

**ANÁLISIS Y DISEÑO DE NUEVAS FUNCIONALIDADES PARA EL MÓDULO
ACTIVIDADES EN LA HERRAMIENTA DE AUTOR WEB CRODA 2.0**

**ANALYSIS AND DESIGN OF NEW FUNCTIONALITIES FOR THE MODULE
ACTIVITIES WEB AUTHOR TOOL CRODA 2.0**

Ing. Yasmany Aguilera Sánchez

Universidad de las Ciencias Informáticas, Cuba, yasmanyas@uci.cu, Los Almasigos, Buenaventura, Calixto García, Holguín

RESUMEN: *Con la incorporación de las Tecnologías de la Información y las Comunicaciones (TICs) al proceso de enseñanza-aprendizaje se evidenciaron avances notables, entre ellos el surgimiento del e-learning. A raíz de dicho surgimiento se incrementa de manera gradual la necesidad de compartir recursos y reutilizarlos en cualquier área del conocimiento, surgiendo así los Objetos de Aprendizaje (OA). Con el objetivo de apoyar la teleformación en la Universidad de las Ciencias Informáticas (UCI), se desarrolló en el Centro de Tecnologías para la Formación (FORTES) la herramienta de autor web CRODA, la cual posibilita la creación y edición de Objetos de Aprendizaje de forma flexible haciendo uso de estándares dentro de los que se encuentran: SCORM, LOM e IMS-QTI. Sin embargo, esta herramienta no presenta un suficiente número de actividades que les permita a los profesores desarrollar el proceso de autoevaluación de los estudiantes. La presente investigación asume como idea central, realizar el análisis y diseño de nuevos componentes, que permitan crear las bases para la implementación de nuevas funcionalidades, con el fin de facilitar la creación de ejercicios correspondientes a la sección de autoevaluación del OA, dentro de la herramienta de autor web CRODA.*

Palabras clave: actividades, e-learning, herramienta de autor, objetos de aprendizaje.

ABSTRACT: *With the incorporation of Information Technology and Communication (TIC) to teaching-learning notable advances, including the emergence of e-learning were evident. Following that rise gradually increases the need for resource sharing and reuse in any area of knowledge, thus resulting Learning Objects (OA). In order to support e-learning in the University of Informatics Science (UCI), was developed at the Center for Training Technologies (FORTES) web author tool Croda, which enables the creation and editing of Learning Objects flexibly using standards within which include: SCORM, LOM and IMS-QTI. However, this tool does not present a sufficient number of activities that allow them to teachers develop the self-assessment process of students. This research takes as central idea; the analysis and design of new components that create the foundation for the implementation of new functionalities, with the end facilitate the creation of exercises under section OA self-assessment within web author tool Croda.*

Keywords: activities, e-learning, author tool, learning objects.

1. INTRODUCCIÓN

El avance de la ciencia y la tecnología han provocado en la actualidad grandes cambios en diversas áreas como es el ámbito educativo. Han sido muchas las novedades diseñadas y principalmente han estado relacionados con la forma en que los profesores imparten sus clases y las evalúan. Una de las nuevas modalidades de la pedagogía ha sido la educación a distancia, la que ha evolucionado a la par de las Tecnologías de la Información y las Comunicaciones (TIC); proceso que se desarrolla de una forma acelerada en las últimas décadas.

La web se ha introducido rápidamente como un recurso de apoyo a los esquemas tradicionales de la enseñanza presencial o a distancia. La vinculación de las tecnologías en el proceso de enseñanza-aprendizaje ha dado paso al surgimiento del e-learning el cual es traducido como “aprendizaje electrónico”, y como tal, puede comprender cualquier actividad educativa que utilice medios electrónicos para realizar el proceso formativo. El e-learning no sustituye el método tradicional de enseñanza, pero si ayuda en gran medida al traer consigo mayor interacción o participación del estudiante en las actividades propuestas.

