

SOFTWARE ARCHITECTURE TO INTEGRATE LEARNING OBJECTS BASED ON WEB SERVICES

ARQUITECTURA DE SOFTWARE PARA INTEGRAR OBJETOS DE APRENDIZAJE BASADA EN SERVICIOS WEB

MSc. Mauricio Rojas Contreras*, PhD. Jonas Montilva**

*Universidad de Pamplona

Ciudadela Universitaria. Pamplona, Norte de Santander, Colombia, Tel: (577)-5685303.

E-mail: mrojas@unipamplona.edu.co

**Universidad de los Andes

Facultad de Ingeniería. Sector La Hechicera, Edif. B, Piso 3, Ala sur.

Merida – Venezuela. Telf. 0274 - 2402811, E-mail: jonas@ula.ve

Abstract: Repositories of learning and LMS in most cases operate as two independent systems. As an alternative to integration, this article describes a software architecture supported with web services for e-learning management system with two main functions: Integration of searches for learning objects distributed repositories and building content-based e-learning Platform reuse of learning objects.

Keywords: Software architecture, learning object, SCORM, LMS, learning object Repository, web services.

Resumen: Los repositorios de aprendizaje y los LMS en la mayor parte de los casos funcionan como dos sistemas independientes. Como alternativa de integración, en este artículo, se describe una Arquitectura de software soportada con servicios web para un sistema de gestión de Elearning con dos funcionalidades principales: Integración de búsquedas de objetos de aprendizaje en repositorios distribuidos y construcción de contenidos para plataformas Elearning basado en reutilización de objetos de aprendizaje.

Palabras clave: Arquitectura de software, objeto de aprendizaje, SCORM, LMS, repositorio de objetos de aprendizaje, servicios web.

1. INTRODUCCIÓN

Los proyectos de educación a distancia soportados con nuevas tecnologías de información y comunicación, han venido evolucionando desde la década de los años 80, hasta llegar a los modelos de E-learning que son métodos de aprendizaje en línea que se caracterizan principalmente por dos aspectos: El aprendizaje se puede hacer en cualquier lugar y a cualquier hora, solo se necesita una conexión a Internet y toda la Información reside en un servidor denominado LMS (*Learning Management System*).

Muchas organizaciones se han dedicado a la creación de Entornos Virtuales de Aprendizaje, existiendo así gran cantidad de plataformas que permiten gestionar el proceso académico, lo cual permite a los docentes crear materiales pedagógicos digitales para manejar cursos de forma virtual. Sin embargo, este material solo está disponible para alumnos de una asignatura específica y en una plataforma determinada, lo cual limita su reusabilidad y durabilidad.

En la actualidad la reutilización y la integración de contenidos docentes es una de las prioridades en el mundo del Elearning. De poco sirve un objeto de aprendizaje con un alto nivel de calidad, si solo es accesible por unos cuantos usuarios de una determinada plataforma. Las instituciones educativas requieren de mecanismos de interoperabilidad, ya que no tiene sentido invertir gran cantidad de tiempo y trabajo en la construcción de un recurso para mantenerlo aislado en un mundo cada vez más interconectado y absorbido por la globalización y que debe priorizar la colaboración institucional como mecanismo para garantizar una educación de calidad. Como respuesta a esta inquietud surgen los objetos de aprendizaje y los repositorios de objetos de aprendizaje que permiten compartir y reutilizar recursos.

Con el fin de garantizar la interoperabilidad, integración y reusabilidad de los objetos de aprendizaje han surgido los repositorios de objetos de aprendizaje implementados con servicios Web, los cuales aplican todas las características de las arquitecturas orientadas a servicios (SOA) a este tipo de sistemas. La utilización de estándares como SCORM en el diseño estructural de los objetos complementados con las tecnologías asociadas a las SOA permite garantizar la independencia total de la plataforma donde se ejecute.

De igual forma, los LMS permiten reutilizar los objetos de aprendizaje seleccionados en los sistemas de repositorio en la construcción de nuevos contenidos o cursos. Sin embargo, los repositorios y los LMS trabajan en forma independiente. Lo anterior implica que la persona encargada de diseñar el material educativo primero debe buscar el objeto de aprendizaje en un repositorio de forma manual y luego debe integrarlo usando un LMS.

