



Universidad de las Ciencias Informáticas

Facultad 4

Desarrollo del plugin External 2.0 para enlazar los objetos de aprendizaje del repositorio RHODA con Moodle v2.3.x

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

Autora:

Daimaris Reyes Curbelo

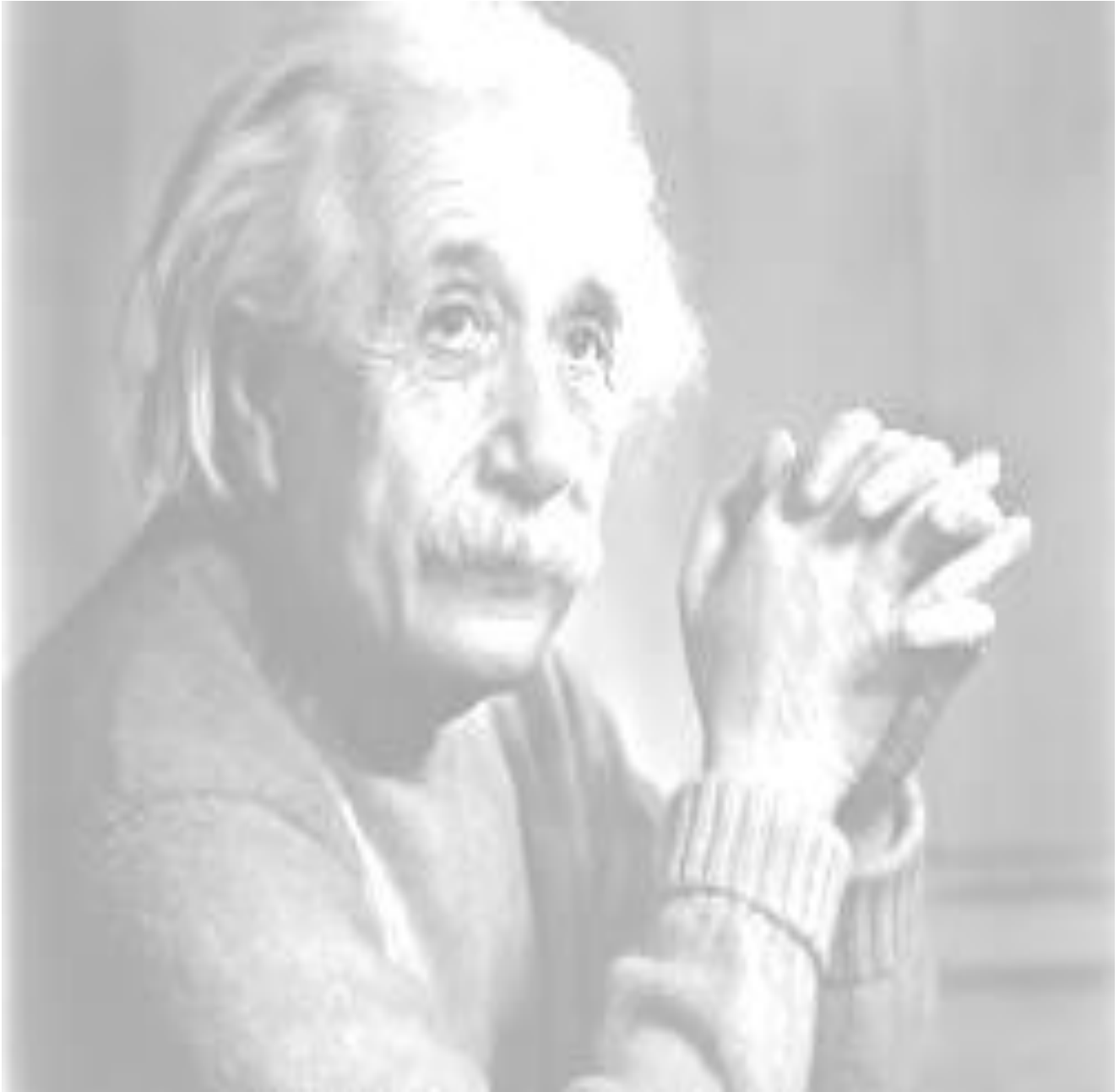
Tutores:

Ing. Carlos Acosta Montejo

Ing. Javier Soler Martín

Co-tutora: Ing. Suset Fernández Rojas

Ciudad de La Habana, Junio de 2013



“Nunca consideres el estudio como una obligación sino como una oportunidad para penetrar en el bello y maravilloso mundo del saber”

Albert Einstein

DECLARACIÓN DE AUTORÍA

Declaración de autoría

Declaro ser la única autora del trabajo “Desarrollo del plugin External 2.0 para enlazar los objetos de aprendizaje del repositorio RHODA con Moodle v2.3.x” y autorizo a la Facultad 4 de la Universidad de las Ciencias Informáticas a hacer uso del mismo, con carácter exclusivo.

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

Daimaris Reyes Curbelo

Firma de la autora

Ing. Carlos Acosta Montejo

Firma de la tutor

Ing. Javier Soler Martín

Firma de la tutor

Ing. Suset Fernández Rojas

Firma de la co-tutora

Datos de contacto

Tutores:

Ing. Carlos Acosta Montejo: Ingeniero en Ciencias Informáticas, profesor Instructor. Se gradúa en el 2007 en la Universidad de las Ciencias Informáticas y comienza a trabajar en la dirección de Teleformación, en el 2009 comienza a laborar en el Centro FORTES, en ambos lugares se ha desempeñado como especialista de implantación administración y soporte de la plataforma Moodle, por lo que acumula 5 años de experiencia en el trabajo con dicha herramienta.

Ing. Javier Soler Martín: Ingeniero en Ciencias Informáticas. Se gradúa en el 2010 en la Universidad de las Ciencias Informáticas y comienza a trabajar en el Centro FORTES, donde se ha desempeñado como desarrollador del Repositorio de Objetos de Aprendizaje de la Universidad. Acumula 2 años de experiencia en el trabajo con dicha herramienta

Suset Fernández Rojas: Ingeniera en Ciencias Informáticas, profesor Asistente. Se gradúa en el 2008 en la Universidad de las Ciencias Informáticas fecha desde la cual comienza a desempeñarse como especialista de la Dirección Técnica de la Infraestructura productiva en temas relacionados con el desarrollo colaborativo de software, además de impartir clases de Metodología de la Investigación Científica por 3 años. Actualmente se desempeña como especialista del Centro de Soporte UCI.

Agradecimientos

- **A mis padres:** gracias por brindarme su amor, su apoyo incondicional, su confianza, su cariño y su aliento para hacer realidad uno de mis más grandes sueños: ser una profesional. A ustedes gracias por enseñarme que una verdadera persona es aquella que se sabe levantar con la frente en alto tras los tropiezos que nos impone la vida. Gracias una vez más por ser esas personas incondicionales, dedicadas y comprensivas, por saberme apoyar y darme aliento cada vez que lo necesito, por forjar en mí sentimientos y valores maravillosos como el amor y la honestidad.
- **A mi novio Yadiel:** por brindarme todo su amor y cariño incondicionalmente, por apoyarme y alentarme durante toda mi carrera. Por siempre encontrar en él un consejo y ayuda en momentos difíciles. Por no dejarme ni un instante sola cuando lo necesité y sobre todas las cosas por su paciencia y comprensión. Gracias mi amor, te quiero mucho.
- **A mis abuelos:** por ser mis segundos padres y aunque todos no estén acá ahora gracias por estar siempre a mi lado dándome apoyo y amor. Gracias por ayudarme a ser la persona que soy hoy, por alentarme cuando en frente tenía una tormenta, por no dejarme caer o ayudarme a levantar para hacer mis sueños realidad.
- **A mis hermanos y cuñadas:** Gracias mis hermanitos por apoyarme y siempre creer en mí, los quiero mucho, son los mejores hermanos del mundo y a mis cuñadas Yailin y Yarkenía muchas gracias por ayudarme a hacer realidad este sueño.
- **A mi familia:** Gracias a todos por confiar en mí, en especial a mis hermanitas Yaidelin, Yailin y Yanelsy que aunque no sean mis hermanitas de sangre las quiero mucho y un eterno agradecimiento a mi padrino Benítez, a Benita y Lázara por apoyarme y estar ahí cuando más los he necesitado.
- **A mis suegros y familia:** Muchas gracias por dejar que entrara en su familia, por recibirme con tanto amor y apoyarme en todo, los quiero mucho.
- **A mis amistades de toda la vida:** A ustedes un millón de gracias por su apoyo, su confianza, por creer en mí y por estar siempre a mi lado cuando más lo necesité, en especial a mis amigos Yulsi, Juan Pablo, Dora, Susel, Yaquelin y Manuel.
- **A mis verdaderos amigos de la escuela:** Gracias a ustedes mis amigos queridos por ayudarme, apoyarme y estar conmigo en aquellos momentos tristes, no tan tristes y muy felices de mi vida en especial a Anita, Liuvis Dianne, Yiliana, Roilandy, Marlon, Augusto y Odenys.
- **A mis tutores:** Muchas gracias por todo su apoyo, su confianza y su fuerza, han sido un motor importante en mi vida, nunca los olvidaré en especial a la profesora Suset.
- **A todas aquellas personas que me brindaron su amistad y ayuda siempre que las necesité, que fueron fuente de fuerza y alegría y me impulsaron a seguir adelante hasta alcanzar la meta.**

Dedicatoria

- *A mi abuelita Matilde (Llille) que aunque ya no esté a mi lado sigue aquí en mi mente y en mi corazón, es mi guía y mi motor de empuje, por ella estoy aquí, cumpliendo mi sueño.*
- *A mis papás por su amor y perseverancia, por ser lo más importante en mi vida, los que desde pequeña me guiaron por el buen camino y me enseñaron a nunca rendirme ante las dificultades. Gracias por ayudarme a convertir mi sueño en realidad, los amo.*
- *A Yadiel por apoyarme y darme todo su amor, por ser esa pareja incondicional. Te amo.*



Resumen

En la Universidad de las Ciencias Informáticas (UCI) se utiliza la plataforma de teleformación Moodle como sustento tecnológico del Entorno Virtual de Aprendizaje (EVA) desde el año 2005. En 2012 se decidió migrar a la versión 2.3.x de Moodle, la cual cuenta con una arquitectura de software diferente a la de las versiones anteriores. El cambio de versión de la plataforma hace incompatibles las funcionalidades, plugins y módulos que han sido desarrollados en la UCI para las anteriormente utilizadas. El centro FORTES se ha propuesto lanzar en septiembre de 2013 el EVA utilizando la versión 2.3.x de Moodle, por lo que es necesario reimplementar las funcionalidades que habían sido añadidas a la plataforma y que por los cambios arquitectónicos de la versión no son funcionales. Como resultado de esta investigación se implementó el plugin External 2.0 que permite enlazar como un recurso los objetos de aprendizaje que se encuentran publicados en el repositorio de la Universidad: RHODA. Para el desarrollo del plugin se utilizó la metodología de desarrollo XP, como herramienta CASE el Visual Paradigm para UML (lenguaje de modelado), como lenguajes de programación para la implementación de esta aplicación se utilizaron XML v1.0, XHTML v1.0, CCS v2.0, Javascript, del lado del cliente y como lenguaje del lado del servidor PHP v5.38. El plugin External 2.0 permite a los profesores y estudiantes mediante una interfaz visual sencilla enlazar el objeto de aprendizaje como un recurso dentro del EVA.

Palabras claves: Moodle, plugin, repositorio, objeto de aprendizaje, RHODA, EVA.

Índice de contenido

INTRODUCCIÓN	1
CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA	6
Introducción	6
1.1 La plataforma Moodle	6
Arquitectura de Moodle	6
Plugins	7
Moodle en la UCI.....	8
1.2 Conceptualizando	9
Objetos de Aprendizaje	9
Estándar para la descripción de contenidos educativos	10
Estándar para el empaquetamiento de contenidos.....	11
1.3 Estado del Arte	12
Internacionalmente.....	12
En Cuba.....	13
Plugin External 1.0	13
Conclusiones del estudio realizado	14
1.4 Metodología, lenguajes y herramientas a utilizar	14
1.4.1 Metodología de desarrollo.....	15
1.4.2 Lenguaje de Modelado	16
1.4.3 Lenguajes empleados para el desarrollo de la aplicación	17
1.4.4 Herramientas de Desarrollo.....	20
Conclusiones parciales	22
CAPÍTULO 2. DISEÑO DEL PLUGIN EXTERNAL 2.0	23
Introducción	23
2.1 Descripción de los procesos vinculados al campo de acción	23
Configurar actividad objeto de aprendizaje.....	23
Enlazar el objeto de aprendizaje.....	23
Visualizar el objeto de aprendizaje.....	24
2.2 Propuesta del plugin External 2.0	24
2.3 Principales particularidades del plugin External 2.0	24
2.3.1 Funcionalidades del sistema (FS)	24

ÍNDICE DE CONTENIDO

2.3.2 Características del sistema.....	25
2.3.3 Personal relacionado con la aplicación.....	26
2.4 Planificación	27
2.4.1 Historias de usuario.....	27
2.4.2 Plan de Entrega.....	31
2.5 Diseño.....	32
2.5.1 Tarjetas CRC	32
2.6 Diagrama de Despliegue	33
2.7 Arquitectura del sistema.....	34
Conclusiones del Capítulo.....	36
<i>CAPÍTULO 3. IMPLEMENTACIÓN Y PRUEBAS DEL PLUGIN 2.0.....</i>	<i>37</i>
Introducción.....	37
3.1 Estimación de esfuerzos por historias de usuario	37
3.2 Iteraciones	37
Primera iteración	38
Segunda iteración	40
Plan de duración de las iteraciones.....	41
3.3 Pruebas	41
Prueba de aceptación.....	42
3.4 Resultados y funcionalidades obtenidas.....	46
3.5 Aporte social y económico	47
Conclusiones del Capítulo.....	48
<i>Conclusiones.....</i>	<i>49</i>
<i>Recomendaciones</i>	<i>50</i>
<i>Bibliografía Referenciada</i>	<i>51</i>
<i>Bibliografía Consultada</i>	<i>55</i>
<i>Glosario de Términos.....</i>	<i>59</i>
<i>Anexos.....</i>	<i>60</i>
Anexo 1: Punto de partida para la selección de la metodología, lenguajes y herramientas.....	60

ÍNDICE DE CONTENIDO

Anexo 2: Entrevista realizada a profesores editores.....	62
Anexo 3: Interfaces de la sistema	63
Anexo 4: Registro de defectos y dificultades detectados	69

ÍNDICE DE TABLA

Tabla 1: Descripción del personal relacionado con el sistema.	27
Tabla 2: Descripción de la Historia de Usuario: Configurar el Objeto de Aprendizaje.....	28
Tabla 3: Descripción de la Historia de Usuario: Crear la actividad Objeto de Aprendizaje.	29
Tabla 4: Descripción de la Historia de Usuario: Buscar en el repositorio RHODA.	30
Tabla 5: Descripción de la Historia de Usuario: Enlazar los objetos de Aprendizaje.	30
Tabla 6: Descripción de la Historia de Usuario: Visualizar el Objeto de Aprendizaje.....	31
Tabla 7: Plan de Entrega.	32
Tabla 8: Tarjeta CRC 1.Clase: Roa.....	32
Tabla 9: Tarjeta CRC 2.Clase: Tree.....	33
Tabla 10: Tarjeta CRC 3.Clase: Moodle_Queck_chooserepositoryfile	33
Tabla 11: Estimación de esfuerzos por historias de usuario.....	37
Tabla 12: Tarea: Configurar el Objeto de Aprendizaje.	39
Tabla 13: Tarea: Crear actividad Objeto de Aprendizaje.....	39
Tabla 14: Tarea: Seleccionar objetos de aprendizaje del repositorio RHODA.....	40
Tabla 15: Tarea: Enlazar los objetos de Aprendizaje.....	40
Tabla 16: Tarea: Visualizar el Objeto de Aprendizaje.....	41
Tabla 17: Plan de duración de las iteraciones.....	41
Tabla 18: Caso de Prueba de Aceptación: Configurar el objeto de aprendizaje	43
Tabla 19: Caso de Prueba de Aceptación: Crear actividad Objeto de Aprendizaje	44
Tabla 20: Caso de Prueba de Aceptación: Seleccionar objetos de aprendizaje del repositorio RHODA	44
Tabla 21: Caso de Prueba de Aceptación: Enlazar los Objetos de Aprendizaje.....	45
Tabla 22: Caso de Prueba de Aceptación: Visualizar el Objeto de Aprendizaje.....	46
Tabla 23: Registro de defectos y dificultades detectados.....	69

INTRODUCCIÓN

Con el uso de las tecnologías educativas, programas educativos, contenidos digitales y aplicaciones basadas en las Tecnologías de la Información y las Comunicaciones (TICs), se crean nuevos entornos de aprendizaje que posibilitan la comunicación, acción e interacción entre alumnos y profesores. La introducción de estas en el ámbito educativo ha propiciado el surgimiento del e-learning como una alternativa viable para la formación a distancia, rompiendo toda barrera de tiempo y espacio. El e-learning no sustituye el modelo tradicional de enseñanza pero sí beneficia en gran medida el proceso educativo. Esta modalidad permite una interacción entre el profesor y el alumno, así como una mayor colaboración y participación de estos en las actividades propuestas.

Un lugar relevante en el mundo del e-learning lo ocupan los Sistemas de Gestión del Aprendizaje (LMS), sistemas que permiten, además de crear y administrar los contenidos, controlar el proceso de enseñanza-aprendizaje. Entre los LMS más difundidos se encuentra el Entorno de Aprendizaje Dinámico Orientado a Objetos y Modular (Moodle), plataforma gratuita de gran flexibilidad que permite la creación de cursos en línea. (Mi Portal, 2011).