A raíz del surgimiento del e-learning muchas fueron las instituciones educativas que se interesaron en su investigación. Entre estas se encuentra la Universidad de las Ciencias Informáticas (UCI); aplicando un modelo de formación caracterizado por encuentros semi-presenciales, utilizando para ello los recursos didácticos que se encuentran publicados en la universidad. Todo esto conlleva a que se eleve la exigencia en la creación de proyectos productivos con fines educativos, logrando que se establezca el Centro de Tecnologías para la Formación (FORTES) teniendo este como principal función la creación de recursos educativos, así como brindarle soporte a herramientas de este tipo.

Uno de los proyectos que forma parte de este centro es la herramienta de autor web CRODA permitiendo la creación de objetos de aprendizaje (OA) reutilizables, accesibles, duraderos e interoperables, de forma flexible, haciendo uso de los estándares Sharable Content Object Reference Model (SCORM) y Learning Object Metadata (LOM). Esta herramienta les permitirá a los profesores

según su nivel de preparación académica y científico-metodológica, el diseño de OA que permitan atraer la atención de sus estudiantes. Al tener en cuenta su caracterización, los conocimientos que poseen y los nuevos que van a enriquecer su cultura.

El módulo Actividades de la herramienta de autor web CRODA en su versión 1.0 permite la creación de cuestionarios agregándole diferentes tipologías de ejercicios; pero todas estas actividades no son suficientes para satisfacer los intereses del profesor cuando hace uso de la herramienta.

Muchos educadores en la actualidad buscan en la creación de los OA evaluar los conocimientos de forma más amena, dinámica y entretenida a los estudiantes. Teniendo como objetivo principal la presentación de disímiles ejercicios que provoquen la motivación y con ello despertar el interés en la adquisición de conocimientos para que dicho proceso de enseñanza-aprendizaje no sea caracterizado como monótono o tedioso. Al tener en cuenta el bajo número de tipologías de ejercicios que presenta la herramienta de autor web CRODA 1.0 se puede afirmar que provoca la invariabilidad tanto en la autoevaluación como en la evaluación de los conocimientos contenidos en los objetos de aprendizaje y consecuentemente la dificulta la identificación de las áreas de aprendizaje con mayor nivel de dificultad en los estudiantes.

Por lo anteriormente expuesto se plantea el siguiente **problema de investigación**: ¿Cómo contribuir con la evaluación del aprendizaje de los estudiantes a partir de la creación de actividades mediante la elaboración de objetos de aprendizaje haciendo uso de la herramienta de autor web CRODA?

El problema descrito genera como objeto de estudio el proceso de desarrollo de herramientas de autor

Para darle solución al problema a resolver mencionado se define como **objetivo general** de la investigación: elaborar el análisis y diseño de las nuevas funcionalidades para el Módulo Actividades en la herramienta de autor web CRODA 2.0.

Definiendo como **campo de acción** el proceso de análisis y diseño de la herramienta de autor web CRODA.

El objetivo general se desglosa en los siguientes **objetivos específicos**:

- Analizar las tendencias y tecnologías existentes relacionadas con las herramientas educativas.
- Realizar el análisis de las nuevas funcionalidades para el Módulo Actividades en la herramienta de autor web CRODA 2.0.
- Diseñar las nuevas funcionalidades para el Módulo Actividades en la herramienta de autor web CRODA 2.0.

La **idea a defender** radica en que, mediante el desarrollo del análisis y diseño de nuevos componentes del módulo Actividades, se crearán las bases para la implementación de un módulo que permita diversificar la autoevaluación del aprendizaje en los OA desarrollados en la herramienta de autor web CRODA 2.0.

Los **métodos teóricos** que se emplearon en el transcurso de esta investigación son el **Histórico-Lógico** para estudiar el origen y evolución que han tenido las actividades de auto-evaluaciones dentro de los OA, así como los estándares y conceptos relacionados con los mismos. El **Analítico – Sintético** para el análisis de la documentación actual relacionada con el tema y las diferentes propuestas y avances existentes en el mundo del e-learning, extrayendo de ello lo necesario para realizar la propuesta del tema en cuestión. También se utiliza el método teórico **Modelación** que permite la definición y descripción de las funcionalidades que conforman el módulo Actividades y la modelación de diseños de interfaz, siendo estas actividades la base para la integración del módulo propuesto a la herramienta de autor CRODA 2.0. También se usa el método empírico **Observación**, este permite observar y obtener información sobre cómo se gestionan las actividades en otras herramientas y proyectos existentes que ayuden a resolver la problemática planteada.