Con el fin de dar solución a este problema, se presenta en este trabajo una Arquitectura de software para la integración de objetos de aprendizaje que integra las dos funcionalidades enunciadas anteriormente en una sola aplicación. Como particularidades de esta arquitectura de software se debe enunciar que se utiliza como estilo arquitectónico la orientación a servicios. De igual manera, el diseño plantea un sistema de búsqueda de objetos de aprendizaje en un conjunto de repositorios distribuidos.

El artículo está organizado de la siguiente manera: En la sección 2, se hace una introducción teórica a

los objetos de aprendizaje. En forma específica, se describen algunas definiciones de objetos de aprendizaje, repositorios de objetos y finalmente se hace una breve descripción del estándar SCORM. En la sección 3, se describen los fundamentos teóricos de las arquitecturas orientadas a servicios. En forma particular se detallan los aspectos principales de las arquitecturas orientadas a servicios y las tecnologías que la soportan. En la sección 4 se describe la propuesta de Arquitectura de software del Sistema de gestión de Elearning soportada con servicios Web. Finalmente, en la sección 5, se presentan las conclusiones del trabajo.

2. OBJETOS DE APRENDIZAJE

2.1 Definición de objetos de aprendizaje

Entre las múltiples definiciones de objeto de aprendizaje se encuentra la proporcionada por el estándar de metadatos LOM, que define objeto de aprendizaje como: "Cualquier entidad, digital o no digital, que puede ser utilizada para el aprendizaje, la educación o la enseñanza". (LOM, 2002).

La anterior definición es bastante genérica y ha hecho que se proporcionen otras definiciones más específicas como las recogidas en el artículo de Polsani, en el que se define como: "Un objeto o conjunto de recursos que pueden ser utilizados para facilitar ciertos resultados educativos y ser extraídos y reutilizados en otros entornos educativos". (Mills, 2002).

Otra definición, describe objeto de aprendizaje como: "Cualquier recurso digital o no digital que pueda ser reutilizado como soporte para el aprendizaje" (Wiley, 2002). En esta definición también se destaca que los objetos de aprendizaje se refieren a material educativo diseñado y creado en pequeñas unidades con el propósito de maximizar el número de situaciones educativas en las que se puede utilizar dicho recurso. Esta idea está directamente recogida en la definición proporcionada por Polsani que define a los objetos de aprendizaje como "unidad didáctica de contenido, autocontenida e independiente, predispuesta para su reutilización en múltiples contextos educativos". (Polsani, 2003).

Para lograr la reutilización y la interoperabilidad de los objetos de aprendizaje varios autores indican que estos deben estar descritos por ciertas sentencias externas a ellos expresadas en un lenguaje, denominados metadatos. Estas

estructuras físicamente son externos al propio recurso, utilizan un formato técnico para su expresión y para su intercambio, generalmente lenguajes definidos sobre XML, también utilizan una serie de descriptores, campos o elementos normalizados para conseguir un cierto grado de interoperabilidad entre diferentes sistemas.

Otra definición un poco más formal y completa es la construida por el Ministerio de Educación Nacional de Colombia: "*Un Objeto de Aprendizaje es un conjunto de recursos digitales, autocontenible y reutilizable, con un propósito educativo y constituido por al menos tres componentes internos: contenidos, actividades de aprendizaje y elementos de contextualización. El Objeto de Aprendizaje debe tener una estructura de información externa (metadatos) que facilite su almacenamiento, identificación y recuperación*". (MEN, 2006).

2.2 La especificación SCORM

Según la visión de *Advanced Distributed Learning* (ADL), la presencia de las distintas especificaciones propuestas por diversos grupos no resultaba suficiente para garantizar los siguientes objetivos fundamentales identificados cuando la iniciativa fue lanzada: (Lozano, 2005)

- Poder trasladar cursos de un LMS a otro
- Reutilizar piezas de contenido en distintos cursos
- Secuenciar estos contenidos reutilizables con soporte para ramificaciones, planes alternativos u otras estrategias de aprendizaje adaptables
- Realizar búsquedas en bibliotecas de contenido o repositorios a través de distintos LMS

En particular, ADL se basó en la afirmación de que, aunque existiesen especificaciones cubriendo estos aspectos de la interoperabilidad, en la práctica esto no era posible por falta de implantación de las especificaciones en algunos casos y por conflictos entre especificaciones en otros casos.

Así, ADL propuso el modelo SCORM con el objetivo de establecer un marco común para el aprendizaje asistido por computadora y basado en la red Internet. Este marco común provee un conjunto de guías, especificaciones y estándares basados en las especificaciones previamente

existentes en el campo propuestas por distintas organizaciones.