La plataforma Moodle es una aplicación web, diseñada para dar soporte a la educación social constructivista¹. (Moodle.org, 2010). Permite la creación de cursos en línea y llevar el control de los contenidos y los usuarios que interactúan dentro de la misma. Para el desarrollo de esta, cuenta con una comunidad oficial en www.moodle.org, donde participan desarrolladores, profesores y estudiantes de aproximadamente, doscientos doce países. Entre sus objetivos, se encuentra el desarrollo colaborativo de módulos y personalizaciones de esta plataforma.

En Cuba, la aplicación de las TIC en los centros de educación está propiciando una serie de transformaciones entre las que se encuentran el uso de herramientas informáticas y telemáticas dentro de nuevas concepciones. De esta manera, en la Universidad de las Ciencias Informáticas (UCI) a partir del año 2005 se utiliza la plataforma de gestión del aprendizaje Moodle, como parte de su programa de teleformación, integrando concepciones pedagógicas y tecnológicas que aumentan la calidad del proceso enseñanza-aprendizaje.

¹ Creación colaborativa de artefactos compartidos de un grupo social para otro.

Durante el curso 2009-2010 en la Facultad 4, se crea el Centro de Tecnologías para la Formación (FORTES). Este se especializa en “desarrollar tecnologías que permiten ofrecer servicios y productos para la implementación de soluciones de formación, aplicando las TIC con diferentes modelos de formación y condiciones tecnológicas” (FORTES, 2010). El mismo está compuesto por 3 departamentos entre los que se encuentra el Departamento de Implantación y Soporte Técnico (DIST). En este último se encuentra el proyecto Moodle encargado de realizar extensiones a la plataforma de teleformación del mismo nombre. La personalización de esta herramienta para la UCI ha sido nombrada EVA: Entorno Virtual de Aprendizaje.

También asociado a la utilización de herramientas informáticas para la enseñanza en el centro FORTES se desarrolló RHODA, un repositorio de objetos de aprendizaje cuyo objetivo fundamental radica en almacenar y gestionar objetos de aprendizaje. Este repositorio facilita la reutilización de los objetos de aprendizaje, así como la disponibilidad de múltiples recursos educativos de utilidad para profesores y estudiantes de la Universidad.

Los Objetos de Aprendizaje (OA) o recursos educativos son archivos o elementos digitales con cierto nivel de interactividad e independencia, que podrán utilizarse o ensamblarse, sin modificación previa, en diferentes situaciones de enseñanza-aprendizaje. (García, 2005).

La instalación estándar de Moodle no cuenta con una herramienta que permita enlazar los objetos de aprendizaje que se encuentran en el repositorio de la UCI (RHODA). Para incluir los recursos educativos publicados en RHODA en la plataforma Moodle original, es necesario realizar el procedimiento siguiente:

1. Acceder al Repositorio de Objetos de Aprendizaje.
2. Buscar el OA deseado.
3. Exportar el OA como un paquete SCORM² (1.2 o 2004).
4. Acceder a la plataforma Moodle con los permisos necesarios para editar un curso y activar la edición del mismo.
5. Agregar una actividad SCORM.

² Conjunto de estándares y especificaciones que permite crear objetos pedagógicos estructurados

6. Completar los datos solicitados.
 - a. Agregar el OA como un recurso del curso.
7. Guardar los cambios realizados.

Este procedimiento implica la realización de un trabajo engorroso, además de duplicar la existencia del recurso en Moodle y RHODA. Por otro lado si un profesor o varios profesores, utilizan el mismo objeto de aprendizaje en más de un curso deben adicionarlo como una actividad para cada uno de esos cursos y como consecuencia se duplica dicha información en el Entorno Virtual del Aprendizaje.

En aras de solucionar las deficiencias antes mencionadas en el Centro FORTES en 2010 se desarrolló un plugin para enlazar como un recurso los objetos de aprendizaje que se encuentran en el repositorio de la UCI para la versión 1.9.12+ de Moodle. Este es utilizado actualmente en el EVA con excelentes resultados.

El 24 de noviembre de 2010 se lanza la versión 2.0 de Moodle y en 2012 en la Universidad se decide migrar a esta nueva versión debido a las ventajas que ofrece sobre las anteriores. Entre ellas: la integración de Moodle con webservices, repositorios, nuevos bloques, actividades condicionales, cohortes, entre otras. Además la nueva versión facilita el desarrollo de plugins y la realización de mejoras gráficas con mayor personalización.

Sin embargo Moodle a partir de la v2.0 presenta cambios significativos en el *núcleo* de la plataforma lo cual impide utilizar las funcionalidades, plugins y módulos desarrollados para las versiones anteriores. En consecuencia, al migrar de versión no se podrá utilizar el plugin existente para enlazar los objetos de aprendizaje disponibles en RHODA con el EVA: External 1.0 y reaparecerán inconvenientes como:

- Complejidad para reutilizar en el EVA los objetos de aprendizaje publicados en RHODA.
- Duplicación de la información en el EVA.
- Utilización inadecuada del espacio de los servidores donde se aloja la plataforma.

Por todo lo planteado anteriormente se tiene como **problema a resolver**: ¿Cómo lograr que en la versión 2.3.x de Moodle se puedan utilizar como un recurso los objetos de aprendizaje publicados en RHODA?

Esta investigación tiene como **objeto de estudio**: Interoperabilidad entre Moodle y los repositorios de recursos educativos.

Para dar solución al problema se define como **objetivo general**: Desarrollar un plugin que permita utilizar desde Moodle versión 2.3.x los objetos de aprendizaje publicados en RHODA.

Se define como **campo de acción**: Interoperabilidad entre Moodle v2.3.x y RHODA.

Se plantea como **idea a defender** la siguiente: El desarrollo del plugin External 2.0 permitirá utilizar como un recurso desde la plataforma de teleformación Moodle v2.3.x los objetos de aprendizaje publicados en RHODA contribuyendo así a la interoperabilidad entre ambos sistemas.

Para dar cumplimiento al objetivo general de la investigación se definen los siguientes **objetivos específicos**:

- Elaborar los referentes teóricos sobre la creación de plugins para vincular Moodle con repositorios de recursos educativos de modo que se avale el problema y se sustente la propuesta de solución.
- Describir la metodología, lenguajes y herramientas a utilizar durante el desarrollo del plugin External 2.0.
- Generar los artefactos del análisis y diseño necesarios para guiar el resto del proceso de desarrollo.
- Implementar las funcionalidades requeridas para el plugin External 2.0.
- Probar las funcionalidades implementadas para detectar posibles errores y así mejorar la calidad del producto final.

Durante el desarrollo de la investigación se utilizaron los siguientes **métodos científicos**:

Métodos Teóricos:

Histórico-Lógico: Se utilizó este método para realizar el estudio del estado del arte, analizar la evolución histórica de soluciones similares y las tendencias más recientes del desarrollo de plugins.

Analítico-Sintético: A partir del análisis y síntesis de la teoría referida al objeto de estudio se extrajeron los elementos necesarios para su comprensión.

Métodos Empíricos:

Entrevista: Se aplicó a profesores editores de cursos del EVA para detectar las deficiencias del sistema y sus necesidades a la hora de utilizar como un recurso los objetos de aprendizaje publicados en RHODA. (Ver Anexo 2)

El presente trabajo está estructurado en tres capítulos:

Capítulo 1: Fundamentación Teórica

Se exponen los elementos teóricos que sustentan el problema científico planteado y avalan la propuesta de solución. Se analiza la interoperabilidad de Moodle con diferentes repositorios de recursos educativos. Se describe la metodología, lenguajes y herramientas a utilizar en la implementación del plugin External 2.0.

Capítulo 2: Análisis y Diseño

Se describe la propuesta de solución. Se especifican los requerimientos funcionales y no funcionales del sistema. Se muestran las Historias de Usuario y las Tarjetas CRC con sus respectivas descripciones textuales.

Capítulo 3: Implementación y Prueba

Se describen los elementos necesarios para la implementación a partir del resultado del diseño. Además, se realizan pruebas al sistema para comprobar que el mismo cumple con la especificación funcional realizada y satisface los requerimientos planteados.

CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

Introducción

En este capítulo se realiza un análisis profundo de la plataforma de teleformación Moodle, su arquitectura, los elementos que la componen y su vinculación con repositorios de recursos educativos. Se refiere en el estado del arte, el estudio de los plugin existentes para Moodle. También se caracterizan las tecnologías y estándares más utilizados en el desarrollo de plugins. Por último se detalla la metodología, lenguajes y herramientas a utilizar en la elaboración de la propuesta de solución.

1.1 La plataforma Moodle

El Entorno de Aprendizaje Dinámico Orientado a Objetos y Modular (Moodle) diseñado por Martin Dougiamas en el 2002, es un paquete de software para la creación y administración de cursos y sitios web en Internet. Se distribuye gratuitamente como software libre bajo la licencia pública GNU. Puede funcionar en cualquier ordenador con PHP instalado y soporta varios tipos de bases de datos, esencialmente MySQL y PostgreSQL.

Moodle cuenta con herramientas que facilitan la creación de actividades como: tareas, cuestionarios, encuestas, libros, glosarios, wikis entre otras que enriquecen los procesos de enseñanza-aprendizaje. Para extender sus prestaciones, la plataforma permite la instalación de módulos, bloques y plugins desarrollados por la comunidad, así como importar y emplear objetos de aprendizaje provenientes de diversas fuentes siempre que sigan los estándares SCORM. (Hidalgo, 2009).

Seguidamente se explican conceptos importantes necesarios para el pleno entendimiento del funcionamiento interno de Moodle, tales como elementos de su arquitectura, el *core* y los plugins, así como su vinculación con repositorios de recursos educativos.

Arquitectura de Moodle

Moodle se basa en un núcleo (conocido como el *Moodle core*) del sistema rodeado de una gran cantidad de plugins que proveen funcionalidades específicas a la plataforma.

La estructura de los módulos en Moodle es muy simple, compuesta por un directorio con archivos PHP, JavaScript, XML y/o CSS y puntos de entrada definidos en varios archivos clave como `locallib.php`, `db/install.xml` o `lib.php` que permiten la comunicación con el *core*.

El Core

El *core* de Moodle proporciona todos los mecanismos necesarios para el funcionamiento de 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, sus componentes principales son los siguientes:

- Cursos y Actividades: Moodle define un curso como una secuencia de actividades agrupadas en secciones. Existen actividades como foro, wiki, glosario y diario, que se caracterizan por ser muy flexibles. Los cursos se organizan en categorías jerárquicas dentro de la plataforma.
- Usuarios y Grupos: Cada curso puede tener un número indeterminado de alumnos y un número indeterminado de grupos que acoja a esos alumnos. Los alumnos tienen un perfil de usuario que pueden editar y opciones específicas por cada curso en el que están matriculados. Los cursos cuentan con históricos, roles, permisos, entre otros.
- Matriculación: En Moodle, los tipos de matriculación disponibles en la plataforma se definen a través de módulos, por lo que se definen nuevos tipos de matriculación con sus peculiaridades de forma sencilla a través de módulos externos. Cada tipo de matriculación puede implementar sus propios mecanismos al margen del *core*. (Presso, 2010)

Plugins

Un plugin es una aplicación informática que añade funcionalidades específicas a un programa principal. Su nombre procede del inglés (plugin significa “enchufable”) y su presencia es muy habitual en los navegadores web, en reproductores de música y en sistemas de gestión de contenidos.

Los plugins en Moodle pueden ser de varios tipos: autenticación, bloques, matriculación, temas y actividades. Algunos de los plugins más importantes en Moodle son:

- Actividades y Recursos: Componen los objetos individuales del curso. Existen actividades variadas como: colaborativas, foro, wiki, y cuestionarios. Es el módulo más desarrollado y se instala en el directorio *mod* dentro de la raíz de Moodle
- Bloques: Los bloques son cajas con contenido añadido en columnas a los lados del contenido principal que ofrecen varias funcionalidades. Residen en el directorio *blocks*.
- Matriculación: Este tipo de plugin no es muy común en la plataforma, los mismos residen en el directorio *enrol*.
- Temas: Los estilos se desarrollan dentro de módulos que existen dentro de la plataforma que residen en el directorio *theme*. (Campos, 2011).

Moodle en la UCI

En la Universidad de las Ciencias Informáticas se comenzó a utilizar Moodle a partir del año 2005. La personalización de dicha plataforma para la UCI se conoce como Entorno Virtual de Aprendizaje (EVA). Este constituye un espacio de apoyo al proceso de formación de la carrera de Ingeniería en Ciencias Informáticas, donde los profesores pueden implementar estrategias de enseñanza-aprendizaje en diversas modalidades educativas.

El EVA incorpora funcionalidades a Moodle a través de plugins entre los que se encuentran el bloque para la exportación de contenidos desde la plataforma de teleformación Moodle 2.1.x y el plugin External 1.0 para utilizar los objetos de aprendizaje que se encuentran en RHODA desde Moodle v1.9.12+”. Ambos, antecedentes de la presente investigación.

A continuación se profundiza en conceptos de obligatoria comprensión para el entendimiento de la interoperabilidad de Moodle con RHODA, así como el funcionamiento del plugin External 1.0.

1.2 Conceptualizando

Objetos de Aprendizaje

Un Objeto de Aprendizaje (OA) es una entidad extensa que desarrolla solamente un objetivo, debe contener actividades y una evaluación o comprobación final. Lo anterior puede ser variado de acuerdo a necesidades específicas. En resumen, se trata de un material educativo concreto, con contenidos y actividades de evaluación.

Un objeto de aprendizaje se puede visualizar, de manera sencilla, como un estándar con el cual se elabora material digital para ser utilizado en un curso en línea. Unificar el formato y la forma de los contenidos obedece a varios propósitos, entre ellos, tener a mano un banco de recursos que pueden ser utilizados, mejorados y adaptados a un determinado curso que contenga el tema tratado en el OA. (García, 2005).

De manera general, se puede afirmar que un OA es un archivo digital que desarrolla un determinado contenido y que se encuentra orientado para su uso en una plataforma de aprendizaje en línea; por tanto, su formato de soporte debe ser adecuado para su incorporación en dichos espacios.

Uno de los posibles destinos de un OA es su utilización en un curso dentro del Entorno Virtual de Aprendizaje. Un OA de conjunto con otras actividades como: foros, blogs, talleres y tareas contribuye a enriquecer una unidad temática dentro de un curso, sirviendo a su vez de apoyo al proceso de enseñanza aprendizaje.

Para facilitar la reutilización de los objetos de aprendizaje o recursos educativos se han especificado diferentes estándares que contribuyen a su adecuada descripción. Seguidamente se profundiza al respecto.

Estándar para la descripción de contenidos educativos

Algunos de los estándares más empleados en la reutilización de contenidos son IMS ³Learning Design (IMS LD), Modelo de Referencia para Objetos de Contenido Compartido (SCORM), Dublin Core y Learning Object Metadata (LOM).

La especificación de metadatos adecuados para los materiales educativos es indispensable a fin de añadir valor a los mismos, en el sentido de facilitar su reutilización. Los materiales enriquecidos convenientemente con metadatos se almacenan en bibliotecas digitales de contenidos educativos (por ejemplo, repositorios de objetos de aprendizaje). Estas bibliotecas soportan, consultas significativas que permiten la recuperación de aquellos materiales almacenados que cubren una determinada necesidad pedagógica.

Desde la comunidad de e-Learning se han realizado distintos esfuerzos para estandarizar los esquemas de metadatos que deben ser utilizados en la producción de contenidos educativos. El esfuerzo más prometedor ha desembocado en el estándar IEEE⁴: LOM (Learnig Object Metadata). (Lara, 2007)

Learning Object Metadata

Metadatos de los objetos de aprendizaje (del inglés Learning Object Metadata, LOM) es un modelo de datos para la comunicación de los objetos de contenido (Data Model for Content Object Communication) desarrollado por el Instituto de Ingenieros Eléctricos y Electrónicos (IEEE, por sus siglas en inglés).

LOM establece un esquema de datos conceptual que define la estructura de un registro de metadatos para un OA. La utilización de este esquema permite a los autores de objetos de aprendizaje especificar qué elementos componen una instancia de metadatos, con el propósito de facilitar la búsqueda, evaluación, adquisición y uso de objetos de aprendizaje por parte de los estudiantes, profesores o sistemas automatizados, así como su intercambio y uso compartido. (Frías, 2012)

³ IMS (Instructional Management System Project): organización mundial sin ánimo de lucro que incluye más de 50 miembros y afiliados y cuya misión es desarrollar y promover especificaciones abiertas para las tecnologías del aprendizaje.

⁴ Instituto de Ingenieros Eléctricos y Electrónicos

Este estándar ha ganado prestigio a nivel internacional por su estabilidad, además, es el definido por SCORM para realizar el empaquetamiento de contenidos.

Estándar para el empaquetamiento de contenidos

Empaquetamiento, según Luis A. Álvarez consiste en “etiquetar el contenido de modo que pueda ser reconocido como tal, por el LMS y permita su carga en el sistema”. (Álvarez, 2003). De lo anterior se puede derivar que un estándar de empaquetamiento es un conjunto de normas y reglas que determinan la forma de etiquetar un contenido. Un ejemplo de estos estándares es SCORM 2004 utilizado en la versión 2.0 del bloque C2SCORM de Moodle.

SCORM se puede definir como un conjunto de especificaciones técnicas interrelacionadas construidas teniendo en cuenta el trabajo de otras organizaciones de estándares como son AICC⁵, IMS e IEEE con la finalidad de crear un modelo de contenido unificado. Estas especificaciones facilitan la reutilización de contenidos de aprendizaje a través de múltiples plataformas y productos e-Learning. Establece las pautas para describir, empaquetar, entregar, ejecutar y auditar contenidos por los diferentes sistemas. (González, 2010)

Los objetos de aprendizaje publicados en RHODA pueden ser exportados en formato SCORM 1.2 o SCORM 2004. Por otro lado Moodle posee entre sus actividades una de tipo SCORM que permite visualizar desde la plataforma un OA empaquetado según este estándar.

En el siguiente epígrafe se analiza la integración de Moodle con diferentes repositorios de OA, entre ellos el repositorio RHODA. Se presta especial atención al funcionamiento del plugin External 1.0, antecedente de la presente investigación.