2. DESARROLLO

1.1 La enseñanza virtual. Introducción al e-learning.

Las TIC se emplean cada vez más en la creación de entornos de aprendizaje más productivos. En todos los sectores de la educación, desde las escuelas primarias, hasta la educación de personas adultas, en centros educativos para alumnos con

necesidades educativas especiales y en las universidades para enriquecer el aprendizaje de los alumnos.

Algunos autores plantean que el e-learning es la “capacitación no presencial que, a través de plataformas tecnológicas, posibilita y flexibiliza el acceso y el tiempo en el proceso de enseñanza-aprendizaje, adecuándolos a las habilidades, necesidades y disponibilidades de cada discente, además de garantizar ambientes de aprendizaje colaborativos mediante el uso de herramientas de comunicación síncrona y asíncrona, potenciando en suma el proceso de gestión basado en competencias” [1].

El e-learning constituye una forma de enseñanza que presenta varias ventajas [2]:

- **Se alternan diversos métodos de enseñanza:** Los participantes pueden trabajar individualmente o de manera grupal.
- **Permite flexibilidad horaria:** El alumno accede en el momento que dispone de tiempo.
- **Aumenta el número de destinatarios:** Esta modalidad de formación se puede dirigir a una audiencia mucho más amplia.
- **Favorece la interacción:** Los alumnos pueden comunicarse unos con otros, con el tutor y con los recursos on-line disponibles en Internet.
- **Disposición de recursos on-line y multimedia:** Internet proporciona acceso instantáneo e ilimitado a una gran cantidad de recursos, como textos, gráficos, videos y animaciones.

1.2 Objetos de aprendizaje.

En la actualidad no existe una definición única sobre OA. Varios autores se han pronunciado al respecto, como Guzmán en el 2005 al plantear que un OA es “cualquier recurso con una intención formativa, compuesto de uno o varios elementos digitales, descrito con metadatos, que pueda ser utilizado y reutilizado dentro de un entorno e-learning puede considerarse un OA”. Por su parte el *Instituto de Ingenieros Eléctrico y Electrónicos* (IEEE) en el 2001 opina que un OA es “una entidad, digital o no digital, que puede ser utilizada, reutilizada y referenciada durante el aprendizaje apoyado con tecnología”. Se considera que “los

objetos de aprendizaje se definen como cualquier entidad, digital o no digital, que puede ser utilizada, reutilizada o referenciada durante el aprendizaje apoyado en la tecnología.” [3].

Como se puede apreciar se dan diversos criterios sobre la definición de OA, para esta investigación se tomará como definición una adaptación a la planteada por la Dra. Clara López, la cual queda de la siguiente manera, un OA es cualquier recurso con una intención formativa, compuesto de uno o varios elementos digitales, descrito con metadatos, que puede ser utilizado y reutilizado para apoyar el aprendizaje [4].

1.3 Comportamiento de un OA en algunas herramientas de la UCI

La herramienta de autor web CRODA posibilita la creación de los OA en un entorno colaborativo; los que son almacenados y publicados en el Repositorio de Objetos de Aprendizaje (RHODA) teniendo en cuenta el uso de estándares que permiten la interoperabilidad entre estos sistemas. A través de este repositorio se tiene acceso a su visualización y descarga. Una vez publicados, los profesores obtendrá la posibilidad de editar aquellos objetos que resalten su interés, perdurando en los metadatos del mismo, las contribuciones de los diferentes autores. Los OA que han sido creados, también serán visualizados como parte de los cursos que se ofrecen en el Entorno Virtual de Aprendizaje (EVA).