La definición del estándar SCORM, así como su evolución y las distintas decisiones de diseño tomadas durante el proceso de especificación, se basan en 6 principios:

- Accesibilidad
- Adaptabilidad
- Asequibilidad
- Durabilidad
- Interoperabilidad
- Reusabilidad

2.3 Repositorios de Objetos de Aprendizaje

La primera aproximación a la reutilización de objetos de aprendizaje reside en el diseño de repositorios o almacenes digitales de contenidos educativos. Así, de esta forma, podrán ser reutilizados por una mayor cantidad de clientes potenciales. (Otón et al., 2010)

A continuación se presentan algunas definiciones de repositorios de objetos de aprendizaje:

- El JORUM+ Project (2004) adopta la siguiente definición: "Un ROA es una colección de OA que tienen información (metadatos) detallada que es accesible vía Internet. Además de alojar los OA los ROA pueden almacenar las ubicaciones de aquellos objetos almacenados en otros sitios, tanto en línea como en ubicaciones locales". (JORUM, 2004)
- Un repositorio de objetos de aprendizaje es un sistema software que almacena recursos educativos y sus metadatos (o solamente estos últimos), y que proporciona algún tipo de interfaz de búsqueda de los mismos, bien para interacción con humanos o con otros sistemas software. (ANCED, 2010)

2.3.1 Tipos y características de ROA

Por la forma en la que se concentran los recursos, principalmente se identifican dos tipos de ROA: (Downes, 2010)

- Los que contienen los objetos de aprendizaje y sus metadatos, en éstos los objetos y sus descriptores se encuentran dentro de un mismo sistema e incluso dentro de un mismo servidor.
- Los que contienen sólo los metadatos, en este caso el repositorio contiene sólo los

descriptores y se accede al objeto a través de una referencia a su ubicación física que se encuentra en otro sistema o repositorio de objetos.

Por la forma en la que los catálogos de metadatos se organizan, se diferencian dos modelos de ROA: (Downes, 2010)

- Centralizados: En este modelo los metadatos de los OA están contenidos en un mismo servidor, aunque el objeto esté localizado en alguno otro.
- Distribuidos: Opera a través de varios servidores, cada uno contiene diferentes grupos de metadatos y se comunican entre ellos para intercambiarlos.

3. ARQUITECTURAS ORIENTADAS A SERVICIOS (SOA)

En la actualidad el concepto de SOA ha resurgido, gracias al auge y madurez de una nueva tecnología: los Servicios Web. Los Servicios Web se han convertido en el estandarte de SOA, ya que esta tecnología posee un conjunto de características que permiten cubrir todos los principios de la orientación a servicios.

Una arquitectura orientada a servicios (SOA) es una forma de compartir funciones de manera generalizada y flexible. (Ort, 2005).

En general se puede decir que SOA es un modelo de arquitectura que establece un marco de diseño para la integración de aplicaciones independientes de manera que desde la red pueda accederse a sus funcionalidades, las cuales se ofrecen como servicios. La forma más habitual de implementarla es mediante Servicios Web, una tecnología basada en estándares e independiente de la plataforma.

Básicamente una arquitectura orientada al servicio es una colección de servicios. Estos servicios se comunican entre sí. La comunicación puede involucrar simplemente el paso de datos o la coordinación de alguna actividad entre varios servicios.

La característica principal de SOA es que es una Arquitectura con acoplamiento débil, esto significa que el cliente de un servicio es independiente de la construcción de ese servicio.

3.1 Tecnologías que soportan a los Servicios Web

Las tecnologías que definen la arquitectura de un Servicio Web se pueden observar en la figura 1.

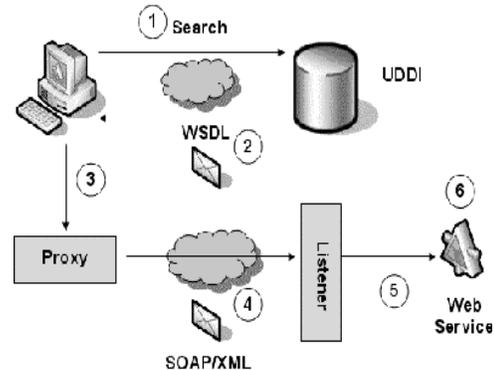


Fig. 1. Proceso y tecnologías de los Servicios Web.