⁵ Autonomous Intelligent Cruise Control o Comité de Entrenamiento Basado en Ordenador de la Industria de la Aviación, desarrolla especificaciones técnicas, dentro de la comunidad de formación de la aviación, para el entrenamiento asistido por ordenador. (Álvarez, 2003)

1.3 Estado del Arte

En el presente epígrafe se analiza la integración de Moodle con diferentes repositorios de OA tanto internacionales como de Cuba.

Internacionalmente

Actualmente en el mundo existen diversos repositorios de objetos de aprendizaje que se han integrado con la plataforma Moodle, a continuación se ejemplifican algunos de ellos:

Repositorio de Objetos de Aprendizaje AMBAR: Este repositorio está actualmente integrado a través de la capa de servicios web con un repositorio de Metadatos, el cual es capaz a través de una interfaz sencilla y usable de soportar el almacenamiento, consulta, uso y reutilización de diversos artefactos generados, así como el manejo de versiones y diferentes niveles de granularidad de los Objetos de Aprendizaje (OA). Este es utilizado en la Universidad Central de Venezuela (UCV). (Ramírez, 2010)

Repositorio DSpace: aplicación de código abierto que ofrece una amplia funcionalidad para gestionar los recursos de la Universidad Nacional de La Plata en Argentina.

MERLOT (Multimedia Educational Resources for Learning an Online Teaching): es un repositorio centralizado que contiene sólo los metadatos y apunta a los objetivos ubicados en sitios remotos. Es independiente y funciona como un portal de OA. (Cartagena, 2010)

EI CAREO (Campus Alberta Repository of Educational Objects): es un repositorio centralizado de objetos de aprendizaje multidisciplinarios de profesores de Alberta (Canadá). Es un repositorio independiente que da acceso a objetos remotos y locales a través de los metadatos contenidos en su colección.

EI ELENA/Edutella es un proyecto europeo que propone mediadores de servicios educativos llamado Smart Spaces, este permite la integración de servicios heterogéneos de aprendizaje como herramientas de tutoría, LMS (Sistema de Gestión de Aprendizaje), sistemas de video conferencia y repositorios. (QUISPE, 2009).

En Cuba

RHODA

El Repositorio de Objetos de Aprendizaje (RHODA) desarrollado en la Universidad de las Ciencias Informáticas es una aplicación web modular multiplataforma, que utiliza el estándar SCORM 1.2 y el SCORM 2004 para gestionar los OA y cuenta con una amplia gama de funcionalidades.

Módulos principales de la versión 2.0 de RHODA:

- Módulo para la creación colaborativa de los OA, donde varias personas podrán estar involucrados en la creación de un mismo OA.
- Módulo para la gestión de los OA con el estándar SCORM 2004. Permitirá tanto la creación/edición como visualización según las pautas de este estándar.
- Módulo para generar reportes estadísticos, donde se podrá escoger con el tipo de gráfica que se quiere generar, además permitirá exportarla a PDF y Excel, así como imprimirlas.
- El sistema cumplirá con las pautas que establece la W3C para la accesibilidad de los discapacitados, garantizando así el acceso para cualquier persona sin importar la discapacidad que tenga.
- Actualización, modificación e incorporación de algunas funcionalidades que tiene la versión 1.0, a raíz de las recomendaciones emitidas por los clientes actuales.
- Actualización de las tecnologías con que se desarrolló el sistema. (Cañizares, 2009).

El repositorio RHODA se intercomunica con Moodle utilizando un plugin External 1.0. A continuación se abordan sus características fundamentales.

Plugin External 1.0

Para la versión 1.9.12+ de Moodle se desarrolló un plugin que enlaza como un recurso los objetos de aprendizaje que se encuentran publicados en el repositorio de la UCI, el mismo cumple con todas las necesidades de los administradores, estudiantes y profesores del EVA, basándose en la integración de la plataforma Moodle con el repositorio RHODA.

Funciones principales de este plugin:

- Configurar los Objetos de Aprendizaje.
- Buscar en el repositorio RHODA.
- Crear la actividad Objeto de Aprendizaje.
- Enlazar los Objetos de Aprendizaje.
- Visualizar los objetos de aprendizaje.

Este plugin no se puede utilizar para versiones de Moodle a partir de la 2.0 debido a los cambios arquitectónicos que se introducen a partir de esta versión de la plataforma.

Conclusiones del estudio realizado

Actualmente en el mundo no se ha desarrollado ningún plugin que permita enlazar como un recurso objetos de aprendizaje publicados en un repositorio. Sin embargo en el Centro FORTES de la Universidad de las Ciencias Informáticas se desarrolló un plugin que permite utilizar desde el EVA los objetos de aprendizaje que se encuentran en el repositorio RHODA.

Los sistemas analizados con anterioridad especialmente el plugin desarrollado para enlazar como un recurso los objetos de aprendizaje que se encuentran en el RHODA de la UCI no se pueden utilizar en la versión 2.3.x de Moodle debido a sus características arquitectónicas. Por tanto, se hace necesaria la construcción de un nuevo plugin. Se propone nombrar al plugin External 2.0.

1.4 Metodología, lenguajes y herramientas a utilizar

En este epígrafe se expone la metodología de desarrollo que rige la implementación del plugin External 2.0. Se analiza el lenguaje de modelado utilizado y los diferentes lenguajes de programación, ya sean del lado del cliente o del lado del servidor. Por último se describen las herramientas empleadas durante el desarrollo.

La selección parte de los resultados de estudios comparativos realizados anteriormente en tesis de grado (Ver anexo 1), resultado del trabajo de investigación de estudiantes y profesores del Centro de Tecnologías para la Formación (FORTES) de la Universidad de las Ciencias Informáticas.

1.4.1 Metodología de desarrollo

Las tendencias actuales del desarrollo proponen la utilización de metodologías ágiles para proyectos cortos, estos buscan la satisfacción del cliente y la entrega temprana de software incremental utilizando métodos informales; un mínimo de productos de trabajo de la ingeniería del software y una simplicidad general del desarrollo.

Para el desarrollo del plugin External 2.0 se decidió utilizar la metodología ágil XP, pues este plugin va a ser desarrollado por una sola persona y en un tiempo reducido. La utilización de dicha metodología permitirá la simplificación del proceso desarrollo. A continuación se detallan algunas características importantes de XP.

Programación Extrema (XP)

La Programación Extrema es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en el desarrollo de software. Este tipo de método se basa en una realimentación continuada entre el cliente y el equipo de desarrollo con una comunicación fluida entre todos los participantes, también busca simplificar las soluciones implementadas y coraje para los múltiples cambios. Este tipo de programación es la adecuada para los proyectos con requisitos imprecisos, muy cambiantes y con un riesgo técnico excesivo. (Canós José H., 2003)

Si bien el ciclo de vida de un proyecto XP es muy dinámico, se puede separar en fases. (Joskowicz, 2008). Estas se describen a continuación.

- Fase de exploración: Es la fase en la que se define el alcance general del proyecto. En esta fase, el cliente define lo que necesita mediante la redacción de sencillas “historias de usuarios”. Los programadores estiman los tiempos de desarrollo en base a esta información. Esta fase dura típicamente un par de semanas, el resultado es una visión general del sistema y un plazo total estimado.

- Fase de planificación: La planificación es una fase corta, en la que el cliente, los gerentes y el grupo de desarrolladores acuerdan el orden en que deberán implementarse las historias de usuario, y asociadas a éstas, las entregas. Típicamente esta fase consiste en una o varias reuniones grupales de planificación.
- Fase de iteraciones: Esta es la fase principal en el ciclo de desarrollo de XP. Las funcionalidades son desarrolladas en esta fase, generando al final de cada una un entregable funcional que implementa las historias de usuario asignadas a la iteración.
- Fase de puesta en producción: Si bien al final de cada iteración se entregan módulos funcionales y sin errores, puede ser deseable por parte del cliente no poner el sistema en producción hasta tanto no se tenga la funcionalidad completa.

Artefactos que genera XP

- Historia de Usuario: Sustituyen a los documentos de especificación funcional y a los casos de uso.
- Plan de Entrega: El cronograma de entregas establece qué historias de usuario serán agrupadas para conformar una entrega, y el orden de las mismas.
- Plan de iteraciones: Las historias de usuarios seleccionadas para cada entrega son desarrolladas y probadas en un ciclo de iteración, de acuerdo al orden preestablecido.
- Tarjetas CRC: Representación de las clases (Clase, Responsabilidades y Colaboraciones).
- Casos de Prueba de Aceptación: Son creadas en base a las Historias de Usuario, en cada ciclo de iteración del desarrollo.

1.4.2 Lenguaje de Modelado

Los lenguajes de modelado son usados en combinación con las metodologías de desarrollo de software para avanzar de una descripción inicial a un plan de implementación y comunicar dicho plan a los

desarrolladores. Con el uso de lenguajes de modelado el ingeniero de software va teniendo una visión del sistema a construir por lo que es de gran ayuda hacer uso de los mismos. (Hernández, 2010).

Lenguaje Unificado de Modelado UML

El Lenguaje Unificado de Modelado (UML, por sus siglas en inglés, Unified Modeling Language). Es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema. Ofrece un estándar para describir un plano del sistema, incluyendo aspectos conceptuales tales como procesos de negocio y funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y componentes reutilizables. UML es un lenguaje de modelado para especificar o para describir métodos o procesos. Se utiliza para definir un sistema y detallar los artefactos en el mismo. (Manage, 2007).

Para el desarrollo de la aplicación se elige utilizar como lenguaje de modelado el UML porque es un estándar que describe las funcionalidades del sistema, aspectos fundamentales a la hora del desarrollo de la herramienta, logrando con eso describir cómo enlazar como un recurso para Moodle los objetos de aprendizaje que se encuentran en RHODA.

1.4.3 Lenguajes empleados para el desarrollo de la aplicación

Las tecnologías que se utilizan en el desarrollo de bloques determinan la eficacia y dinamismo de los mismos. En el sitio oficial de la plataforma Moodle (<http://moodle.org/>) se plantea que los lenguajes a tener en cuenta para el desarrollo de contribuciones para este LMS son: XML, XHTML, PHP, CSS, JavaScript. Es por esto, que en el proyecto Personalizaciones de Moodle se determina la utilización de estas en la solución del problema identificado en la presente investigación.

1.4.3.1 Lenguajes del lado del cliente

Las tecnologías del lado del cliente están orientadas preferentemente, como su nombre indica, para ejecutarse en los puestos cliente, por ejemplo XHTML y JavaScript, entre otros. Esto proporciona las capacidades del cliente que hacen posible la creación de aplicaciones dinámicas de Internet al aprovechar el poder de procesamiento local de las computadoras y los dispositivos. A continuación se mencionarán las tecnologías del lado del cliente a emplear.

XML

Metalenguaje extensible de etiquetas y permite definir la gramática de lenguajes específicos. XML no ha nacido sólo para su aplicación en Internet, sino que se propone como un estándar para el intercambio de información estructurada entre diferentes plataformas. Se puede usar en bases de datos, editores de texto, hojas de cálculo y casi cualquier cosa imaginable. XML, con todas las tecnologías relacionadas, representa una manera distinta de hacer las cosas, más avanzada, cuya principal novedad consiste en permitir compartir los datos con los que se trabaja a todos los niveles, por todas las aplicaciones y soportes. Juega un papel importantísimo en este mundo actual, que tiende a la globalización y la compatibilidad entre los sistemas, ya que es la tecnología que permitirá compartir la información de una manera segura, fiable, fácil. (Desarrolloweb, 2009).

Este lenguaje es utilizado en la plataforma de teleformación, para integrar el repositorio RHODA con Moodle logrando un vínculo entre ambas plataformas. Este lenguaje es utilizado en su versión 1.0.

XHTML

XHTML (lenguaje extensible de marcado de hipertexto), es el lenguaje de marcado pensado para sustituir a HTML como estándar para las páginas web. XHTML es la versión XML de HTML, por lo que tiene, básicamente, las mismas funcionalidades, pero cumple las especificaciones, más estrictas, de XML. (Álvarez, 2012).

Este lenguaje es utilizado en la plataforma Moodle como estándar para las páginas web. La misma plataforma utiliza XHTML en su versión 1.0 y es muy parecida al lenguaje HTML pero esta cumple con las especificaciones más estrictas de XML, pues la misma incorpora una nueva concepción o si lo preferimos, una nueva filosofía de modelación de las páginas web, que busca la creación de una web semántica.

Javascript

Javascript es un lenguaje de programación interpretado, es decir, que no requiere compilación, utilizado principalmente en páginas web, con una sintaxis semejante a la del lenguaje Java y el lenguaje C. Es un lenguaje orientado a objetos, todos los navegadores modernos interpretan el código Javascript integrado dentro de las páginas web. Se puede crear todo tipo de programa que puede ser ejecutado en cualquier

ordenador del mercado: Linux, Windows y Apple. El mismo permite a los desarrolladores crear acciones en sus páginas web, creando diferentes efectos e interactuando con los usuarios. (MaestrosdelWeb, 2007).

Javascript es utilizado por la plataforma de teleformación Moodle para desarrollar diferentes acciones dentro de la misma como manejar objetos, estos facilitan la programación de páginas interactivas y es dinámico, responde a eventos en tiempo real como presionar un botón, pasar el puntero del mouse sobre un determinado texto, entre otros.

CSS

CSS es un lenguaje de hojas de estilos creado para controlar el aspecto de los elementos definidos con HTML y XHTML. CSS es la mejor forma de separar los contenidos y su presentación y es imprescindible para crear páginas web complejas. Este lenguaje permite además, mejorar la accesibilidad del documento, reduce la complejidad de su mantenimiento y permite visualizar el mismo documento en infinidad de dispositivos diferentes. (Pérez, 2008).

Este lenguaje es utilizado para darle estilo a los elementos definidos con XHTML en la plataforma Moodle, cualquier cambio en el estilo marcado para un elemento en la CSS afectaría a todas las páginas vinculadas a este en las que aparezca ese elemento. Para la aplicación se utilizara CSS en su versión 2.0.

1.4.3.2 Lenguajes del lado del servidor

Los lenguajes del lado del servidor proporcionan un entorno rápido de creación de scripts y soporte para los estándares más importantes. Además de las aplicaciones tradicionales de bases de datos, las aplicaciones dinámicas de Internet prometen la integración de las comunicaciones bidireccionales y los datos en tiempo real en las aplicaciones, en este sentido, PHP se perfila como un ejemplo necesario del lado del servidor.

PHP

PHP es un lenguaje "Open Source" interpretado y de alto nivel, especialmente pensado para desarrollo Web y el cual puede ser incluido en páginas HTML. La mayor parte de su sintaxis es similar a C, Java y

Perl y es fácil de aprender. Permite a los creadores de sitios Web, desarrollar páginas dinámicas de una manera rápida y fácil. (TechTerms, 2010).

Por otro lado el código escrito en PHP es independiente de la plataforma, esto quiere decir que una aplicación escrita en este lenguaje puede ser ejecutada en cualquier sistema operativo, presenta un gran número de funciones predefinidas, es de fácil aprendizaje por su similitud con otros lenguajes de programación y se puede integrar de manera sencilla con múltiples bases de datos.

La plataforma de teleformación Moodle utiliza PHP5 como lenguaje del lado del servidor pues este posee una serie de características similares a otros lenguajes, permite aplicar técnicas de programación orientada a objeto y a su vez es muy fácil de aprender.

1.4.4 Herramientas de Desarrollo

A continuación se listarán las herramientas de desarrollo a utilizar en la realización del plugin External 2.0.

1.4.4.1 Herramienta CASE

Las herramientas CASE, 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 pueden ayudar en todos los aspectos del ciclo de vida de desarrollo del software en tareas como el proceso de realizar un diseño del proyecto, cálculo de costes, implementación de parte del código automáticamente con el diseño dado, compilación automática, documentación o detección de errores entre otras. (Sistemas, 2005).

Esta herramienta se utiliza para la realización del diagrama de despliegue ya que el mismo otorga una apreciación visual de la ubicación física de los componentes del plugin y los protocolos de comunicación, logrando así una mayor organización a la hora de implementar el plugin External 2.0.

Visual Paradigm para UML

Es una de las herramientas UML CASE del mercado, fue creada para el ciclo vital completo del desarrollo del software, automatizándolo y acelerándolo, partiendo de la captura de requisitos, análisis,

diseño e implementación. Proporciona características como generación de código, ingeniería inversa y generación de informes. (Paradigm, 2012).

El software de modelado UML ayuda a una más rápida construcción de aplicaciones de calidad, mejores y a un menor coste. Permite dibujar todos los tipos de diagramas de clase, código inverso, generar código desde diagramas y generar documentación. Para la realización de esta aplicación informática se utilizará la versión 8.0 de Visual Paradigm.

Esta herramienta permite aumentar la calidad del software, a través de la mejora de la productividad en el desarrollo y mantenimiento del software. Aumenta el conocimiento informático de una empresa ayudando así a la búsqueda de soluciones para los requisitos. También permite la reutilización del software, portabilidad y estandarización de la documentación, además del uso de las distintas metodologías propias de la ingeniería del software.

Para el desarrollo del plugin External 2.0 se utiliza la herramienta Visual Paradigm para desarrollar el diagrama de despliegue, con el cual se dará paso a la implementación del mismo.

1.4.4.2 Entorno de Desarrollo Integrado (IDE)

Un entorno de desarrollo integrado (en inglés *Integrated Development Environment*) es un programa informático compuesto por un conjunto de herramientas de programación. Un IDE es un entorno de programación que ha sido empaquetado como un programa de aplicación. Consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica. Los IDEs pueden ser aplicaciones por sí solas o pueden ser parte de aplicaciones existentes. Puede dedicarse en exclusiva a un solo lenguaje de programación o bien, poder utilizarse para varios. En esta aplicación se hará uso del IDE NetBeans en su versión 7.2

IDE NetBeans

El IDE NetBeans, es una herramienta para programadores pensada para escribir, compilar, depurar y ejecutar programas. Está escrito en Java, pero puede servir para cualquier otro lenguaje de programación. Existe además un número importante de módulos para extender el IDE NetBeans. Es un producto libre y gratuito sin restricciones de uso y contiene todos los módulos necesarios para el

desarrollo de aplicaciones Java en una sola descarga, permitiéndole al usuario comenzar a trabajar inmediatamente. (NetBeans, 2011).