1.4 Instructional Management Systems (IMS).

Las especificaciones IMS (2004) son el resultado de una activa iniciativa que está desarrollando y proponiendo especificaciones basadas en tecnologías abiertas (XML) para facilitar las actividades de aprendizaje sobre tecnología web, principalmente para el intercambio de contenidos y de información sobre los estudiantes. Es una propuesta ambiciosa que cubre, entre otros rubros, accesibilidad y adaptación del estudiante, la definición de competencias, el empaquetamiento de contenidos, información de agentes del proceso educativo, el diseño del aprendizaje a través de un lenguaje para expresar diferentes modelos pedagógicos, así como la formación de repositorios de contenidos digitales.

1.5 Tecnologías y herramientas para el desarrollo del módulo.

Para la selección de estas se realiza un estudio de las más conocidas actualmente analizando sus características y ventajas además de mantener las políticas de desarrollo del centro FORTES, en su actual migración al software libre. Al mismo tiempo como el módulo que se propone será incorporado a la herramienta de autor web CRODA en su versión 2.0, se hace una propuesta de las tecnologías de desarrollo necesarias, analizando las empleadas en la herramienta actualmente y otras de interés, ya que el módulo presentado puede ser incorporado a otras herramientas o sistemas.

1.5.1 Metodologías de desarrollo de software.

La ingeniería de software es una disciplina de la ingeniería que comprende todos los aspectos de la producción de software desde las etapas iniciales de la especificación del sistema, hasta el mantenimiento de este después que se utiliza [5]. La base y guía de la ingeniería del software actual son las metodologías de desarrollo de software, estas abarcan dependiendo de las características propias de cada una, todo un conjunto de actividades necesarias en un orden correcto para controlar un proceso de desarrollo de software desde inicio a fin.

1.5.2 Rational Unified Process (RUP)

RUP es un proceso de desarrollo de software perteneciente al grupo de metodologías pesadas, como cualquier proceso de desarrollo de software define quién hace qué, cómo y cuándo. RUP define cuatro elementos: trabajadores (roles), que responden a la pregunta ¿Quién?, las actividades que responden a la pregunta ¿Cómo?, los artefactos (productos), que responden a la pregunta ¿Qué? y los flujos de trabajo de las disciplinas que responde a la pregunta ¿Cuándo? A continuación se detallan estos elementos:

Trabajadores (“quién”): Define el comportamiento y responsabilidades (rol) de un individuo, grupo de individuos, sistema automatizado o máquina, que trabajan en conjunto como un equipo. Ellos realizan las actividades y son propietarios de elementos.

Actividades (“cómo”): Es una tarea que tiene un propósito claro, es realizada por un trabajador y manipula elementos.

Artefactos (“qué”): Productos tangibles del proyecto que son producidos, modificados y usados por las actividades. Pueden ser modelos, elementos dentro del modelo, código fuente y ejecutables.

Flujo de actividades (“Cuándo”): Secuencia de actividades realizadas por trabajadores y que produce un resultado de valor observable.

El ciclo de vida de RUP está caracterizado por ser centrado en la arquitectura, guiado por casos de uso y es iterativo e incremental. RUP divide el proceso en 4 fases, dentro de las cuales se realizan varias iteraciones en número variable según el proyecto y en las que se hace un mayor o menor énfasis en las distintas actividades [6]:

- Inicio: Se hace un plan de fases, se identifican los principales casos de uso y se identifican los riesgos. Se define el alcance del proyecto.
- Elaboración: Se hace un plan de proyecto, se completan los casos de uso y se eliminan los riesgos.
- Construcción: Se concentra en la elaboración de un producto totalmente operativo y el manual de usuario.
- Transición: Se instala el producto en el cliente y se entrena a los usuarios. Como consecuencia de esto suelen surgir nuevos requisitos a ser analizados.

Teniendo en cuenta las características del módulo a desarrollar, las particularidades del equipo de trabajo del proyecto y las características propias de la metodología antes mencionada, se llega a la conclusión que la metodología más adecuada para dirigir el proceso de desarrollo del software es RUP, a pesar de estar recomendada para proyectos extensos y grandes grupos de trabajo, esta define claramente actividades realizadas por roles generando a su paso artefactos que sustentan el proceso de desarrollo del producto. RUP constituye una metodología robusta y como tal basa su éxito en mantener una documentación ordenada a lo largo de todo el proceso de desarrollo, siendo este último un aspecto importante, teniendo en cuenta que el personal del proyecto es muy cambiante y muchas veces

de poca experiencia, por lo que la documentación permitirá que no se pierdan elementos del producto que pueden ser de utilidad para su continuidad. Además el lenguaje utilizado por RUP para la modelación del sistema es UML, el cual brinda amplias posibilidades en la representación, es de fácil uso y conocido por el equipo de proyecto.