3.1.1 XML (Extensible Markup Language)

Es un estándar para describir datos y crear etiquetas. Las características especiales son la independencia de datos, o la separación de los contenidos de su presentación.

El lenguaje XML es una forma, estándar industrial e independiente del sistema, de representar datos. Los datos que se representan usando XML se pueden publicar en múltiples medios porque XML describe la estructura de los datos, no su formato, al contrario que el HTML, los datos de XML se pueden pasar entre aplicaciones porque la estructura de los datos se puede especificar en un esquema, lo que permite que un analizador de sintaxis valide y procese los datos que siguen el esquema.

3.1.2 WSDL (Web Services Description Language)

Es un formato en XML estandarizado para describir servicios web, en este archivo se describe el nombre, la ubicación y la forma de comunicarse con el servicio, así como los métodos y parámetros que utiliza y la forma en que devuelve la respuesta. Así, un usuario puede crear una aplicación cliente que comunica con el servicio web. (Barco, 2006).

3.1.3 SOAP (Simple Object Access Protocol)

Es un protocolo de comunicación, por paso de mensajes XML, se utiliza para la formación de los mensajes intercambiados entre los sistemas distribuidos y la red.

Los mensajes SOAP son independientes de los sistemas operativos y de los protocolos, y pueden ser transportados usando una variedad de protocolos Internet, incluyendo SMTP, y HTTP. El mensaje está compuesto de tres partes: un sobre, un encabezado y el cuerpo.

3.1.4 UDDI (*Universal Description, Discovery And Integration*)

Universal Descripción de Descubrimiento e Integración, es un elemento básico sobre el que se asientan los Servicios web, hace posible que empresas pueden tanto publicar como encontrar Servicios web.

UDDI está construido sobre los estándares de Internet del W3C y de la IETF (Internet Engineering Task Force), como XML, HTTP. Para describir las interfaces hacia los servicios web, utiliza el lenguaje WSDL, las aplicaciones interesadas pueden consultar utilizando mensajes SOAP, lo cual posibilita una interoperabilidad total. (Mateu, 2004).

4. ARQUITECTURA DE SOFTWARE DE UN SISTEMA DE GESTIÓN DE E-LEARNING SOPORTADO CON SERVICIOS WEB

4.1 Definición de arquitectura de software

Los señores Bass, Clements y Kazman (Bass *et al.*, 2003) definen arquitectura de software de la siguiente manera:

“La arquitectura de software de un programa o sistema de computo es la estructura o estructuras del sistema, lo que comprende a los componentes del software, sus propiedades externas visibles y las relaciones entre ellos”.

La arquitectura de software es una representación que permite:

1. Analizar la efectividad del diseño para cumplir los requerimientos establecidos.
2. Considerar alternativas arquitectónicas en una etapa en la que hacer cambios al diseño todavía es relativamente fácil.
3. Reducir los riesgos asociados con la construcción del software. (Pressman, 2010).

4.2 Importancia de la arquitectura de software

A continuación se enumeran tres razones clave por las que es importante la arquitectura de software:

- Las representaciones de la arquitectura del software permiten la comunicación entre todas las partes (participantes) interesadas en el desarrollo de un sistema basado en computadora.
- La arquitectura resalta las primeras decisiones que tendrán un efecto profundo en todo el trabajo de ingeniería software siguiente y, también importante, en el éxito último del sistema como entidad operacional.
- La arquitectura “constituye un modelo relativamente pequeño y asequible por la vía intelectual sobre cómo está estructurado el sistema y la forma en la que sus componentes trabajan juntos”. (Bass *et al.*, 2003)

4.3 Arquitectura de software del sistema de gestión de e-learning

La arquitectura de software del sistema está estructurada en 7 capas como se muestra en la figura 2.

En forma general, la arquitectura de software esta diseñada tomando como política de construcción la utilización de servicios web en cada una de las capas y en la comunicación entre capa y capa a través de adaptadores como se observa en el diagrama de componentes de la figura 2.

La estructura de la arquitectura de software se describe a continuación de forma ascendente:

Capa almacenamiento: En esta capa se ubican los repositorios de objetos de aprendizaje con sus respectivos metadatos. La estructura interna de cada repositorio debe ajustarse al estándar SCORM.