Este IDE 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 la herramienta ideal para llevar a cabo la implementación de la aplicación en su versión 7.2.

Conclusiones parciales

El entendimiento de conceptos estrechamente relacionados con Moodle y su vinculación a repositorios de recursos educativos como el de objeto de aprendizaje y los estándares internacionales para su empaquetamiento, hace evidente la necesidad del plugin External 2.0.

La metodología, lenguajes, tecnologías y herramientas de programación a utilizar en la implementación del plugin External 2.0 se corresponden con las utilizadas en investigaciones similares realizadas en el Centro FORTES anteriormente.

Se decide utilizar como metodología de desarrollo XP, como lenguajes de programación XML, JavaScript, XHTML y CSS del lado del cliente y del lado del servidor PHP. Como herramienta CASE para el modelado se eligió Visual Paradigm para UML 8.4 y como IDE de desarrollo NetBeans 7.2.

CAPÍTULO 2. DISEÑO DEL PLUGIN EXTERNAL 2.0

Introducción

En este capítulo se realiza una descripción de los procesos y subprocesos vinculados al campo de acción. Se describe de forma general la propuesta del sistema. Se especifican las características y funcionalidades de la aplicación y se definen las relaciones que pueden establecerse entre los usuarios que participarán en la misma. Además se elaboran las Historias de Usuario y las Tarjetas CRC.

2.1 Descripción de los procesos vinculados al campo de acción

Actualmente en la Universidad de las Ciencias Informáticas el enlace de objetos de aprendizaje que se encuentran en el repositorio RHODA con la plataforma Moodle en su versión 1.9.12+ se lleva a cabo utilizando el plugin External 1.0. Seguidamente se describe este proceso en detalle:

Configurar actividad objeto de aprendizaje

Este proceso tiene como objetivo configurar el plugin External para que se pueda conectar al repositorio RHODA y así poder utilizar los objetos de aprendizaje que se encuentran en el mismo. El Administrador es el encargado de realizar la configuración, para que así los estudiantes y profesores puedan acceder desde la plataforma de teleformación a los objetos de aprendizaje que se encuentran en el repositorio RHODA. Como entrada se proveerá el usuario, contraseña y la URL del repositorio al cual se va a conectar. Logrando así una conexión con el repositorio RHODA.

Enlazar el objeto de aprendizaje

Este proceso tiene como objetivo crear la actividad objeto de aprendizaje y buscar los objetos de aprendizaje dentro del repositorio RHODA. El inicio de este proceso depende de la realización del proceso “**Configurar actividad objeto de aprendizaje**”. Los profesores son los encargados de crear dicha actividad y buscar el objeto de aprendizaje que va a utilizar en el curso. Como entrada se proveerá nombre del objeto de aprendizaje y el tipo de recurso a seleccionar. Logrando así una conexión con el repositorio RHODA desde la plataforma Moodle, mostrando una lista con todos los objetos de aprendizaje que aparecen en dicho repositorio.

Visualizar el objeto de aprendizaje

Este proceso tiene como objetivo visualizar en Moodle el objeto de aprendizaje enlazado desde RHODA. El inicio de este proceso depende de la realización de los procesos “**Configurar actividad objeto de aprendizaje**” y “**Enlazar el objeto de aprendizaje**”. Los estudiantes son los encargados de visualizar el objeto de aprendizaje que se encuentre en el curso al que ellos hayan accedido. Como entrada se suministrará una lista con todos los objetos de aprendizaje disponibles en dicho curso. Logrando visualizar el objeto de aprendizaje en la plataforma de teleformación Moodle.

2.2 Propuesta del plugin External 2.0

El plugin External 2.0 permite realizar los procesos antes descritos, además de ser compatible con la versión 2.3.x de Moodle. Mediante una interfaz visual agradable y sencilla permite a los administradores, profesores y estudiantes:

- Configurar un repositorio externo o crear un recurso en Moodle que permita establecer una conexión segura entre el Entorno Virtual de Aprendizaje (EVA) y el repositorio de objetos de aprendizaje RHODA.
- Enlazar el objeto de aprendizaje: Para lograr establecer el enlace primeramente hay que crear la actividad en Moodle y luego seleccionar el objeto de aprendizaje que se vaya a utilizar.
- Instalar y desinstalar el plugin desarrollado dentro de la plataforma de teleformación Moodle.
- Utilizar el objeto de aprendizaje en el curso seleccionado dentro de la plataforma de teleformación. Visualizar el objeto de aprendizaje en el curso.

2.3 Principales particularidades del plugin External 2.0

2.3.1 Funcionalidades del sistema (FS)

Las funcionalidades del sistema son declaraciones de los servicios que debe proporcionar el sistema, de la manera en que éste debe reaccionar a entradas particulares y de cómo se debe comportar en situaciones particulares. Además, estas describen lo que el sistema debe hacer. Estas funcionalidades dependen del tipo de software que se desarrolle, de los posibles usuarios del software y del enfoque general tomado por la organización al redactar requerimientos. Es una descripción de cómo se debe

comportar el sistema, de un atributo o una propiedad. Una condición o capacidad que debe cumplir o poseer un sistema o componente de un sistema.

Las funcionalidades con las cuales debe cumplir la aplicación a desarrollar son:

FS1: Configurar el plugin External 2.0: permite configurar el plugin External 2.0 para que se pueda conectar al repositorio RHODA.

FS2: Enlazar Objetos de Aprendizaje: debe permitir crear la actividad objeto de aprendizaje y realizar una búsqueda de los objetos de aprendizaje que se encuentren publicados en el repositorio RHODA.

FS2.1: Crear la Actividad Objeto de Aprendizaje: El sistema debe permitir crear la actividad objeto de aprendizaje dentro de la plataforma de teleformación Moodle.

FS2.2: Seleccionar objetos de aprendizaje del repositorio RHODA: El sistema debe permitir seleccionar el objeto de aprendizaje que se vaya a utilizar en el curso, dentro de la lista de objetos de aprendizaje que se encuentren publicados en el repositorio RHODA.

FS3: Visualizar Objeto de Aprendizaje: El sistema debe ser capaz de visualizar el objeto de aprendizaje que haya sido seleccionado.

2.3.2 Características del sistema

Las características del sistema no son más que restricciones de los servicios, cualidades o propiedades que el producto debe tener. Son importantes para que clientes y usuarios puedan valorar las características no funcionales del producto, como cuán usable, seguro, conveniente o disponible puedan ser estas.

CS1: Apariencia o interfaz externa:

No debe usarse tecnología de frames (marcos), pues algunos navegadores no lo soportan, otra de las razones es que los frames quitan espacio en la pantalla, este problema se hace más usual en definiciones de pantalla pequeña o dispositivos móviles.

Cada página no debe exceder 100kb en las imágenes.

La interfaz debe ser agradable e intuitiva para conseguir la confianza de los usuarios en la utilización del sistema.

Se deben tener en cuenta algunos elementos del diseño como gráficos de encabezamiento, estilos y formatos de texto establecidos para no afectar el diseño de Moodle.

CS2: Usabilidad

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

CS3: Mantenimiento

Utilizar estándares para el desarrollo de aplicaciones web (XML, CSS 2.0).

CS4: Portabilidad

Al plugin se debe acceder desde cualquier sistema operativo.

Al plugin se debe acceder desde cualquier navegador.

CS5: Seguridad y privacidad

Confidencialidad: La información manejada por el plugin debe estar protegida ante el acceso no autorizado y la divulgación. El acceso al plugin se restringirá a los usuarios de acuerdo al rol que desempeñen. El sistema debe garantizar que datos sensibles no viajen en texto plano por la red.

Integridad: La información manejada por el sistema será objeto de cuidadosa protección contra la corrupción y estados inconsistentes, de la misma forma será considerada igual la fuente o autoridad de los datos.

CS5: Restricciones de diseño e implementación

- El plugin se implementará con tecnología PHP utilizando como entorno de desarrollo al IDE NetBeans 7.2.

2.3.3 Personal relacionado con la aplicación

Se define como persona relacionada al sistema toda aquella que obtiene un resultado del valor de uno o varios procesos que se ejecutan en el mismo. Además de aquellas que se encuentran involucradas en dichos procesos, pues participan en ellos pero no obtienen ningún resultado de valor.

Personal Relacionado	Justificación
Administrador	Es el encargado de dar permisos a los profesores y estudiantes para que puedan acceder a la aplicación.
Profesores	Son aquellas personas que harán uso de la aplicación para enlazar los objetos de aprendizaje.
Estudiantes	Son aquellas personas que visualizarán e interactuarán el objeto de aprendizaje.

Tabla 1: Descripción del personal relacionado con el sistema.

2.4 Planificación

La planificación es una fase de pocos días, en la que el cliente establece la prioridad de cada historia de usuario, y correspondientemente, los programadores realizan una estimación del esfuerzo necesario de cada una de ellas. El resultado de esta fase es un Plan de Entregas.

Típicamente, esta fase consiste en una o varias reuniones grupales de planificación, donde se toman acuerdos sobre el contenido de la primera entrega y se determina un cronograma en conjunto con el cliente. Una entrega debe obtenerse en no más de tres meses.

2.4.1 Historias de usuario.

Las “Historias de usuario” sustituyen a los documentos de especificación funcional, y a los “casos de uso”. Estas “historias” son escritas por el cliente, en su propio lenguaje, como descripciones cortas de lo que el sistema debe realizar. La diferencia más importante entre estas historias y los tradicionales documentos de especificación funcional se encuentra en el nivel de detalle requerido. Las historias de usuario deben tener el detalle mínimo como para que los programadores puedan realizar una estimación poco riesgosa del tiempo que llevará su desarrollo. Cuando llegue el momento de la implementación, los desarrolladores dialogarán directamente con el cliente para obtener todos los detalles necesarios. Las historias de usuario deben poder ser programadas en un tiempo entre una y tres semanas. Si la estimación es superior a tres semanas, debe ser dividida en dos o más historias. Si es menos de una semana, se debe combinar con otra historia. (Joskowicz, 2008).

Historia de Usuario			Número:	1
Nombre: Configurar el Objeto de Aprendizaje				
Prioridad: Alta	Complejidad: Alta	Estimación: 1 Semana	Iteración: 1	
<p>Descripción: Esta historia de usuario permite configurar el plugin External 2.0 para que se pueda conectar al repositorio RHODA. El administrador seleccionará la opción “External” y el sistema mostrará un formulario sencillo con los campos: usuario, contraseña y URL (dirección del repositorio al cual se va a conectar). Una vez completados los campos necesarios el administrador oprime la tecla Guardar Cambios y el sistema muestra una lista con todos los objetos de aprendizaje que se encuentran publicados en RHODA.</p>				
<p>Observaciones:</p> <ul style="list-style-type: none"> ▪ En caso de dejar el campo de texto del usuario esté en blanco, el sistema muestra un mensaje de información: “Por favor escriba su usuario”. ▪ En caso de dejar el campo de texto de la contraseña en blanco, el sistema muestra un mensaje de información: “Por favor escriba una contraseña”. ▪ En caso de dejar el campo de la dirección del URL del repositorio en blanco, el sistema muestra un mensaje de información: “Por favor especifique la dirección URL”. ▪ En caso de que los datos de usuario y contraseña sean incorrectos, el sistema muestra un mensaje de información: “El nombre de usuario y la contraseña no coinciden.” ▪ En caso de que la dirección URL sea incorrecta, el sistema muestra un mensaje de información: “La dirección URL es incorrecta.” 				
Información Adicional: Da cumplimiento al requisito FS1				

Tabla 2: Descripción de la Historia de Usuario: Configurar el Objeto de Aprendizaje

Historia de Usuario			Número:	2
Nombre: Crear actividad Objeto de Aprendizaje				
Prioridad: Alta	Complejidad: Media	Estimación: 1 Semana	Iteración: 1	
<p>Descripción: Esta historia de usuario permite crear la actividad objeto de aprendizaje dentro de la plataforma de teleformación. Para realizar esta acción se crea un formulario sencillo que cuenta con varios campos como nombre, resumen y tipo de recurso (archivo digital como: texto, video, multimedia, pdf, imágenes, audio, páginas web, foto). De estos campos son de obligatorio cumplimiento el nombre y el recurso. Una vez completados los campos</p>				

necesarios el profesor oprime la “Guardar cambios y mostrar” y el sistema muestra la actividad creada dentro del curso.
Observaciones: <ul style="list-style-type: none"> ▪ En caso de dejar el campo de texto del nombre en blanco, el sistema muestra un mensaje de información: “Debe suministrar un valor aquí”. ▪ En caso de dejar el campo de texto del recurso en blanco, el sistema muestra un mensaje de información: “Debe suministrar un valor aquí”. ▪ En caso de que los datos del nombre y el recurso sean incorrectos, el sistema muestra un mensaje de información: “Estos campos son obligatorios.” ▪ En caso del tipo recurso sea incorrecto, el sistema muestra un mensaje de información: “El tipo de recurso es incorrecto.”
Información Adicional: Da cumplimiento al requisito FS2.1

Tabla 3: Descripción de la Historia de Usuario: Crear la actividad Objeto de Aprendizaje.

Historia de Usuario			Número:	3
Nombre: Seleccionar objetos de aprendizaje del repositorio RHODA				
Prioridad: Alta	Complejidad: Alta	Estimación: 1 Semana	Iteración: 1	
Descripción: Permite realizar una búsqueda de los objetos de aprendizaje que se encuentran publicados en el repositorio RHODA desde Moodle. Para realizar esta acción primero se tienen que buscar todos los objetos de aprendizaje que se encuentren publicados en el repositorio. Seguidamente el profesor verifica si el objeto de aprendizaje que va a utilizar en el curso, según su categoría (Matemática, Sistemas Digitales, Formación Científica, Ingeniería y Gestión de Software, entre otros) se encuentra dentro de la lista de objetos de aprendizaje disponibles en el repositorio. Luego de haber realizado el proceso de selección del objeto de aprendizaje, este podrá ser utilizado en el curso por el profesor.				
Observaciones: <ul style="list-style-type: none"> ▪ En caso de que lista de objetos de aprendizaje este vacía, el sistema muestra un mensaje de información: “La lista de objetos de aprendizaje está vacía.” ▪ En caso de que el objeto de aprendizaje que va a utilizar el profesor en el curso no se encuentre en dicha lista, el sistema muestra un mensaje de información: “No se encuentra el objeto de aprendizaje en el 				

repositorio.”

Información Adicional: Da cumplimiento al requisito FS2.2

Tabla 4: Descripción de la Historia de Usuario: Buscar en el repositorio RHODA.

Historia de Usuario			Número:	4
Nombre: Enlazar los objetos de Aprendizaje				
Prioridad: Alta	Complejidad: Alta	Estimación: 2 Semana	Iteración: 2	
<p>Descripción: Esta historia de usuario permite a los profesores enlazar un objeto de aprendizaje disponible en RHODA a un curso en Moodle. Primeramente el profesor crea la actividad objeto de aprendizaje dentro de la plataforma de teleformación. Luego elige la opción “Agregar” .El sistema muestra la lista de objetos de aprendizaje que se encuentran publicados en el repositorio RHODA. El profesor selecciona el objeto de aprendizaje deseado. El sistema establece el enlace entre Moodle y el objeto seleccionado.</p>				
<p>Observaciones:</p> <ul style="list-style-type: none"> ▪ En caso de dejar el campo de texto del nombre en blanco, el sistema muestra un mensaje de información: “Debe suministrar un valor aquí”. ▪ En caso de dejar el campo de texto del recurso en blanco, el sistema muestra un mensaje de información: “Debe suministrar un valor aquí”. ▪ En caso de que los datos del nombre y el recurso sean incorrectos, el sistema muestra un mensaje de información: “Estos campos son obligatorios.” ▪ En caso del tipo recurso sea incorrecto, el sistema muestra un mensaje de información: “El tipo de recurso es incorrecto.” ▪ En caso de que lista de objetos de aprendizaje este vacía, el sistema muestra un mensaje de información: “La lista de objetos de aprendizaje está vacía.” ▪ En caso de que el objeto de aprendizaje que va a utilizar el profesor en el curso no se encuentre en dicha lista, el sistema muestra un mensaje de información: “No se encuentra el objeto de aprendizaje en el repositorio.” 				
Información Adicional: Da cumplimiento al requisito FS2				

Tabla 5: Descripción de la Historia de Usuario: Enlazar los objetos de Aprendizaje.

Historia de Usuario			Número:	5
Nombre: Visualizar el Objeto de Aprendizaje				
Prioridad: Alta	Complejidad: Alta	Estimación: 2 Semana	Iteración: 2	
Descripción: Esta historia de usuario permite visualizar desde un curso en Moodle un objeto de aprendizaje que se encuentre en el RHODA. Para realizar esta acción primeramente te tiene que ver creado la actividad objeto de aprendizaje y haber buscado en el repositorio RHODA el objeto de aprendizaje que se va a utilizar en el curso. El estudiante entra al curso en el cual está matriculado y visualiza este objeto de aprendizaje en la plataforma de teleformación.				
Observaciones:				
<ul style="list-style-type: none"> ▪ En caso de que lista de objetos de aprendizaje este vacía, el sistema muestra un mensaje de información: “La lista de objetos de aprendizaje está vacía.” ▪ En caso de que el objeto de aprendizaje que va a utilizar el profesor en el curso no se encuentre en dicha lista, el sistema muestra un mensaje de información: “No se encuentra el objeto de aprendizaje en el repositorio.” 				
Información Adicional: Da cumplimiento al requisito FS3				

Tabla 6: Descripción de la Historia de Usuario: Visualizar el Objeto de Aprendizaje.