1.5.3 Visual Paradigm para UML 6.4.

Visual Paradigm es una herramienta CASE que soporta todo el ciclo de vida del desarrollo de un software: Análisis y Diseño, Construcción, Pruebas y Despliegue. Además permite dibujar todos los diagramas de clases, caso de uso, diagramas de actividades, etc., genera código y documentación desde los diagramas y posibilita el diseño de prototipos de interfaz de usuario. Proporciona además diferentes tutoriales que sirven para un mejor entendimiento de la herramienta [8].

Permite el diseño de software con el UML, posibilita la captura de requisitos con diagrama de requisitos definidos en Systems Modeling Language (SysML), brinda la posibilidad de diseñar bases de datos con el diagrama entidad relación, en general proporciona un entorno unificado de diseño de software para el analista de sistemas y desarrollador de software para analizar, diseñar y mantener aplicaciones de software en una disciplina [8].

Es una herramienta que está disponible en múltiples sistemas operativos como Windows, Linux y Unix y es muy fácil su uso, siendo estas sus principales ventajas.

1.6 Modelo de dominio.

El modelo de dominio es una representación visual de los principales conceptos u objetos del mundo real, significativos para un problema o área de interés. Este es de gran ayuda para desarrolladores y usuarios, ya que de esta forma utilizan un vocabulario común y pueden entender el contexto en que se enmarca el sistema.

1.6.1 Conceptos de las clases del modelo de dominio.

Autor: Es la persona encargada de crear los OA en la herramienta de autor web CRODA.

CRODA: Herramienta de autor web cuyo objetivo principal es permitir la creación de OA, reutilizables, accesibles, duraderos e interoperables, de forma flexible, empleando los estándares SCORM y LOM.

OA: Recurso digital dirigido a la formación, fácilmente identificado y encontrado a través de sus metadatos, con capacidades para integrarse en sistemas e-learning diferentes, para ser usados con distintos propósitos educativos y para combinarse con otros objetos formando una sola entidad.

Recurso: Cualquier tipo de archivo de tipo imagen, audio, video, sonido o texto.

Metadato: Conjunto de información que describe los recursos utilizados dentro del OA para facilitar su búsqueda y reutilización dentro del mismo.

Ejercicios: Conjunto de preguntas de diferentes tipos englobadas en un mismo cuestionario.

1.6.2 Descripción del modelo de dominio.

El autor es un rol autenticado en la herramienta de autor web CRODA que utiliza la misma para la creación de uno o más OA, a estos OA si el autor lo considera le puede incorporar recursos entre los que se encuentran objetos multimedia como imágenes, videos o animaciones y a su vez describirlas haciendo uso de metadatos. Cuando se esté confeccionando el OA el autor del mismo puede realizar uno o varios ejercicios de autoevaluación los cuales podrán estar conformados por una o varias preguntas de diferentes o una misma tipología como pueden ser verdadero o falso, enlazar columnas, selección simple y múltiple, llenar espacios en blanco, orden cronológico y búsqueda en el texto.

1.7 Requisitos del sistema.

Los requerimientos funcionales son declaraciones de los servicios que debe proporcionar el sistema, de la manera en que este debe reaccionar a entradas particulares y de cómo se debe comportar en situaciones particulares.

RF1: La aplicación debe permitir crear preguntas evaluadoras permitiendo seleccionar si las mismas emplearán el estándar QTI o no.

RF2: La aplicación debe permitir crear galería de imágenes.

RF3: La aplicación debe permitir eliminar

galería de imágenes.

RF4: La aplicación debe permitir editar galería de imágenes.

RF5: La aplicación debe permitir exportar a formatos doc, pdf y HTML.

RF6: La aplicación debe permitir visualizar dentro del navegador los archivos de formato pdf.