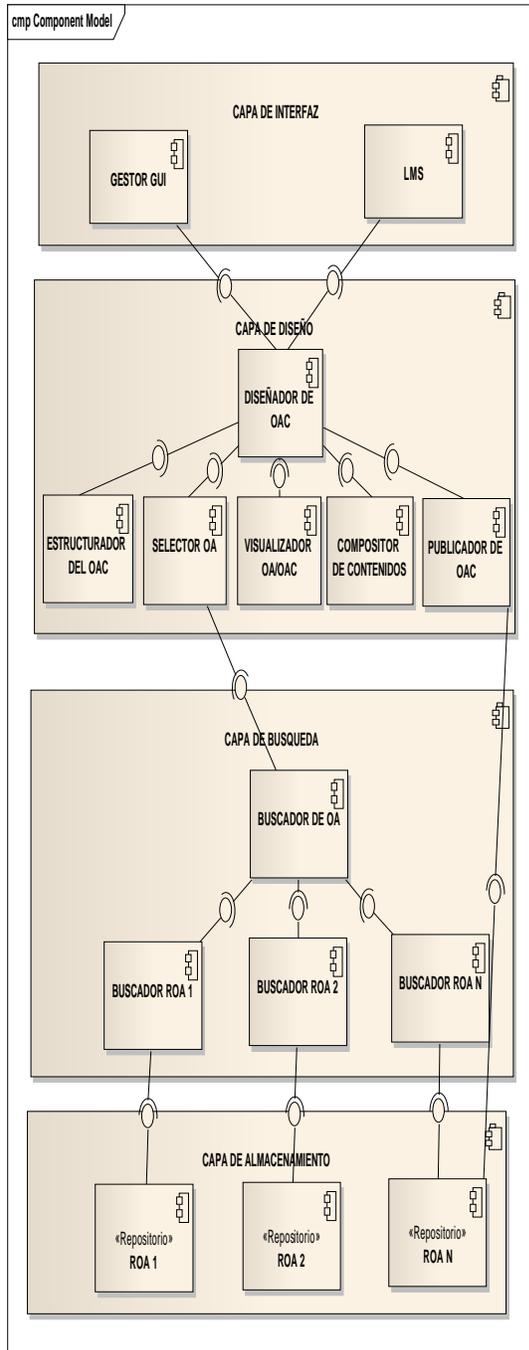


Fig. 2: Arquitectura de software Sistema de gestión de E-learning

Capa de búsqueda: En esta capa se ubican los servicios de búsqueda para cada uno de los repositorios de la capa de almacenamiento. Estos servicios deben suministrar como salida los ítems que coinciden con los criterios de búsqueda seleccionados por el cliente.

Capa de búsquedas: En esta capa se ubica un servicio que toma los ítems de cada uno de los

buscadores de la capa de búsqueda y los integra para mostrar una sola lista al cliente a través del servicio BUSCADOR OA.

Capa de diseño: permite al usuario diseñar y publicar un curso en línea, o un componente de él, mediante la integración de objetos de aprendizaje provenientes de diferentes fuentes o repositorios. El diseño se puede complementar con la adición de otras actividades académicas y/o contenidos de aprendizaje adicionales.

Capa de interfaz: permite al usuario acceder a los servicios de diseño de contenidos instruccionales. Esta capa, también, permite a un LMS acceder directamente a los servicios que ofrece el integrador usando sus interfaces de programación.

5. CONCLUSIONES

La arquitectura de software es la estructura del sistema, lo que comprende en forma general a sus componentes y relaciones entre ellos. Permite analizar la efectividad del diseño para cumplir con requerimientos funcionales y no funcionales del sistema. En el nivel arquitectónico, no se especifican las propiedades internas de los componentes sólo se especifican los componentes y sus relaciones.

La arquitectura propuesta permite integrar dos grandes funcionalidades en un solo sistema que en épocas anteriores se debían realizar a través de sistemas independientes de repositorios y LMS. En forma específica, la arquitectura en las capas de almacenamiento y datos, de búsqueda y de integración de búsqueda integra los servicios que permiten implementar un sistema de búsqueda de objetos de aprendizaje con repositorios distribuidos.

La capa de selección ofrece un conjunto de servicios que permiten a un diseñador mejorar los criterios y argumentos para la selección de un objeto de aprendizaje a través de funcionalidades que permiten la visualización previa de los objetos de aprendizaje, comparación de dos objetos y descripción de nuevas búsquedas.

Las capas de diseño, implementación permiten al usuario integrar a un curso un objeto de aprendizaje complementando el diseño con la adición de otras actividades académicas y/o objetos de aprendizaje adicionales.