2.4.2 Plan de Entrega

Una vez que el cliente culmina la elaboración de las HU, se comienza con la creación del Plan de Entregas. El mismo se hace con la intención de que los programadores obtengan una estimación de dichas historias en cuanto al nivel de detalle, o sea, para fijar el período de tiempo que se puede tardar en la implementación de cada una.

Historias de Usuario	Iteración 1(17-03-13)	Iteración 2(10-04-13)
Configurar el Objeto de Aprendizaje Crear actividad Objeto de Aprendizaje Buscar en el repositorio RHODA los objetos de aprendizaje	3	Terminado
Enlazar los objetos de Aprendizaje Visualizar el Objeto de Aprendizaje	No empezado	4

Tabla 7: Plan de Entrega.

2.5 Diseño

El diseño del sistema utilizando la Metodología XP sigue rigurosamente el principio MS (mantenlo sencillo). Un diseño sencillo siempre se prefiere sobre una representación más compleja. Además, el diseño guía la implementación de una historia conforme se escribe. Esta metodología estimula el uso de tarjetas CRC como un mecanismo eficaz para pensar en el software en un contexto orientado a objetos. (Joskowicz, 2008).

2.5.1 Tarjetas CRC

Para poder diseñar el sistema como un equipo, se deben cumplir tres requisitos: Cargo o Clase, Responsabilidad y Colaboración (CRC). Las tarjetas CRC permitirán desprenderse del método de trabajo basado en procedimientos y trabajar con una metodología basada en objetos, permiten también que el equipo completo contribuya en la tarea del diseño. Estas registran el nombre de las clases, sus responsabilidades y las otras clases con la que colaboran. Además de la forma de registrar la información de las clases.

Tarjeta CRC :1	
Clase: Roa	
Responsabilidades: Es la clase controladora que se encarga de ser la mediadora entre el repositorio RHODA y la plataforma Moodle. Esta posee una serie de funcionalidades importantes dentro de las que se encuentran <code>get_Session</code> , <code>Sen_file</code> , <code>by_category</code> , <code>send_local</code> , entre otras, encargadas de gestionar los procesos de la aplicación.	Colaboraciones:

Tabla 8: Tarjeta CRC 1.Clase: Roa

Tarjeta CRC :2	
Clase: Tree	
Responsabilidades: Esta clase es la encargada de	Colaboraciones:

pintar las carpetas es decir la jerarquía dentro del plugin en la plataforma Moodle.	
--	--

Tabla 9: Tarjeta CRC 2.Clase: Tree

Tarjeta CRC :3	
Clase: Moodle_Queck_chooserepositoryfile	
Responsabilidades: Esta clase es utilizada para seleccionar un archivo de los que se encuentran en la carpeta curso para luego agregarle el objeto de aprendizaje a visualizar dentro del curso.	Colaboraciones: Moodle_Queck_form_Grup

Tabla 10: Tarjeta CRC 3.Clase: Moodle_Queck_chooserepositoryfile

2.6 Diagrama de Despliegue

La metodología XP plantea que para un mejor entendimiento de las tareas, flujos y métodos de desarrollo de las funcionalidades, se pueden crear diagramas, siempre que su creación no implique mayor esfuerzo que la implementación del mismo. Siguiendo este principio, se elaboró el diagrama de despliegue que permite apreciar de forma visual cómo se encuentran relacionados físicamente los componentes en la aplicación. Un diagrama de despliegue es un modelo de objetos que describe la distribución física del sistema, en términos de cómo se distribuye la funcionalidad entre nodos de cómputo. (Jacobson, 2000).

Para que los usuarios accedan a la plataforma e interactúen con el plugin External 2.0, deben contar con una computadora que se conecte mediante el protocolo HTTP a un servidor web Apache donde se mostrarán la plataforma Moodle y el repositorio RHODA. Este último debe estar conectado a un servidor de Base Datos mediante el protocolo ADO.

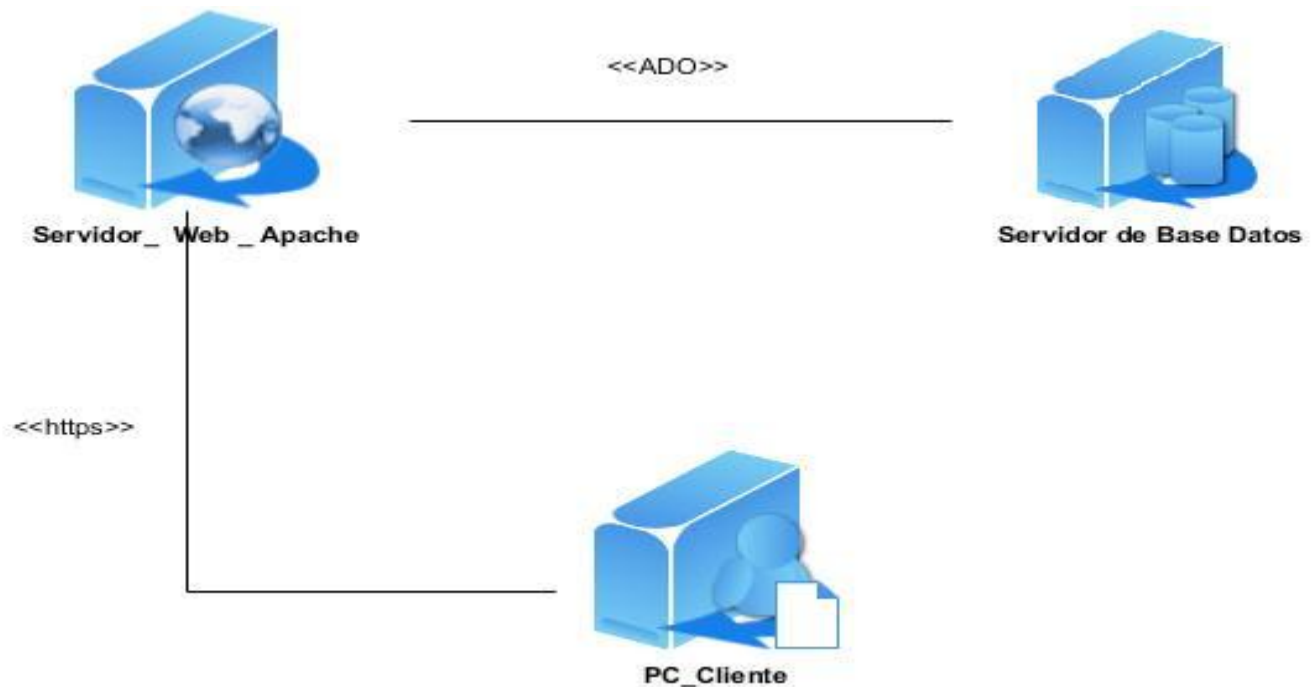


Figura 1: Diagrama de Despliegue.

2.7 Arquitectura del sistema

El diseño de la arquitectura de un sistema, es el proceso por el cual se define una solución para los requisitos técnicos y operacionales del mismo. Este proceso define qué componentes conforman el sistema, cómo se relacionan entre ellos, y cómo mediante su interacción llevan a cabo la funcionalidad especificada.

El diseño arquitectónico define la relación entre los elementos estructurales principales del software, los patrones de diseño que se pueden utilizar para lograr los requisitos que se han definido para el sistema, y las restricciones que afectan a la manera en que se pueden aplicar los patrones de diseño Arquitectónicos (Garlan, 2009).

Los patrones expresan el esquema fundamental de organización para sistemas de software. Proveen un conjunto de subsistemas predefinidos; especifican sus responsabilidades e incluyen reglas y guías para organizar las relaciones entre ellos; así como ayudan a especificar la estructura fundamental de una aplicación (Pressman, 2009).

Se hará uso de una Arquitectura Física de Tres Capas ya que la misma es una arquitectura cliente servidor cuyo objetivo primordial es la separación de la lógica de negocios de la lógica de diseño; un

ejemplo básico de esto consiste en separar la capa de datos de la capa de presentación al usuario. Dicha arquitectura se divide en tres capas que tienen diferentes propósitos e interactúan entre sí, pero de manera independiente. A continuación se describen:

- Capa de presentación: es la que ve el usuario, presenta el sistema al usuario, le comunica la información y captura la información del usuario en un mínimo de proceso (realiza un filtrado previo para comprobar que no hay errores de formato). También es conocida como interfaz gráfica y debe tener la característica de ser "amigable" (entendible y fácil de usar) para el usuario. En esta se encuentra la Interfaz Web de la publicación, que es el medio de comunicación directo entre el sistema y el usuario. Esta capa se comunica únicamente con la capa de negocio.
- Capa de negocio: es donde residen los programas que se ejecutan, se reciben las peticiones del usuario y se envían las respuestas tras el proceso. Se denomina capa de negocio porque es aquí donde se establecen todas las reglas que deben cumplirse. Esta envuelve el software de aplicación usado en la programación de la solución. Esta capa se comunica con la capa de presentación, para recibir las solicitudes y presentar los resultados, y con la capa de datos, para solicitar al gestor de base de datos almacenar o recuperar datos de él.
- Capa de datos: es donde residen los datos y es la encargada de acceder a los mismos. Está formada por uno o más gestores de bases de datos que realizan todo el almacenamiento de datos, reciben solicitudes de almacenamiento o recuperación de información desde la capa de negocio. En esta se encuentra la base de datos usada para el almacenamiento de la información en la plataforma Moodle.

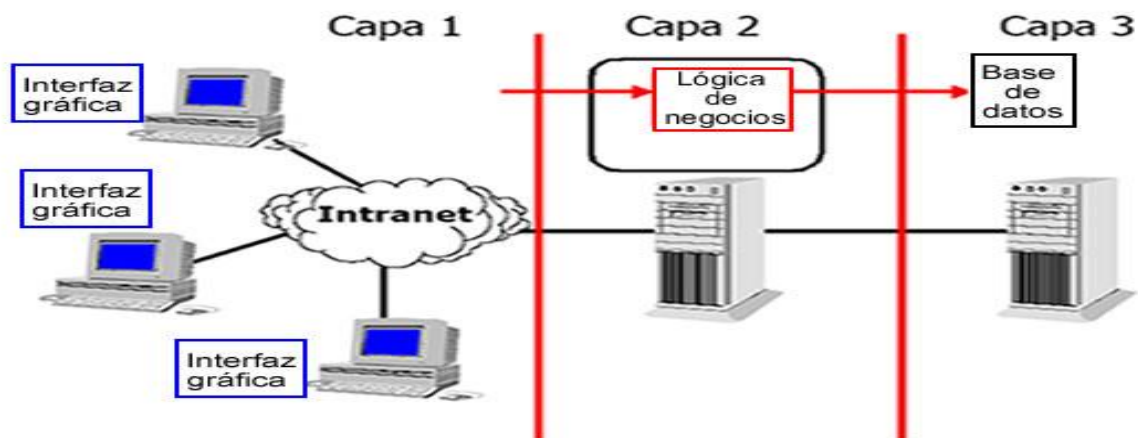


Figura 2: Arquitectura del Sistema.

Conclusiones del Capítulo

A partir de la descripción de los procesos vinculados al campo de acción se confeccionó una propuesta de solución de las necesidades de los administradores del Entorno Virtual de Aprendizaje de la UCI.

La definición y caracterización de las funcionalidades que debe poseer el plugin External 2.0 así como las Historias de usuario descritas guiarán la implementación del sistema.

La elaboración del Plan de entrega y las Tarjetas CRC constituyen artefactos claves para la organización del trabajo, pues estos ordenarán qué implementar en cada instante.

La funcionalidad se implementará haciendo uso de una arquitectura de tres capas, la cual deriva en buenas prácticas de programación y está acorde a las características de la plataforma *Moodle*.

CAPÍTULO 3. IMPLEMENTACIÓN Y PRUEBAS DEL PLUGIN 2.0

Introducción

En el presente capítulo se describe la etapa de implementación de la solución propuesta en el capítulo anterior, generándose el desarrollo de las tareas de ingeniería. Se detallan los casos de prueba o test de aceptación realizados a las historias de usuario correspondientes a cada iteración, demostrando de esta forma el cumplimiento de los requerimientos definidos.

3.1 Estimación de esfuerzos por historias de usuario

Para el buen desarrollo del sistema propuesto, se realizó una estimación para cada una de las historias de usuario identificadas, llegando a los resultados que se muestran a continuación:

Historia de Usuario	Puntos de Estimación
Configurar el Objeto de Aprendizaje	1
Enlazar el Objeto de Aprendizaje	1
Crear la Actividad Objeto de Aprendizaje	1
Seleccionar objetos de aprendizaje del repositorio RHODA	2
Visualizar el Objeto de Aprendizaje	2

Tabla 11: Estimación de esfuerzos por historias de usuario.

3.2 Iteraciones

Esta es la fase principal en el ciclo de desarrollo de XP. Las funcionalidades son desarrolladas en esta fase, generando al final de cada una un entregable funcional que implementa las historias de usuario

asignadas a la iteración. Las iteraciones son también utilizadas para medir el progreso del proyecto. Una iteración terminada sin errores es una medida clara de avance.

Después de ser descritas e identificadas las historias de usuario y estimado el esfuerzo propuesto para la realización de cada una de ellas, se procede a la planificación de la etapa de implementación del sistema. Este plan detalla exactamente cuáles historias de usuario serán implementadas para cada iteración del sistema y las posibles fechas para estas liberaciones. (Joskowicz, 2008).

Para el desarrollo total de la aplicación se definieron dos iteraciones de forma tal que al concluir cada iteración se pudiera obtener un producto funcional que cumpliera con las características deseadas por el cliente y el cual se pudiera someter a las pruebas correspondientes. Para la asignación de los puntos de estimación se tuvo en cuenta que un punto equivale a una semana la cual cuenta con cinco días de trabajo y cada día cuenta con ocho horas laborales.

Primera iteración

En esta iteración se implementan las historias de usuario 1, 2 y 3. Al concluir esta iteración se debe haber implementado todas las funcionalidades propuestas de forma tal que se le pueda mostrar el resultado al cliente mediante un producto funcional.

Tareas por Historia de Usuario definidas en la primera iteración:

Tarea	
Número de la Tarea: 1	Número de HU: 1
Nombre de la tarea: Configurar el Objeto de Aprendizaje	
Tipo de Tarea: Desarrollo	Puntos de Estimación: 1
Fecha de inicio: 17-03-13	Fecha de fin: 24-03-13
Programador Responsable: Daimaris Reyes Curbelo	

Descripción: El administrador podrá configurar el plugin External 2.0 para que se pueda acceder al repositorio RHODA

Tabla 12: Tarea: Configurar el Objeto de Aprendizaje.

Tarea	
Número de la Tarea: 2	Número de HU: 2
Nombre de la tarea: Crear actividad Objeto de Aprendizaje	
Tipo de Tarea: Desarrollo	Puntos de Estimación: 1
Fecha de inicio: 25-03-13	Fecha de fin: 01-04-13
Programador Responsable: Daimaris Reyes Curbelo	
Descripción: El usuario tendrá la posibilidad de crear la actividad objeto de aprendizaje con el fin de que estos puedan utilizar los objetos de aprendizaje publicados en el repositorio RHODA.	

Tabla 13: Tarea: Crear actividad Objeto de Aprendizaje

Tarea	
Número de la Tarea: 3	Número de HU: 3
Nombre de la tarea: Seleccionar objetos de aprendizaje del repositorio RHODA	
Tipo de Tarea: Desarrollo	Puntos de Estimación: 1
Fecha de inicio: 02-04-13	Fecha de fin: 09-04-13
Programador Responsable: Daimaris Reyes Curbelo	

Descripción: El usuario podrá seleccionar objetos de aprendizaje del repositorio RHODA con el objetivo de que puedan utilizar este objeto de aprendizaje dentro de un curso de los que se encuentran en la plataforma de teleformación Moodle.

Tabla 14: Tarea: Seleccionar objetos de aprendizaje del repositorio RHODA

Segunda iteración

En esta iteración se implementarán las historias de usuarios 4 y 5. Al concluir la misma se debe obtener una segunda versión de la aplicación con nuevas funcionalidades listas para mostrar al usuario. También al finalizar esta iteración se contará con el 100% del plugin.

Tarea	
Número de la Tarea: 4	Número de HU: 4
Nombre de la tarea: Enlazar los objetos de Aprendizaje	
Tipo de Tarea: Desarrollo	Puntos de Estimación: 2
Fecha de inicio: 10-04-13	Fecha de fin: 24-04-13
Programador Responsable: Daimaris Reyes Curbelos	
Descripción: Luego de que el usuario haya podido crear la actividad objeto de aprendizaje y haya realizado la búsqueda de los objetos de aprendizaje dentro del repositorio RHODA, podrá este enlazar dicho objeto de aprendizaje dentro de la plataforma de teleformación Moodle.	

Tabla 15: Tarea: Enlazar los objetos de Aprendizaje

Tarea	
Número de la Tarea: 5	Número de HU: 5
Nombre de la tarea: Visualizar el Objeto de Aprendizaje	
Tipo de Tarea: Desarrollo	Puntos de Estimación: 2

Fecha de inicio: 25-04-13	Fecha de fin: 10-05-13
Programador Responsable: Daimaris Reyes Curbelos	
Descripción: El usuario tendrá la posibilidad de visualizar el objeto de aprendizaje dentro de la plataforma de teleformación Moodle.	

Tabla 16: Tarea: Visualizar el Objeto de Aprendizaje

Plan de duración de las iteraciones

Como parte del ciclo de vida de un proyecto utilizando la Metodología XP se crea el plan de duración de cada una de las iteraciones, en este caso se hace para el único equipo de desarrollo con el cual se cuenta. Este plan tiene como finalidad mostrar la duración de cada iteración, así como el orden en que serán implementadas las historias de usuario en cada una de las mismas.