RF7: La aplicación debe permitir agregar actividades en forma de sopas de palabras.

RF8: La aplicación debe permitir la opción de añadir una nueva palabra a la sopa de palabras.

RF9: La aplicación debe permitir agregar actividades en forma de crucigramas.

RF10: La aplicación debe permitir ubicar las descripciones de las palabras a descubrir en el crucigrama.

RF11: La aplicación debe permitir insertar palabras en el crucigrama dada una descripción.

RF12: La aplicación debe permitir modificar palabras en el crucigrama dada una descripción.

RF13: La aplicación debe permitir crear un puzzle con una imagen.

RF14: La aplicación debe permitir seleccionar la imagen a utilizar en el puzzle.

RF15: La aplicación debe permitir obtener una vista previa del puzzle.

RF16: La aplicación debe permitir cargar una imagen como fondo para el puzzle.

RF17: La aplicación debe permitir crear un puzzle de intercambio con contenido textual.

RF18: La aplicación debe permitir obtener una vista previa del puzzle de intercambio con contenido textual.

RF19: La aplicación debe permitir editar el texto de cada una de las casillas del puzzle de intercambio con contenido textual.

RF20: La aplicación debe permitir cargar una imagen como fondo para el puzzle de intercambio con contenido textual.

RF21: La aplicación debe permitir crear un juego de memoria con elementos iguales.

RF22: La aplicación debe permitir obtener una vista previa del juego de memoria con elementos iguales.

RF23: La aplicación debe permitir crear un ejercicio de asociación simple entre imágenes y texto.

RF24: La aplicación debe permitir crear una actividad de exploración.

RF25: La aplicación debe permitir obtener una vista previa de la actividad de exploración.

RF26: La aplicación debe permitir agregar actividades de selección de palabras o frases dado un conjunto de opciones.

RF27: La aplicación debe permitir seleccionar una palabra o frase dado un conjunto de opciones.

RF28: La aplicación debe permitir agregar actividades que permitan la construcción de textos dada una pregunta específica.

RF29: La aplicación debe permitir que las preguntas dentro de cada examen puedan ser cambiadas de posición cuando el examen este siendo modificado.

1.8 Requisitos no funcionales del sistema.

Los requerimientos no funcionales son propiedades o características que el producto debe tener. Deben pensarse en estas propiedades como las cualidades que permiten que el producto sea atractivo, usable, rápido o confiable. En este trabajo se identificaron los siguientes:

Usabilidad

RNF 1. El sistema podrá ser usado por cualquier persona que posea conocimientos básicos en el manejo de la computadora y de un ambiente Web en sentido general.

RNF 2. En la interacción con el software el sistema mostrará mensajes que sean de entendimiento común con el usuario, para de esta forma elevar la comunicación con el mismo.

RNF 3. En los campos que sean complejos se mostrarán notas indicando ejemplos concretos para guiar el proceso.

Accesibilidad

RNF 4. Las interfaces deben tener vínculos entre ellas para facilitar el proceso.

RNF 5. Protección contra acciones no autorizadas que puedan afectar la integridad de los datos.

Disponibilidad

RNF 6. Una vez publicado el sistema, deberá estar disponible para los usuarios en cualquier momento.

Eficiencia

RNF 7. Tiempos de respuestas rápidas como máximo 10 segundos al igual que la velocidad de procesamiento de la información.

RNF 8. El sistema funcionará de manera óptima en los navegadores Web Mozilla Firefox, Internet Explorer 6 o superior y Opera.

Hardware

RNF 9. La PC del servidor debe de tener al menos 1GB de RAM.

RNF 10. La PC del servidor debe ser Pentium 4 o superior.

RNF 11. La capacidad de almacenamiento del disco duro del servidor debe ser como mínimo 120GB.

RNF 12. La PC cliente debe tener como mínimo 80GB de capacidad de almacenamiento.

RNF 13. La PC cliente debe tener al menos 1GB de RAM.

RNF 14. La PC cliente debe ser Pentium 4 o superior.

Software

RNF 15. Se utilizará PostgreSQL versión 8.4 como gestor de base de datos.