La capa de uso permite al usuario estudiante acceder a los contenidos de los cursos y al usuario docente/orientador hacer seguimiento y control sobre las actividades del curso.

En forma general, la arquitectura de software permite verificar en un tiempo rápido el cumplimiento de los requerimientos del sistema y minimiza el costo de cambios o ajustes al sistema en una etapa temprana de implementación. Adicionalmente, permite reducir los riesgos de implementación del sistema en un tiempo corto.

La utilización de servicios web en el diseño de este tipo de arquitectura complementado con el uso de estándares permite dar solución a problemas de interoperabilidad, reutilización e integración de objetos de aprendizaje en sistemas de este tipo que permiten la producción de contenidos de cursos para entornos virtuales.

REFERENCIAS

- ANCED, 2010. Libro de buenas prácticas de E-Learning. ANCED, Asociación Nacional de Centros de e-Learning y Distancia. Disponible: <http://www.buenaspracticaselearning.com/capitulo-16-estandares-e-learning.html>
- Barco, Antonio, 2006. WSDL: El contrato de un servicio. 2006. Disponible en: <http://arquitecturaorientadaaservicios.blogspot.com/2006/12/articulo-tecnologico-wsdl-el-contrato-de.html>
- Bass, L.; P. Clements and R. Kazman, 2003. Software Architecture in Practice, 2a. ed., Addison-Wesley.
- Downes, Stephen, 2010. The Learning Marketplace. Meaning, Metadata and Content Syndication in the Learning Object Economy. <http://www.downes.ca/files/book3.html>
- JORUM, 2004. JORUM+ Proyect. The JISC Repository for [learning an teaching materials]. http://www.jorum.ac.uk/docs/Vol1_fin.pdf
- LOM, 2002. "Draft Standard for Learning Object Metadata. IEEE 1484".12.1-2002, 15 July. Versión en línea. Consultado el 5 de Abril: http://ltsc.ieee.org/wg12/files/LOM_1484_12_1_v1_Final_Draft.pdf
- Lozano, Javier, 2005. Análisis de los estándares de desarrollo de contenidos de e-Learning: Evolución y ventajas sobre la calidad del producto final. Seminario Internacional Virtual Educa Cono Sur: La calidad en los entornos virtuales de aprendizaje: <http://www.iesevirtual.edu.ar/virtualeduca/po-nencias/>
- Mateu, Carles, 2004. Desarrollo de aplicaciones web. Primera edición. 2004. Fundación Universidad Oberta de Cataluña. http://www.uoc.edu/masters/softwarelibre/esp/materials/Desarrollo_web.pdf
- MEN, Ministerio de Educación Nacional Colombiano, 2006. Objetos Virtuales de Aprendizaje e Informativos. Consultado abril 6 de 2010, en Portal Colombia Aprende. Disponible desde: <http://www.colombiaprende.edu.co/html/directivos/1598/article-172369.htm>.
- Mills, S., 2002. "Learning about learning objects with learning objects", en Proceedings of Society for Information Technology and Teacher Education International Conference, Vol. 1, AACE, págs. 1158-1160.
- Ort, Ed., 2005. Service-Oriented Architecture and Web Services: Concepts, Technologies, and Tools. 2005. Disponible en: <http://java.sun.com/developer/technicalArticles/WebServices/soa2/>.
- Otón, Salvador; Ortiz, Antonio y Hilera, Ramón, 2010. SROA: Sistema de Reutilización de Objetos de Aprendizaje. Dpto. Ciencias de la Computación. ETS de Ingeniería Informática Universidad de Alcalá. <http://161.67.140.29/iecom/index.php/IECom/article/viewFile/21/15>.
- Polsani, P. R., 2003. "Use and abuse of reusable learning objects", Journal of Digital Information, Vol. 3, No. 4, Artículo 164. Disponible en: <http://jodi.ecs.soton.ac.uk/Articles/v03/i04/Polsani/>
- Pressman, R., 2010. Ingeniería del software un enfoque práctico, 7a ed., McGraw Hill, 2010.
- Wiley, D. A., 2002. "Connecting learning objects to instructional design theory: A definition, a metaphor and a taxonomy", en D. A. Wiley (ed.), The instructional use of learning objects , Agency for Instructional Technology and Association for Educational Communications and Technology, Bloomington, Indiana, págs. 3-24. Disponible: <http://www.reusability.org/read/>