Iteraciones	Orden de las Historias de usuario a implementar	Duración total de las iteraciones
Iteración 1	<ol style="list-style-type: none"> 1. Configurar el Objeto de Aprendizaje 2. Crear actividad Objeto de Aprendizaje 3. Seleccionar objetos de aprendizaje del repositorio RHODA 	3 Semanas
Iteración 2	<ol style="list-style-type: none"> 1. Enlazar los objetos de Aprendizaje 2. Visualizar el Objeto de Aprendizaje 	4 Semanas

Tabla 17: Plan de duración de las iteraciones

3.3 Pruebas

Según William Hetzel en su libro *The Complete Guide to Software Testing* “La prueba de software es una actividad dirigida a la evaluación de un atributo o capacidad de un programa o sistema y la determinación de que cumple con los resultados requeridos”. En palabras de Jiantao Pan en su artículo *Software Testing*

“El propósito de las pruebas puede ser la garantía de calidad, verificación y validación, o estimación de la fiabilidad” [Pan, 1999]. Teniendo en cuenta estas opiniones se puede concluir que las pruebas evalúan el producto garantizando que el mismo cumpla con la calidad y fiabilidad requerida (Beck, 2000).

La metodología XP plantea que las pruebas se deben dividir en dos grupos fundamentales:

- Pruebas unitarias: El propósito de estas pruebas es aislar cada parte del programa y mostrar que las partes individuales son correctas. Las pruebas unitarias son otra forma de comunicación ya que describen el diseño de las clases y los métodos al mostrar ejemplos concretos de cómo utilizar su funcionalidad.
- Pruebas de aceptación: “Las pruebas de aceptación son consideradas como pruebas de caja negra (“Black box system tests”). Los clientes son responsables de verificar Suite de Administración para servidores de entornos productivos: Módulo PackageManager que los resultados de estas pruebas sean correctos. Asimismo, en caso de que fallen varias pruebas, deben indicar el orden de prioridad de resolución”.

Prueba de aceptación

Se considera que las pruebas de aceptación poseen un peso superior a las unitarias, pues las mismas arrojan a vista de los desarrolladores la satisfacción del cliente. Por este motivo se puede decir que estas ponen fin a una iteración dando inicio a la siguiente.

Durante las iteraciones, las Historias de Usuario correspondientes se expresarán en pruebas de aceptación. El cliente desde su punto de vista creará los casos de prueba en conjunto con los desarrolladores para probar que una HU ha sido implementada correctamente. Por cada HU se elaborarán todas las pruebas de aceptación que se necesiten para asegurar su correcto funcionamiento. (XP, 2006).

A continuación se muestran los casos de prueba de aceptación basados en Historias de usuario para el plugin “External 2.0”. La siguiente descripción fue elaborada por la desarrolladora de conjunto con el cliente (administradores del EVA) y llevados a cabo por el cliente.

Caso de Prueba de Aceptación	
Código: HU1_P1	Historia de Usuario: 1
Nombre: Configurar el objeto de aprendizaje	
Descripción: La aplicación permite la conexión de la plataforma Moodle con el repositorio RHODA.	
Condiciones de Ejecución: El módulo permite configurar el objeto de aprendizaje dentro de la plataforma Moodle.	
Entrada/ Pasos de ejecución: Se instalan las herramientas necesarias para poder instalar Moodle, luego se crea el plugin External, después el mismo se conecta al servidor donde se encuentra el repositorio de objeto de aprendizaje introduciendo usuario, contraseña, y la dirección URL de dicho repositorio.	
Resultado Esperado: Se realiza la configuración satisfactoriamente.	
Evaluación de la Prueba: Satisfactoria.	

Tabla 18: Caso de Prueba de Aceptación: Configurar el objeto de aprendizaje

Caso de Prueba de Aceptación	
Código: HU2_P2	Historia de Usuario: 2
Nombre: Crear actividad Objeto de Aprendizaje	
Descripción: La Aplicación permite que se pueda crear la actividad objeto de aprendizaje dentro de la plataforma Moodle	
Condiciones de Ejecución: El módulo permite que se cree la actividad objeto de aprendizaje	

Entrada/ Pasos de ejecución: Se instala la plataforma Moodle, luego se crea el plugin External, después de haber sido configurado el objeto de aprendizaje y se lograra conexión con el repositorio RHODA, entonces se crea la actividad objeto de aprendizaje.
Resultado Esperado: Se realiza la creación de la actividad objeto de aprendizaje satisfactoriamente
Evaluación de la Prueba: Satisfactoria

Tabla 19: Caso de Prueba de Aceptación: Crear actividad Objeto de Aprendizaje

Caso de Prueba de Aceptación	
Código: HU2_P3	Historia de Usuario: 3
Nombre: Seleccionar objetos de aprendizaje del repositorio RHODA	
Descripción: La Aplicación permite que se pueda seleccionar dentro del repositorio RHODA los objetos de aprendizaje	
Condiciones de Ejecución: El moduló permite buscar dentro del repositorio RHODA los objetos de aprendizaje que se utilizarán en Moodle.	
Entrada/ Pasos de ejecución: Se instala la plataforma Moodle, luego se crea el plugin External, después de haber sido configurado el objeto de aprendizaje se logra la conexión con el repositorio RHODA se muestra una lista con todos los objetos de aprendizajes disponibles en el repositorio.	
Resultado Esperado: Se realiza la búsqueda de los objetos de aprendizaje dentro del repositorio RHODA	
Evaluación de la Prueba: Satisfactoria	

Tabla 20: Caso de Prueba de Aceptación: Seleccionar objetos de aprendizaje del repositorio RHODA

Caso de Prueba de Aceptación

Código: HU4_P4	Historia de Usuario: 4
Nombre: Enlazar los Objetos de Aprendizaje	
Descripción: La Aplicación permite que se enlacen los objetos de aprendizaje dentro de la plataforma Moodle	
Condiciones de Ejecución: El moduló permite crear la actividad objeto de aprendizaje dentro de la plataforma Moodle y buscar dentro del repositorio RHODA los objetos de aprendizaje que se utilizarán en dicha plataforma para así lograr el enlace de los objetos de aprendizaje	
Entrada/ Pasos de ejecución: Se instala la plataforma Moodle, luego se crea el plugin External, después de haber sido configurado el objeto de aprendizaje se logra la conexión con el repositorio RHODA, también se muestra una lista con todos los objetos de aprendizajes disponibles en el repositorio y por último se enlaza el objeto de aprendizaje seleccionado.	
Resultado Esperado: Se realiza el enlace de los objetos de aprendizaje satisfactoriamente	
Evaluación de la Prueba: Satisfactoria	

Tabla 21: Caso de Prueba de Aceptación: Enlazar los Objetos de Aprendizaje

Caso de Prueba de Aceptación	
Código: HU5_P5	Historia de Usuario: 5
Nombre: Visualizar el Objeto de Aprendizaje	
Descripción: La Aplicación permite visualizar el objeto de aprendizaje dentro de un curso de los que se encuentran en la plataforma Moodle	
Condiciones de Ejecución: El módulo permite que después de que se haya configurado el objeto de aprendizaje, de que se haya creado la actividad objeto de aprendizaje y que se hayan buscados los mismo dentro del repositorio RHODA se puedan visualizar estos dentro de un curso en la plataforma Moodle	
Entrada/ Pasos de ejecución: Se instala la plataforma Moodle, luego se crea el plugin External, después de	

haber sido configurado el objeto de aprendizaje se logra la conexión con el repositorio RHODA y se muestra una lista con todos los objetos de aprendizajes disponibles en el repositorio, se selecciona el objeto de aprendizaje que se va a visualizar en la plataforma Moodle
Resultado Esperado: Se realiza la visualización del objeto de aprendizaje dentro de la plataforma Moodle satisfactoriamente
Evaluación de la Prueba: Satisfactoria

Tabla 22: Caso de Prueba de Aceptación: Visualizar el Objeto de Aprendizaje

3.4 Resultados y funcionalidades obtenidas

Una evaluación de prueba es una evaluación de los resultados de los esfuerzos de prueba, tales como la cobertura del caso de prueba, la cobertura de código y el estado de los defectos. (Joskowicz, 2008)

Durante el transcurso de las pruebas a la aplicación se detectaron No Conformidades, clasificadas según su importancia en Significativas, No Significativas y de Recomendación. Entiéndase por Significativa aquellas NC que puedan afectar el funcionamiento del módulo, No Significativas las enfocadas en el diseño u otro aspecto que no afecte el funcionamiento de la propuesta de solución y de Recomendación, las sugerencias emitidas por los probadores. A continuación se muestra el resumen de las no conformidades detectadas.

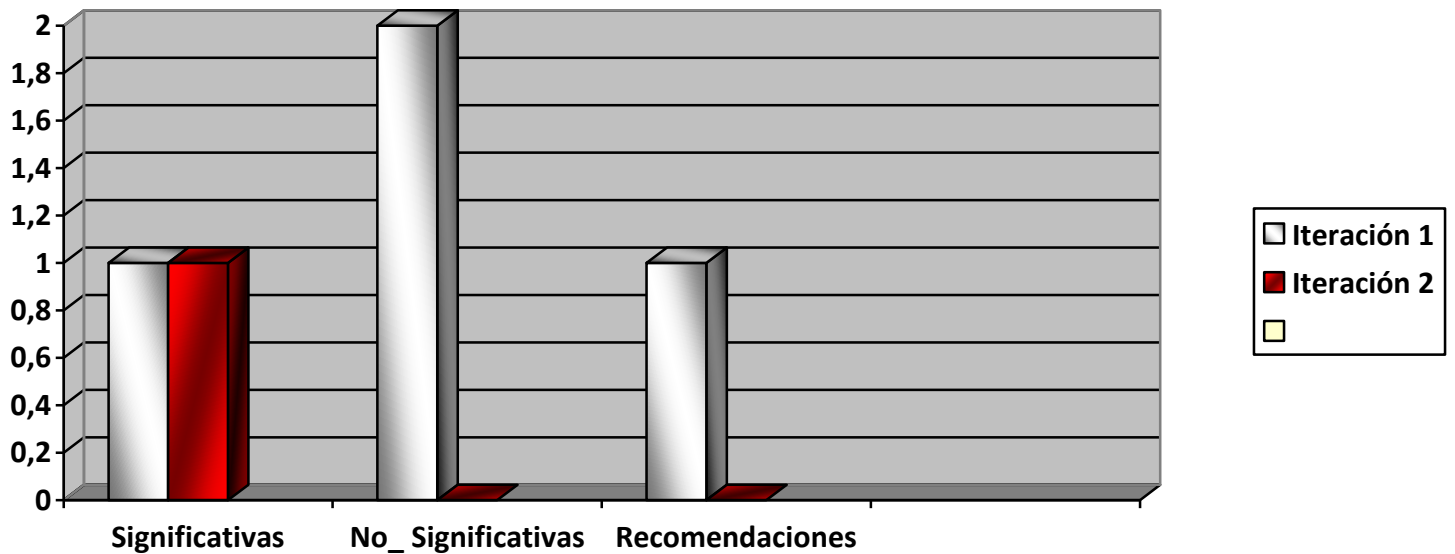


Figura 3: Resumen de las No Conformidades.

Como se evidencia en la gráfica anterior se detectaron 4 NC en la primera iteración y 1 en la segunda, las cuales fueron resueltas favorablemente contribuyendo a un mejor funcionamiento del Plugin External 2.0. (Ver Anexo 4)

Como resultado del trabajo desarrollado se obtuvo el plugin “External 2.0” la cual cumple con todas las especificaciones requeridas e integra funcionalidades que permiten a los administradores, profesores y estudiantes configurar el objeto de aprendizaje, enlazar el objeto de aprendizaje, visualizar el objeto de aprendizaje dentro de la plataforma de teleformación Moodle, logrando la integración de la plataforma Moodle con el repositorio RHODA. El usuario podrá acceder a las funcionalidades de acuerdo a los privilegios de su rol.

3.5 Aporte social y económico

Toda investigación o producto cuyos objetivos estén encaminados a resolver un determinado problema tanto en la economía como en la sociedad, generalmente tiene un impacto. La reciente investigación con su culminación obtuvo una aplicación que enlaza como un recurso los objetos de aprendizaje que se

encuentran publicados en el repositorio RHODA con la plataforma de teleformación Moodle en su versión 2.3.x. El sistema permite una integración entre el repositorio RHODA con la plataforma de teleformación Moodle, logrando que los estudiantes y profesores puedan utilizar los objetos de aprendizaje que se encuentran en dicho repositorio desde el Entorno Virtual de Aprendizaje (EVA). En el aspecto económico contribuye al ahorro del país al no tener que invertir en la implantación y/o actualización de software extranjeros.

Conclusiones del Capítulo

El plugins “External 2.0” se desarrolló teniendo en cuenta las tareas de ingeniería correspondientes para dar solución a las Historias de usuario.

Se obtuvo como resultado un módulo que satisface los requerimientos planteados.

Los resultados obtenidos en las pruebas de aceptación luego de tres iteraciones garantizan la conformidad del cliente con la solución.

Conclusiones

Una vez finalizada la investigación se puede concluir que:

- La no existencia de plugins para lograr la integración de la plataforma de teleformación Moodle v2.3.x con Repositorios de Objetos de Aprendizaje confirmó la necesidad del desarrollo del plugin External 2.0.
- Como metodología de desarrollo de software se decide utilizar XP, asistida por la herramienta CASE Visual Paradigm 8.0, pues es la más adecuada para guiar el proceso de desarrollo de la solución dada su complejidad y alcance. Por las ventajas que brinda NetBeans 7.2 a los desarrolladores y por su condición de software libre se selecciona como IDE para el desarrollo de la propuesta de solución.
- Entre los lenguajes para la propuesta de solución seleccionados se encuentran en primer lugar el Lenguaje Unificado de Modelado (UML), del lado del servidor los lenguajes de programación PHP 5.3.8 debido a que es el lenguaje utilizado en Moodle y del lado del cliente JavaScript, XHTML 1.0 y CCS 2.0.
- La implementación del plugin External 2.0 garantiza la posibilidad de utilizar los OA publicados en el repositorio RHODA desde la plataforma Moodle.
- Los resultados obtenidos durante las pruebas de aceptación realizadas demuestran el cumplimiento de los requerimientos planteados.

Recomendaciones

Incluir el plugin External 2.0 en las personalizaciones que se desarrollen de la plataforma de teleformación Moodle v2.3.x en la UCI.

La ampliación del sistema partiendo de la identificación de nuevas necesidades de los usuarios de la plataforma de teleformación Moodle.

Publicar los resultados de la investigación en la comunidad de Moodle, eventos y revistas científicas.

Bibliografía Referenciada

- ADL, Advanced Distributed Learning. 2006.** Sharable Content Object Reference Model (SCORM®). *ADLNet.org*. [En línea] 2006. [Citado el: 14 de mayo de 2013.] <http://www.adlnet.org>.
- ADL. 2009.** ADL. [En línea] 2009. [Citado el: 10 de enero de 2013.] <http://www.adlnet.gov/Technologies/scorm/SCORMSDocuments/What%20Is%20SCORM.aspx>
- Álvarez, José Valentín Álvarez. 2003.** *Uso de estándares e-learning en espacios educativos*. 2003.
- Álvarez, Miguel Ángel. 2012.** Desarrolloweb: Una breve definición del lenguaje XHTML. [En línea] 2012. [Citado el: 13 de marzo de 2013.] www.desarrolloweb.com/
- Beck, K. 2000.** Extreme Programming Explained. [En línea] 2000. [Citado el: 8 de mayo de 2013.] <http://www.amazon.com/Extreme-Programming-Explained-Embrace-Change/dp/0201616416>
- Blanco, Carlos. 2012.** Vigo:Unión Webmaster Libres: Ide (Entornos de Desarrollo Integrado). [En línea] 21 de abril de 2012. [Citado el: 15 de marzo de 2013.] <http://carlosblanco.pro/2012/04/entornos-desarrollo-integrado-introduccion/>.
- Castro, Miguel. 2008.** *ESTÁNDARES EN E-LEARNING: DE FUTURO INCIPIENTE A REALIDAD*. 2008.
- Campos, Oscar. 2011.** Genbeta Deb: Escribiendo módulos para Moodle: . [En línea] 29 de septiembre de 2011. [Citado el: 10 de febrero de 2013.] <http://www.genbetadev.com/desarrollo-web/escribiendo-modulos-para-moodle-introduccion>.
- 2004.** campusFormación: Soluciones de E-learning. [En línea] Novochip, 2004. [Citado el: 15 de enero de 2013.] <http://www.campusformacion.com/glosario.asp>.
- Cañizares, Roxana. 2009.** Informática 2010: Diseño de un repositorio de objetos de aprendizaje interoperable utilizando. [En línea] 2009. [Citado el: 13 de enero de 2013.] <http://www.informaticahabana.com/?>.
- Canós José H., Letelier Patricio y Penad María del Carmen. 2003.** Google: Metodologías Ágiles en el Desarrollo de Software. [En línea] 2003. [Citado el: 16 de febrero de 2013.] <http://www.willydev.net/descargas/prev/TodoAgil.Pdf>.
- Cartagena, Universidad Politécnica de. 2010.** MERLOT: Repositorios de Objeto Aprendizaje. [En línea] 2010. [Citado el: enero de 25 de 2013.] 2013.
- Desarrolloweb. 2009.** Desarrolloweb: ¿ Qué es XML? [En línea] 2009. [Citado el: 28 de enero de 2013.]