RNF 16. Se utilizará Exist-db versión 1.4 como base de datos de XML nativo.

Restricciones del diseño

RNF 17. Para el modelado visual se deberá usar Visual Paradigm 6.4 como herramienta CASE.

Lenguaje del lado del servidor

RNF 18. Se utilizará como lenguaje de programación del lado del servidor PHP 5.

Entorno de desarrollo integrado IDE

RNF 19. Se hará uso de Netbeans 6.9 como entorno de desarrollo IDE.

Interfaz

RNF 20. Identificación de colores y formatos con la herramienta de autor CRODA.

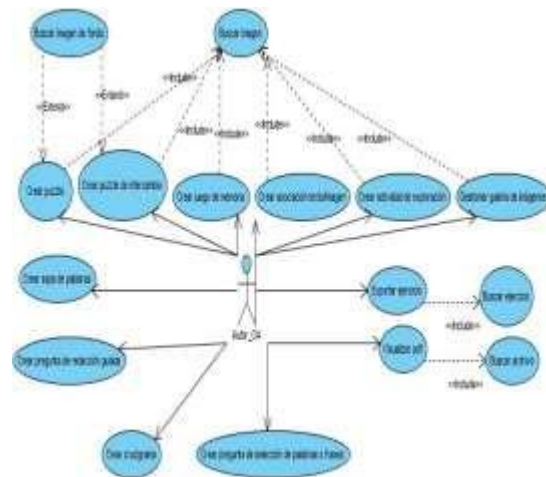


Figura 1: Diagrama de Casos de Usos del Sistema

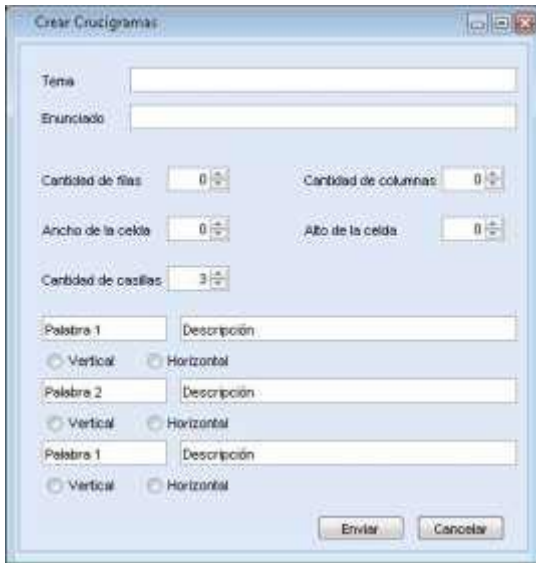


Figura 2: Prototipo de interfaz para crear crucigramas.

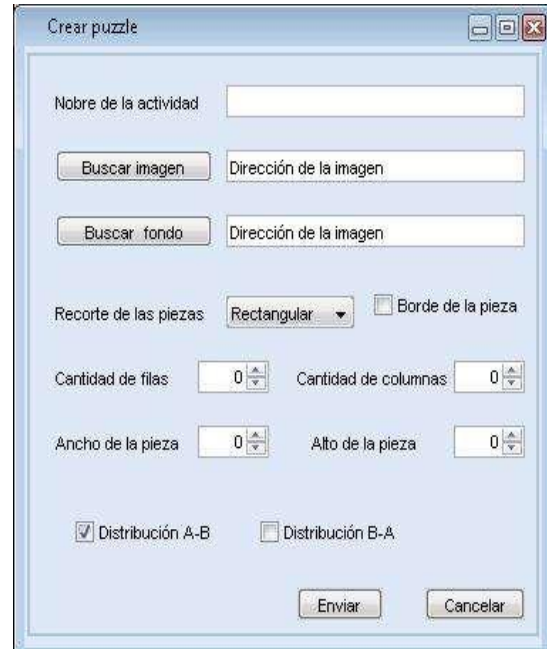


Figura 4: Prototipo de interfaz para crear Puzzle.



Figura 3: Prototipo de interfaz para crear Galería de imágenes.

