016.] <definicion.de/xml/>.
30. AJPD-Softjpdsof. [En línea] [Citado el: 10 de enero de 2016.] <http://www.ajpdsoft.com/modules.php?name=Encyclopedia&op=content&tid=820..>
31. Tecnología. [En línea] [Citado el: 12 de enero de 2016.] [http://www.ehowenespanol.com/ventajas-apache-web-server-lista\\_109947/](http://www.ehowenespanol.com/ventajas-apache-web-server-lista_109947/).
32. geekWare. [En línea] [Citado el: 20 de enero de 2016.] [http://danielpecos.com/docs/mysql\\_postgres/x15.html..](http://danielpecos.com/docs/mysql_postgres/x15.html..)
33. González, Carlos D. Curso Base de Datos PostgreSQL, SQL avanzado y PHP. *Curso Base de Datos PostgreSQL, SQL avanzado y PHP*. [En línea] [Citado el: 18 de febrero de 2016.] <http://www.usabilidadweb.com.ar/postgre.php>.
34. Slide Share. *Slide Share*. [En línea] [Citado el: 23 de febrero de 2016.] <http://es.slideshare.net/HernanOkamura/sistemas-de-gestores-de-base-de-datos-13332504>.
35. Programación Web y Tecnologías Informáticas. [En línea] [Citado el: 14 de febrero de 2016.] <http://zenkius.blogspot.com/2008/02/tecnologias-del-ladodelcliente.html>.
36. W3C. [En línea] [Citado el: 15 de febrero de 2016.] <http://www.w3.org>.
37. visualweb. [En línea] [Citado el: 16 de febrero de 2016.] <http://crearweb.com/>.
38. *PHP y MySQL Tecnologías para el desarrollo de aplicaciones web*. s.l. : Ediciones Díaz de Santos, 2005. ISBN 84-7978-706-6.
39. Maestros del Web. [En línea] [Citado el: 16 de febrero de 2016.] <http://www.maestrosdelweb.com/editorial/%C2%BFque-es-javascript/>.
40. Pérez, Javier Eguíluz. *Introducción a JavaScript*. s.l. : s.l.: Autoedición, 2009. pág. pág. 140p.
41. jQuery. *jQuery*. [En línea] [Citado el: 10 de diciembre de 2016.] <https://jquery.com>.
42. LIBROSWEB. *LIBROSWEB*. [En línea] [Citado el: 10 de diciembre de 2016.] [http://librosweb.es/libro/css\\_avanzado/capitulo\\_5/el\\_framework\\_yui.html](http://librosweb.es/libro/css_avanzado/capitulo_5/el_framework_yui.html).
43. updatestar. *updatestar*. [En línea] [Citado el: 18 de febrero de 2016.] <http://netbeans-ide.updatestar.com/es>.

44. James Rumbaugh, Ivar Jacopson, Grady Booch. *El Lenguaje Unificado de Modelado. Manual de referencia.* 2005.
45. Slide Share. *Slide Share.* [En línea] [Citado el: 20 de febrero de 2016.] <http://es.slideshare.net/guestf131a9/herramientas-case>.
46. Visual Paradigm. [En línea] [Citado el: 22 de febrero de 2016.] <http://www.visual-paradigm.com/>.
47. Ingeniería de Software. [En línea] [Citado el: 26 de febrero de 2016.] [http://ingenieriadesoftware.mex.tl/63758\\_AUP.html](http://ingenieriadesoftware.mex.tl/63758_AUP.html).
48. Sánchez, Tamara Rodríguez. *Metodología de desarrollo para la Actividad productiva de la UCI.* La Habana : s.n.
49. *MODELO DEL DOMINIO.* Larman, Craig. s.l. : Prentice Hall, 2003, Vol. 2ª Edición.
50. Pressman, Roger. *Ingeniería del Software: Un Enfoque Práctico.* 2005. . 2005. ISBN: 9701054733. .
51. Sommerville, Iam. *Ingeniería de Software Séptima Edición.* Madrid : Pearson Educación : s.n., 2005. ISBN 84-7829-074-5. .
52. *Métodologías ágiles para el desarrollo de software: eXtreme Programming (XP).* Letelier, Patricio. 26, Buenos Aires : s.n., 2006, Vol. 5. ISSN 1666-1680.
53. Monografías.com. *Monografías.com.* [En línea] [Citado el: 3 de mayo de 2016.] <http://www.monografias.com/trabajos28/proyecto-uml/proyecto-uml.shtml>.
54. Genbeta:dev. *Genbeta:dev.* [En línea] [Citado el: 25 de abril de 2016.] <http://www.genbetadev.com/desarrollo-web/escribiendo-modulos-para-moodle-introduccion>.
55. Juan, Francisco Javier Martínez. Departamento de Informática. *Departamento de Informática.* [En línea] [Citado el: 25 de abril de 2016.] <http://di002.edv.uniovi.es/~cueva/ asignaturas/PFCOviedo/PFCpatronesJava.pdf>.
56. Prácticas de software. *Prácticas de software.* [En línea] [Citado el: 25 de abril de 2016.] <http://www.practicadesoftware.com.ar/2011/03/patrones-grasp/>.
57. Osmosis Latina. *Osmosis Latina.* [En línea] [Citado el: 26 de abril de 2016.] <http://www.osmosislatina.com/lenguajes/uml/clasesob.htm>.
58. ITDOCS. *ITDOCS.* [En línea] [Citado el: 29 de abril de 2016.] <http://www.it-docs.net/doc/3795/introduccion-al-diseno-de-bases-de-datos.html>.

59. Slide Share. *Slide Share*. [En línea] [Citado el: 1 de mayo de 2016.] <http://es.slideshare.net/joshell/diagramas-uml-componentes-y-despliegue>.
60. Eleven Path. *Eleven Path*. [En línea] [Citado el: 3 de mayo de 2016.] <http://blog.elevenpaths.com/2014/09/qa-pruebas-para-asegurar-la-calidad-del.html>.
61. Bueno, Carlos Blanco. open course ware. *open course ware*. [En línea] [Citado el: 3 de mayo de 2016.] <http://ocw.unican.es/enseanzas-tecnicas/ingenieria-del-software-ii/materiales>.
62. Angelfire. *Angelfire*. [En línea] [Citado el: 20 de mayo de 2016.] <http://www.angelfire.com/empire2/ivansanes/bywbox.htm>.