- FORTES. 2010.** Centro de Tecnologías para la Formación. *Centro de Tecnologías para la Formación*. [En línea] UCI, 2010. [Citado el: 1 de diciembre de 2012.] <http://gespro.fortes.prod.uci.cu/>.
- Frías, Boris Luis Correa. 2012.** *Bloque para la exportación de contenidos desde la plataforma de teleformación Moodle 2.1.x*. Habana : s.n., 2012.
- García, Lorenzo. 2005.** Objetos de aprendizaje. [En línea] 10 de febrero de 2005. [Citado el: 2012 de noviembre de 25.] <http://e-spacio.uned.es/fez/eserv.php?pid=bibliuned:329&dsID=editorialfebrero2005.pdf>.
- Garlan, M Shaw y D. 2009.** *Software Architecture*: New York : Prentic-Hall. [En línea] 2009. [Citado el: 5 de mayo de 2013.] <http://stackoverflow.com/questions/704855/software-design-vs-software-architecture>.
- González, Luis A. Álvarez. 2010.** *Empaquetamiento y Visualización de Objetos de Aprendizaje SCORM en LMSs de Código Abierto*. Valdivia. 2010.
- Hernández, Enrique. 2010.** Departamento de Informática de Sistemas y Computadores: Lenguaje de Modelado. [En línea] 2010. [Citado el: 22 de enero de 2013.] <http://www.willydev.net/descargas/prev/ToDoAgil.Pdf>.
- Hidalgo, Jesús. 2009.** Herramientas para la reutilización de contenidos a partir de Moodle. [En línea] junio de 2009. [Citado el: 10 de enero de 2013.]
- Jacobson, Ivar. 2000.** El Proceso Unificado de Desarrollo de Software. [En línea] 2000. [Citado el: 18 de abril de 2013.] <http://bibliodoc.uci.cu/pdf/8478290362.pdf>.
- Jesús Hidalgo Guillén, Yolanda Sardiñas Suárez. 2011.** INCORPORACION DEL ESTANDAR SCORM 2004 AL BLOQUE C2SCORM DE MOODLE. [En línea] 28 de octubre de 2011. [Citado el: 12 de diciembre de 2012.]
- Joskowicz, José. 2008.** Reglas y Prácticas en XP: Nuevas Técnicas de Desarrollo de Software en Ingeniería Telemática. [En línea] 10 de febrero de 2008. [Citado el: 16 de abril de 2013.] <https://www.google.com.cu>
- Lara, Pablo. 2007.** *Revista de Universidad y Sociedad del Conocimiento: Interoperabilidad de los contenidos en las plataformas de e-learning*. 2007.
- MaestrosdelWeb. 2007.** Maestros del web: ¿Qué es Javascript? [En línea] 3 de julio de 2007. [Citado el: 6 de marzo de 2013.] <http://www.maestrosdelweb.com/editorial/%C2%BFque-es-Javascript/>.
- Manage, Free Download. 2007.** Free Download Manager: Paradigma Visual para UML. [En línea] 5 de marzo de 2007. [Citado el: 4 de febrero de 2013.] <http://www.freedownloadmanager.org/es/downloads/contact.php>.

Mi Portal. [En línea] Ministerio de Educación, 2011 de mayo de 5. [Citado el: 10 de noviembre de 2012.] http://www.miportal.edu.sv/Home/Tecnologia_en_el_Aula/sistemas+elearning.htm.

Moodle. 2012. Moodle-Docs. *moodle.org*. [En línea] 6 de mayo de 2012. [Citado el: 11 de junio de 2013.] <http://docs.moodle.org>.

Moodle.org. [En línea] 4 de agosto de 2007. [Citado el: 18 de noviembre de 2012.] http://docs.moodle.org/es/Acerca_de_Moodle.

Navarro, Susana Martínez. 2006. Plan de acciones para la convergencia europea (PACE): “Los objetos de aprendizaje como recurso para la docencia universitaria: criterios para su elaboración. [En línea] 2006. [Citado el: 15 de marzo de 2013.] http://www.upv.es/vece/central_pace.htm.

Negrín, Miguel. 2011. *INCORPORACIÓN DEL ESTÁNDAR SCORM 2004 AL BLOQUE C2SCORM PARA LA EXPORTACIÓN DE CURSOS A PAQUETES*. Habana : s.n., 2011.

NetBeans. 2011. NetBeans IDE. org : IDE de Desarrollo. [En línea] 2011. [Citado el: 6 de abril de 2013.] <https://netbeans.org/features/index.html>.

Paradigm, Visual. 2012. Visual Paradigm : Visual Paradigm for UML. [En línea] 2012. [Citado el: 17 de marzo de 2013.] <http://www.visual-paradigm.com>.

Pérez, Javier Eguíluz. 2008. librosweb.es: Introducción a CSS. [En línea] 2008. [Citado el: 10 de abril de 2013.] <http://www.librosweb.es/css/index.html>.

Pressman, Roger S. 2009. Intercambios Virtuales: Ingeniería del Software: Un Enfoque Práctico. [En línea] 2009. [Citado el: 2 de mayo de 2013.] <http://www.intercambiosvirtuales.org/libros-manuales/ingenieria-del-software-un-enfoque-practico-roger-pressman-sexta-edicion>.

Presso, María Isabel. 2010. Aprendizaje en red comunidad moodle-presso: Arquitectura de Moodle. [En línea] Agosto de 2010. [Citado el: 6 de diciembre de 2012.] <http://www.slideshare.net/mariapresso66/aprendizaje-en-red-comunidad-moodlepresso#btnNext>.

QUISPE, Celia Rosa Cristobal. 2009. Comisión Sectorial De Enseñanza: Los Repositorios de Objetos de Aprendizaje como soporte a un entorno e-learning. [En línea] 2009. [Citado el: 12 de febrero de 2013.] <http://data.cse.edu.uy/node/34>.

Ramírez, Elizabeth. 2010. Integración del Repositorio de Objetos de Aprendizaje de Ambar con la plataforma Moodle. [En línea] 2010. [Citado el: 3 de junio de 2013.] <http://www.slideshare.net/vcmiguel/integracin-del-repositorio-de-objetos-de-aprendizaje-de-ambar-con-la-plataforma-moodle>.

Sevilla, Universidad de. 2007. Centro de Formación Permanente. [En línea] 2007. [Citado el: 8 de diciembre de 2012.] <http://www.cfp.us.es/web/contenido.asp?id=3417>.

Sistemas, Centro de Estudios y Diseño de. 2005. CEDS e- Learnind: Introducción a los Sistemas y Herramientas CASE. [En línea] Registro Gral. de la Propiedad Intelectual de Madrid, 2005. [Citado el: 12 de abril de 2013.] <http://ceds.nauta.es/informes/case01.htm>.

TechTerms. 2010. TechTerms.com: PHP. [En línea] octubre de 2010. [Citado el: 20 de marzo de 2013.] <http://www.techterms.com/definition/php>.

XP. 2006. A gentle introduction. [En línea] 2006. [Citado el: 8 de mayo de 2013.] <http://www.extremeprogramming.org>.

Zapata, Miguel. 2005. Secuenciación de Contenidos y Objetos de Aprendizaje. [En línea] 15 de febrero de 2005. [Citado el: 14 de diciembre de 2012.] <http://redalyc.uaemex.mx/redalyc/src/inicio/ArtPdfRed.jsp?iCve=54709510> .

Bibliografía Consultada

1. Mi Portal. [En línea] 2011. [Citado el: 10 de noviembre de 2012.] http://www.miportal.edu.sv/Home/Tecnologia_en_el_Aula/sistemas+elearning.htm.
2. Moodle. org. [En línea] 4 de agosto de 2007. [Citado el: 18 de noviembre de 2012.] http://docs.moodle.org/es/Acerca_de_Moodle.
3. García, L. (2005). Objetos de aprendizaje. [Citado el: 25 de noviembre de 2012.] Recuperado de <http://e-spacio.uned.es/fez/eserv.php?pid=bibliuned:329&dsID=editorialfebrero2005.pdf>
3. FORTES. 2010. Centro de Tecnologías para la Formación. [En línea] 2010. [Citado: 1 de diciembre, 2012.] <http://gespro.fortes.prod.uci.cu/>.
4. Univ.de Sevilla, Grupo de trabajo de CFP de la. CFP. [En línea] 2007. [Citado: 8 de diciembre, 2012.] <http://www.cfp.us.es/web/contenido.asp?id=3417>.
5. Negrín León, Julio Cesar y Rojas Fonseca, Naydeline. INCORPORACIÓN DEL ESTÁNDAR SCORM 2004 AL BLOQUE C2SCORM PARA LA EXPORTACIÓN DE CURSOS A PAQUETES. 2011. [Citado: 12 de diciembre, 2012.]
6. García, L. (2005). Objetos de aprendizaje. [Citado: 14 de diciembre, 2012.] Recuperado de <http://e-spacio.uned.es/fez/eserv.php?pid=bibliuned:329&dsID=editorialfebrero2005.pdf>
7. Zapata, M. (2005). Secuenciación de contenidos y objetos de aprendizaje. [Citado: 6 de enero, 2013.] Recuperado de <http://redalyc.uaemex.mx/redalyc/src/inicio/ArtPdfRed.jsp?iCve=54709510> .
8. Martínez, F., Prendes, M. (2007). Matemática en red. Los objetos de aprendizaje en sistemas presenciales de enseñanza secundaria. Recuperado de http://digitum.um.es/xmlui/bitstream/10201/12636/1/Proyecto_OA.pdf
9. Cañizares, R. y otros, 2009: Diseño de un repositorio de objetos de aprendizaje interoperable utilizando estándares abiertos. [Citado: 9 de enero, 2013.] Disponible en <http://www.informaticahabana.com/?q=es/trabajo&trid=1098>
10. ADL. (2009). "What Is SCORM." [Citado: 10 de enero, 2013.] Desde <http://www.adlnet.gov/Technologies/scorm/SCORMSDocuments/What%20Is%20SCORM.aspx>.
11. Hidalgo G, J. Gutiérrez G, Y. Herramientas para la reutilización de contenidos a partir de Moodle. Universidad de las Ciencias Informáticas, 2008. Tesis de Grado. [Citado: 15 de enero, 2013.]
12. campusFormación. com. [En línea] <http://www.campusformacion.com/glosario.asp>. 2004. [Citado el: 15 de enero de 2013.]

13. Technology Evaluation Centers. [En línea] 2008. [Citado el: 20 de enero de 2013.] <http://lms2.technologyevaluation.com/es/>.
14. 2011 Universidad España <http://noticias.universia.es/ciencia-ntt/reportaie/2007/04/19/651176/5/revolucion-pedagogica-entorno-moodle/caracteristicas-de-moodle.html>
15. Archivo .pdf de la Universidad Politécnica de Valencia con una amplia explicación sobre metodologías ágiles y la programación extrema (XP). <http://www.willydev.net/descargas/prev/ToDoAgil.pdf>
16. Desarrolloweb.com. 2009 [Online] [Citado el: 28 de enero de 2013.]¿Qué es XML? <http://www.desarrolloweb.com/articulos/449.php>.
17. Maestros del web. [Online] Julio 3, 2007. <http://www.maestrosdelweb.com/editorial/%C2%BFque-es-Javascript/>.
18. Dondo, Agustín. PHP en castellano. [Online] <http://www.programacion.net/php/articulo/porquephp/#porquephp> contras.
19. "NetBeans - EcuRed", s.f., <http://www.ecured.cu/index.php/NetBeans>. [Citado el: 4 de febrero de 2013.]
20. **Joskowicz José, 2008.** Nuevas Técnicas de Desarrollo de Software en Ingeniería Telemática", del Doctorado de Ingeniería Telemática de la Universidad de Vigo, España. Reglas y Prácticas en eXtreme Programming. pdf
21. Genbeta Deb: Escribiendo módulos para Moodle: Introducción. Citado: 10 de febrero de 2012 | 01:23 CET <http://www.genbetadev.com/desarrollo-web/escribiendo-modulos-para-moodle-introduccion>
22. Canós, José H., Letelier, Patricio y Penad, María del Carmen. Google. Metodologías Ágiles en el Desarrollo de Software. [En línea] Universidad Politécnica de Valencia. [Citado el: 16 de febrero de 2013.] <http://www.willydev.net/descargas/prev/ToDoAgil.Pdf>.
23. Enrique Hernández Orallo (ehernandez@disca.upv.es): Lenguaje de Modelado. [En línea] www.disca.upv.es/enheror/pdf/ActaUML.PDF
24. TechTerms.com. TechTerm.com. PHP. [En línea] octubre de 2012. [Citado el: 25 de febrero de 2012.] <http://www.techterms.com/definition/php>.
25. Miguel Ángel Álvarez: Una breve definición del lenguaje XHTML. [Citado el: 15 marzo de 2013.] www.desarrolloweb.com › Manuales › Qué es cada tecnología.
26. Pérez, Javier Eguíluz. 2009. librosweb.es. Introducción a CSS. [En línea] 2008. [Citado el: 10 de abril de 2013.] <http://www.librosweb.es/css/index.html>

27. Pan, Jiantao. Electrical & Computer Engineering. [En línea] 1999. [Citado el: 25 de abril de 2013.] Universidad Carnegie Mellon. http://www.ece.cmu.edu/~koopman/des_s99/sw_testing/#reference.
28. Jacobson, Ivar, Booch, Grady y Rumbaugh, James. El Proceso Unificado de Desarrollo de Software. [En línea] 2000. [Citado el: 2 de mayo de 2013.] <http://bibliodoc.uci.cu/pdf/8478290362.pdf>.
29. Shaw, M. y Garlan, D. "Software Architecture". New York : Prentic-Hall, 1996. [Citado el: 2 de mayo de 2013.]
30. Pressman, Roger S. "Ingeniería de Software. Un enfoque práctico." 5ta Edición (traducción de la edición original en Inglés "Software Engineering. A practical approach"). Madrid : Mac Graw Hill, 2001. [Citado el: 10 de mayo de 2013.]
31. **Cartagena, Universidad Politécnica de. 2010.** MERLOT: Repositorios de Objeto Aprendizaje. [En línea] 2010. [Citado el: enero de 25 de 2013.] 2013.
32. **TechTerms. 2010.** TechTerms.com: PHP. [En línea] octubre de 2010. [Citado el: 20 de marzo de 2013.] <http://www.techterms.com/definition/php>.
33. **XP. 2006.** A gentle introduction. [En línea] 2006. [Citado el: 8 de mayo de 2013.] <http://www.extremeprogramming.org>.
34. **Sistemas, Centro de Estudios y Diseño de. 2005.** CEDS e- Learnind: Introducción a los Sistemas y Herramientas CASE. [En línea] Registro Gral. de la Propiedad Intelectual de Madrid, 2005. [Citado el: 12 de abril de 2013.] <http://ceds.nauta.es/informes/case01.htm>
35. **QUISPE, Celia Rosa Cristobal. 2009.** Comisión Sectorial De Enseñanza: Los Repositorios de Objetos de Aprendizaje como soporte a un entorno e-learning. [En línea] 2009. [Citado el: 12 de febrero de 2013.] <http://data.cse.edu.uy/node/34>.
36. **Paradigm, Visual. 2012.** Visual Paradigm : Visual Paradigm for UML. [En línea] 2012. [Citado el: 17 de marzo de 2013.] <http://www.visual-paradigm.com>.
37. **Manage, Free Download. 2007.** Free Download Manager: Paradigma Visual para UML. [En línea] 5 de marzo de 2007. [Citado el: 4 de febrero de 2013.] <http://www.freedownloadmanager.org/es/downloads/contact.php>.
38. **Beck, K. 2000.** Extreme Programming Explained. [En línea] 2000. [Citado el: 8 de mayo de 2013.] <http://www.amazon.com/Extreme-Programming-Explained-Embrace-Change/dp/0201616416>.
39. Crispin, L. y House, T. Testing Extreme Programming. s.l. : Addison Wesley, 2002.

40. Guía Breve de Tecnologías XML. [En línea] [Citado el: 19 de febrero de 2013.] <http://www.w3c.es/divulgacion/guiasbreves/tecnologiasXML>.
41. Programación en castellano: Montar una web con apache+php+postgresql. [En línea] [Citado el: 28 de marzo de 2013.] <http://www.programacion.net/comentarios/id=apache&obj=escrito&num=5/>.
42. Wells, Don. Extreme Programming. [En línea] 2009. [Citado el: 5 de Mayo de 2013.] <http://www.extremeprogramming.org/>.
43. Joskowicz, Jose. Instituto de Ingeniería Eléctrica. [En línea] 10 de 2 de 2008. [Citado el: 7 de mayo de 2013.] <http://iee.fing.edu.uy/~josej/docs/XP%20-%20Jose%20Joskowicz.pdf>.
44. Solís., Manuel Calero. Una explicación de la programación extrema (XP). Madrid: s.n., 2003.

Glosario de Términos

E-learning: La modalidad educativa e-learning o aprendizaje electrónico es una revolucionaria forma de capacitación que posibilitó internet, posicionándose como la forma de capacitación predominante del futuro.

LMS: Sistemas Gestores del Aprendizaje, también conocido como plataforma de aprendizaje es un software que automatiza la administración de acciones de formación: gestión de usuarios, gestión y control de cursos, gestión de los servicios de comunicación, etc.

Moodle: Entorno de Aprendizaje Dinámico Orientado a Objetos y Modular; paquete de software para la creación de cursos y sitios web, que permite el diseño de cursos estructurados incluyendo recursos y actividades, y llevar un seguimiento detallado de las acciones del educando y sus avances, cuestiones necesarias para dirigir un buen proceso de aprendizaje en línea.

Estándar: Son normas que se utilizan como punto de partida para el desarrollo de servicios, aplicaciones, protocolos. En el campo técnico la estandarización es el proceso por el cual se establecen normas comúnmente aceptadas que permiten la cooperación de diferentes factores.

Plugins: Es una aplicación adicional que se relaciona con otra para aportarle una nueva función y generalmente muy específica. Esta aplicación es ejecutada por la aplicación principal e interactúan por medio de la API.

XML: Lenguaje de Marcas Extensible (eXtensible Markup Language) es un lenguaje de marcas desarrollado por el World Wide Web Consortium(W3C).

CSS: es el lenguaje usado para definir la presentación de un documento estructurado escrito en HTML o XML, separando la estructura del documento de su presentación.

Javascript: lenguaje de programación interpretado (no requiere compilación), usado principalmente para páginas web.

Anexos

Anexo 1: Punto de partida para la selección de la metodología, lenguajes y herramientas.