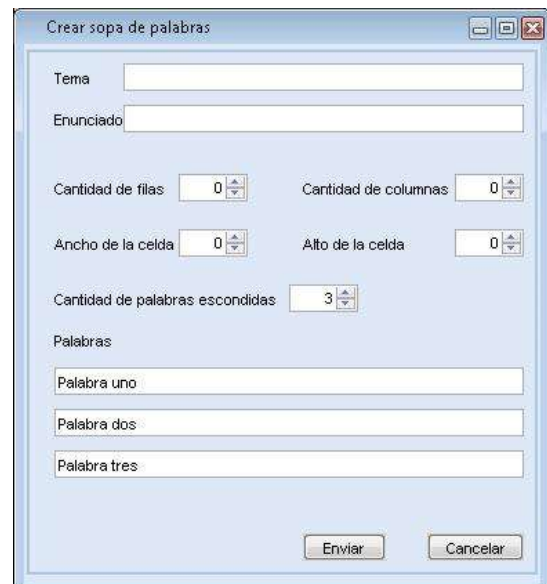


Figura 5: Prototipo de interfaz para crear Sopas de palabras.

3. CONCLUSIONES GENERALES

Con el propósito de darle cumplimiento al objetivo general y a la problemática planteada en el presente trabajo, se han llevado a cabo satisfactoriamente cada una de las tareas que fueron trazadas al comienzo del mismo. Al finalizar la presente investigación, se puede afirmar que:

Con los artefactos generados en los flujos de trabajo Requerimientos y Análisis y diseño de la metodología RUP, se crearon las bases para la futura implementación de las nuevas funcionalidades propuestas para el Módulo Actividades en la herramienta de autor web CRODA 2.0.

Las herramientas seleccionadas soportaron el análisis y diseño de la propuesta de solución.

Se demostró lo beneficioso que resulta la utilización de la metodología de desarrollo de software RUP, lo que propiciará una futura implementación de las nuevas funcionalidades.

4. REFERENCIAS BIBLIOGRÁFICAS

[1] **F. J. García Peñalvo**, «Estado Actual de los sistemas e-learning. 2.» Ediciones Universidad de Salamanca, 2005 , España, 2005.

[2] «Ayuntamiento de Cartagena.,» 2011. [En línea]. Available:
<http://welearn.cartagena.es/moodle/mod/book/view.php?id=509&chapterid=11..> [Último acceso: 3 noviembre 2011].

[3] **LTSC**, «IEEE Standard for Learning Object Metadata,» IEEE-SA, New York, 2002.

[4] **R. H. L. Ramos**, «Implementación de IMS QTI para la realización de cuestionarios en CRODA.,» La Habana, 2011.

[5] **I. Sommerville**, Ingeniería del Software., Madrid: PEARSON EDUCACIÓN, SA. , 2005.

[6] **J. P. Gómez Gallejo**, «Fundamentos de la Metodología RUP,» Universidad Tecnológica de Pereira , 2007.

[7] «IBM,» 2007. [En línea]. Available:
<http://www-142.ibm.com/software/products/es/es/enterprise/> . [Último acceso: 15 Noviembre 2011].

[8] «Visual Paradigm for UML 8.1 Community Edition.,» [En línea]. Available:
<http://www.visual-paradigm.com/product/vpuml/editions/community.jsp> . [Último acceso: 12 Noviembre 2011].

5. SINTESIS CURRICULAR

Yasmany Aguilera Sánchez nació en la provincia Holguín, Cuba. Se Graduó en el año 2012 como Ingeniero en Ciencias de la Informática en la Universidad de Ciencias Informáticas (UCI). Ha participado en múltiples eventos como Pedagogía 2015, el VIII Congreso Internacional Didáctica de las Ciencias y el IV Taller sobre Software, retos y realidades en el siglo XXI del Evento FIMAT XXI. Ha cursado postgrados sobre Redacción de textos científicos y Métodos Estadísticos en la Investigación Científica. Introdujo Resultados Científicos en la Enseñanza de la Física de Secundaria Básica. Actualmente es profesor Instructor en la UCI y se encuentra desarrollando una maestría en la Dirección de Desarrollo del Capital Humano de la propia Universidad.