Investigaciones donde se realiza un estudio comparativo entre diferentes metodologías de desarrollo, lenguajes y herramientas y se seleccionan las más adecuadas para el desarrollo de plugins, funcionalidades y módulos para el LMS Moodle:

Investigación 1

- **Título:** Herramienta Web para la Autonomía en el Aprendizaje y la Consulta de contenidos de Informática.
- **Autores:** Deiler Sevilla Fernández y Yanirys Silva Lemus
- **Año:** 2008
- **Descripción:** Implementación de un sistema con autonomía total sobre la planificación para el estudio de contenidos informáticos.
- **Criterio de referencia:** Esta investigación se estudió para llegar a la conclusión de que metodología de desarrollo utilizar para la confección y desarrollo del plugin.
- **Referencia:** http://repositorio_institucional.uci.cu/jspui/handle/ident/TD_1136_08

Investigación 2

- **Título:** Portal de Software Libre 2.0
-
- **Autores:** Karel Rodríguez Varona y Yordanis Cabreja Nuñez
- **Año:** 2007
- **Descripción:** Investigación llevada a cabo para la selección de las herramientas y metodología para el desarrollo del Portal de Software Libre 2.0. Contiene un estudio de las aplicaciones más populares y útiles que dan soporte a las comunidades virtuales en sus sitios oficiales y de las

tecnologías más efectivas y de uso actual para el desarrollo de portales. Fue realizado con el objetivo de dotar al nuevo portal con las últimas y más novedosas funcionalidades que pudieran utilizarse en el entorno universitario. Además se tuvo en cuenta la utilización de una metodología de desarrollo de software que permitiera la rápida puesta en marcha del proyecto y que este desarrollo estuviera estrechamente vinculado al desarrollo de software libre.

- **Criterio de referencia:** Esta investigación se estudió para llegar a la conclusión de que metodología de desarrollo utilizar para la confección y desarrollo del plugin.
- **Referencia:** http://repositorio_institucional.uci.cu/jspui/handle/ident/TD_0858_07

Investigación 3

- **Título:** Bloque para la exportación de contenidos desde la plataforma de teleformación Moodle 2.1.x
- **Autores:** Linet Katuska Remón Salcedo y Boris Luis Correa Frías
- **Año:** 2012
- **Descripción:** Se desarrolla una extensión para la plataforma Moodle 2.1.x que permite la exportación de los contenidos de sus cursos facilitando así la reutilización de estos en otros entornos educativos. Se emplean los métodos de investigación científica histórico-lógico, analítico-sintético y modelación. Como resultado de la investigación se obtuvo C2SCORM 3.0, un bloque para la exportación de contenidos de los cursos de la plataforma Moodle 2.1.x en forma de paquete SCORM 2004, que facilitó su reutilización en otros entornos compatibles con el estándar.
- **Criterio de referencia:** Esta investigación se estudió para llegar a la conclusión de que lenguajes, y herramientas de desarrollo utiliza la plataforma de teleformación Moodle.
- **Referencia:**
 - http://repositorio_institucional.uci.cu/jspui/handle/ident/TD_05589_12

Anexo 2: Entrevista realizada a profesores editores

Objetivo: Detectar la necesidad de incluir funcionalidades en la plataforma de teleformación Moodle que permitan la integración de la misma con repositorios de Objetos de Aprendizaje.

Usted ha sido seleccionado por sus conocimientos formativos, por sus años de experiencia en el trabajo con la plataforma y los resultados alcanzados en su labor profesional; como profesor editor a evaluar las necesidades prioritarias en cuanto a la utilización de objetos de aprendizaje en la plataforma de teleformación Moodle. Se le agradece de antemano su cooperación.

Temática que se investiga: Desarrollo del plugin External 2.0 para enlazar los objetos de aprendizaje del repositorio RHODA con Moodle v2.3.x

Nombre y Apellidos: Javier Soler Martin

Rol que desempeña:

La entrevista a profundidad estuvo guiada por las siguientes preguntas:

- ¿Cómo usted utiliza en la versión 2.3.X de Moodle un objeto de aprendizaje?

Puedo utilizar el módulo de SCORM pero es solo para archivos locales y es muy arcaica.

- ¿Por qué se pueden utilizar en la plataforma Moodle en su versión 2.3.x los objetos de aprendizaje?

Porque no tiene en cuenta el estándar SCORM y además es solo para algunos repositorios internacionales

- ¿Cree pertinente el tema de investigación que se propone desarrollar?.

Sí, es muy pertinente por dar otra óptica a la integración de los repositorios para obtener OA como actividades de un curso.

Anexo 3: Interfaces de la sistema

▼ Plugins

- Vista general de plugins
- ▼ Módulos de actividad
 - Gestionar actividades
 - Tarea
 - Libro
 - Chat
 - Base de datos
 - Externo**
 - Carpeta (folder)
 - Foro
 - Glosario
 - Paquete contenido IMS
 - Lección
 - Herramienta externa
 - Página
 - Examen

Externo

Repositorio Valor por defecto: URL del repositorio
external_url

Usuario Valor por defecto: Usuario que puede utilizar para entrar a RHONA
external_user

Contraseña Desenmascarar
external_password

Colección Valor por defecto: Colección que puede usar
external_collection

Figura 4: Interfaz_ Configuración del Plugin Externo 2.0.

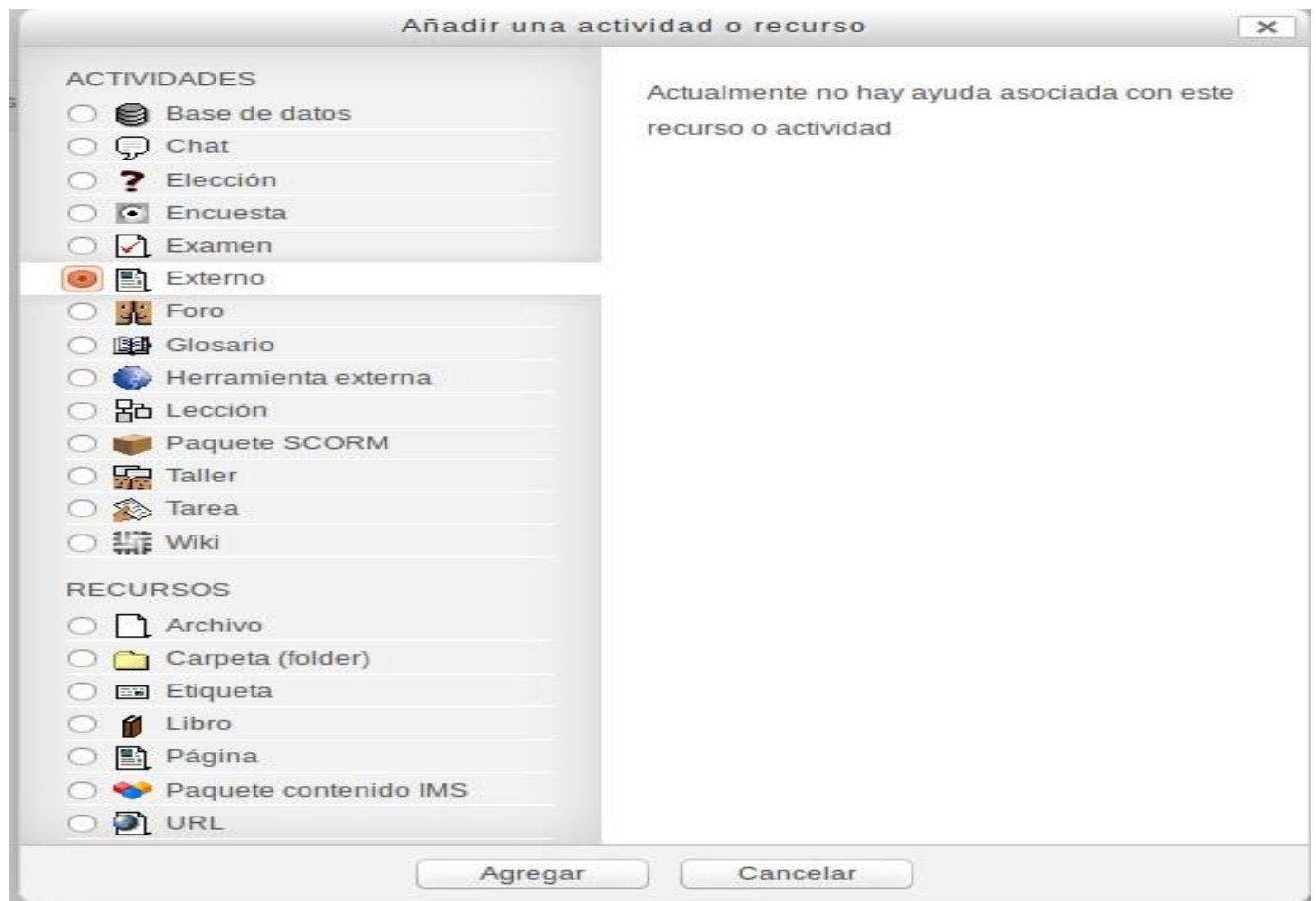


Figura 5: Interfaz_ Añadiendo la actividad Objeto de Aprendizaje

Agregando Externo

General

Nombre*

Resumen ?

Familia Font Tamaño letra Párrafo

B *I* U ABC X₁ X₂ [Listas] [Insertar] [Borrar] [Color] [Fondo] [HTML] [ABC]

Ruta: p

Tipo de recurso

Archivo

Dirección*

Configuraciones comunes del módulo

Modo de grupo ?

Visible

Número ID ?

En este formato hay campos obligatorios*

Figura 6: Interfaz_ Crear actividad Objeto de Aprendizaje

P » Archivos
Resultado de la Búsqueda en el repositorio

« 1 »














- Título:Cadena de suministro on-line.
Descripción : El Objeto de Aprendizaje contiene una animación de cómo se lleva a cabo el proceso de gestión de la cadena de suministro on-line.
Categoría/Subcategoría : Ciencias Empresariales
- Título:Tormenta de Ideas
Descripción : Ejemplo ilustrativo de un equipo de trabajo tomando decisiones a partir de la técnica Tormenta de Ideas.
Categoría/Subcategoría : Ciencias Empresariales
- Título:Papel de los graduados de la UCI en las empresas e instituciones del país.
Descripción : Se presenta un intercambio y debate realizado en un panel con directivos de las empresas Desoft, Softel, y Procyon del Ministerio de la Informática y las Comunicaciones, con la participación de profesores del Departamento Docente Cen
Categoría/Subcategoría : Ciencias Empresariales
- Título:Conozca la Empresa Desoft
Descripción : Desoft es una empresa que pertenece al Ministerio de la Informática y las Comunicaciones y en este recurso didáctico se da a conocer información y datos relativos al funcionamiento de esta empresa cubana.
Categoría/Subcategoría : Ciencias Empresariales
- Título:Conozca la Empresa Softel
Descripción : Softel es una empresa que pertenece al Ministerio de la Informática y las Comunicaciones y en este recurso didáctico se da a conocer información y datos relativos al funcionamiento de esta empresa cubana.
Categoría/Subcategoría : Ciencias Empresariales

[Descargar](#) [Salir](#)

Figura 7: Interfaz_ Buscar el OA a utilizar

[Página Principal \(home\)](#) ▶ [Mis cursos](#) ▶ [P](#)

P » Archivos

	Nombre	Tamaño	Modificado	Acción
<input type="checkbox"/>	 Computadoras	0 bytes	31 dic 1969, 7:00	Elegir Renombrar
<input type="checkbox"/>	 Mundo informático, personalidades y hackers más famosos.	0 bytes	31 dic 1969, 7:00	Elegir Renombrar
<input type="checkbox"/>	 Importancia del agua para la vida	0 bytes	31 dic 1969, 7:00	Elegir Renombrar
<input type="checkbox"/>	 Planificación de procesos	0 bytes	31 dic 1969, 7:00	Elegir Renombrar
<input type="checkbox"/>	 Sistemas de Cabledo Estructurado	0 bytes	31 dic 1969, 7:00	Elegir Renombrar
<input type="checkbox"/>	 Puerto paralelo programable y temporizador de la IBM PC	0 bytes	31 dic 1969, 7:00	Elegir Renombrar
<input type="checkbox"/>	 Definición y características de las Vistas.	0 bytes	31 dic 1969, 7:00	Elegir Renombrar
<input type="checkbox"/>	 El Modelo Relacional y las Matemáticas.	0 bytes	31 dic 1969, 7:00	Elegir Renombrar
<input type="checkbox"/>	 Conceptos asociados al Modelo Relacional	0 bytes	31 dic 1969, 7:00	Elegir Renombrar
<input type="checkbox"/>	 Aprendiendo sobre: "Las armas no letales"	0 bytes	31 dic 1969, 7:00	Elegir Renombrar
<input type="checkbox"/>	 INTERNET COMO CAMPO DE BATALLA	0 bytes	31 dic 1969, 7:00	Elegir Renombrar
<input type="checkbox"/>	 Preparación de la Economía para la Defensa	0 bytes	31 dic 1969, 7:00	Elegir Renombrar
<input type="checkbox"/>	 Proceso de cambio de estructura en la industria cubana de software.	0 bytes	31 dic 1969, 7:00	Elegir Renombrar

Archivo a elegir...

[Crear un directorio](#) [Seleccionar todos](#) [No seleccionar ninguno](#) [Buscar](#)

Figura 8: Interfaz_ Enlazar el Objeto de Aprendizaje

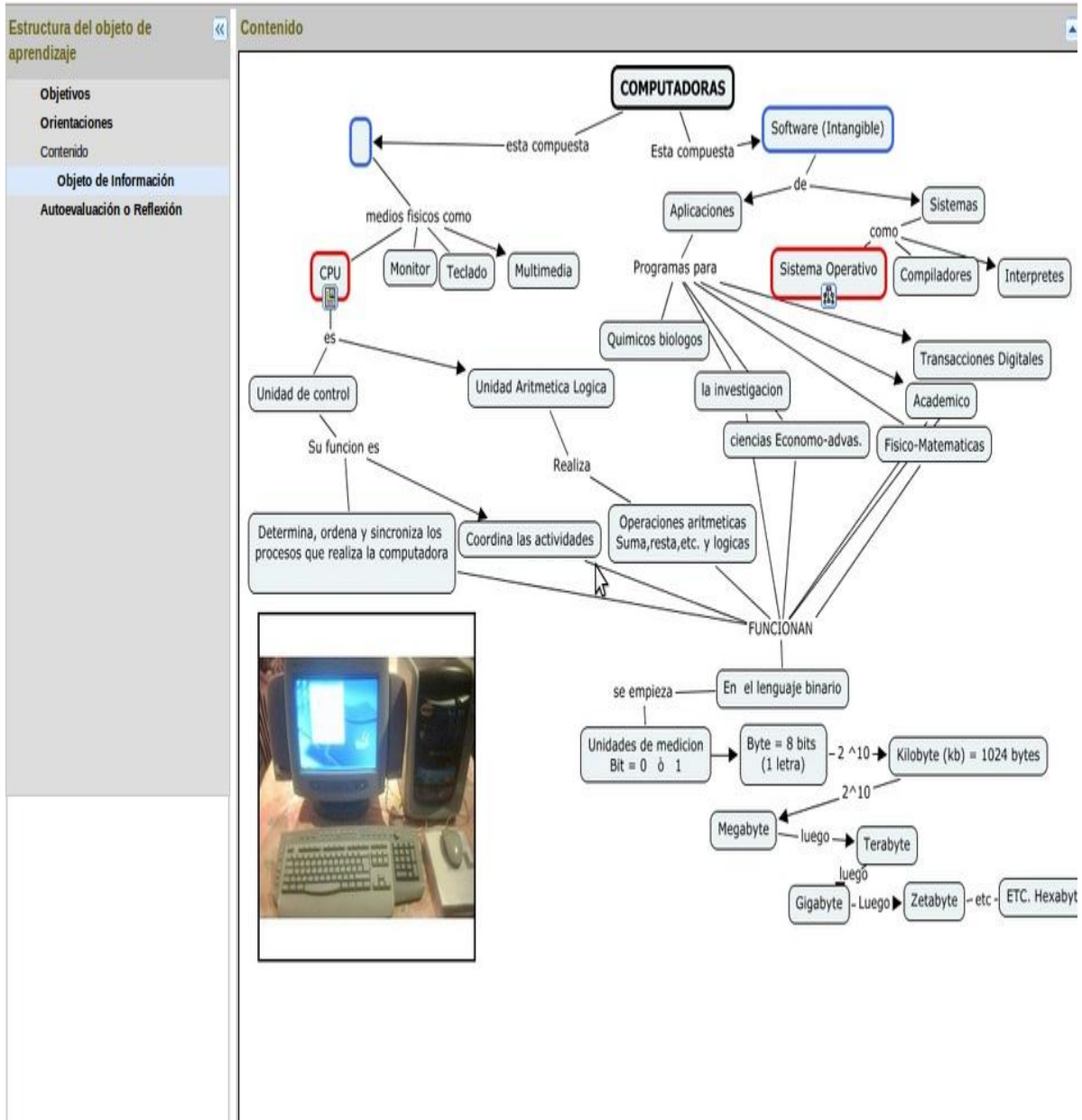


Figura 9: Interfaz_ Visualizar el Objeto de Aprendizaje

Anexo 4: Registro de defectos y dificultades detectados

Elemento	No	No conformidad	Aspecto correspondiente	Etapas de detección	Clasificación
External 2.0	1	Error de traducción	Palabra "External" sin traducir.	1ra iteración	NS
External 2.0	2	Error de traducción	Palabra "Actividades" sin traducir.	1ra iteración	NS
External 2.0	3	Botón no funcional	Botón "Buscar" no funcional.	1ra iteración	S
External 2.0	4	Botón no funcional	Botón "Visualizar OA" no funcional debido a la falta de la librería wso2 para php	2ra iteración	S
External 2.0	5	Interfaz Gráfica	Problemas en la interfaz de la funcionalidad crear la actividad Externo	1ra iteración	R

Tabla 23: Registro de defectos y dificultades detectados.