

Propuesta de Arquitectura basada en Componentes de Software para el desarrollo de Materiales Educativos Didácticos Web bajo el modelo de Educación Basada en Web

Architecture Proposal Based on Software Components for the Development of Web Educational Materials under the Web-Based Education model

Rubén Peredo Valderrama¹ Iván Peredo Valderrama²

¹ Escuela Superior de Cómputo del Instituto Politécnico Nacional, Av. Juan de Dios Bátiz S/N esquina con Miguel Othón de Mendizábal, México, D.F., 07738. México
rperedo@ipn.mx

² Universidad Politécnica de Querétaro, Carretera Estatal 420 S/N el Rosario el Marqués, México, Querétaro, CP. 76240.
ivan.peredo@upq.edu.mx

Fecha de recepción: 29 de julio de 2019

Fecha de aceptación: 30 de agosto de 2019

Resumen. La presente propuesta presenta un modelo basado en componentes de software para el desarrollo de materiales educativos didácticos Web para los cambiantes y nuevos escenarios escolares. Desarrollar Materiales Educativos Didácticos para la Web (World Wide Web, WWW por sus siglas en inglés) es una labor compleja para un profesor, ya que hay múltiples factores a considerar como: integración de múltiples tecnologías, estándares, multimedia e interactividad, reutilización, modelos pedagógicos, nuevos escenarios escolares, etc. La presente propuesta presenta un punto de vista innovador basada en componentes de software, la presente propuesta busca disminuir dependencias de contexto propietarias de propuestas previas, apeándose más a estándares internacionales, y desarrollar una infraestructura, además de permitir composición y secuenciación dinámica de los materiales educativos. La presente propuesta propone el uso de tecnologías de Web semántica para el procesamiento automático de la información generada por la interacción de los alumnos y los materiales educativos.

Palabras clave: Materiales Educativos, Componentes de Software, Patrones de Diseño, Web semántica.

Summary. This proposal presents a model based on software components for the development of educational web teaching materials for changing and new school settings. Developing Educational Teaching Materials for the Web (World Wide Web, WWW) is a complex task for a teacher, since there are multiple factors to consider such as: integration of multiple technologies, standards, multimedia and interactivity, reuse, models pedagogical, new school settings, etc. This proposal presents an innovative point of view based on software components, this proposal seeks to reduce context dependencies that are proprietary to previous proposals, adhering more to international standards, and developing an infrastructure, in addition to allowing dynamic composition and sequencing of educational materials. This proposal proposes the use of semantic Web technologies for the automatic processing of the information generated by the interaction of the students and the educational materials.

Keywords: Educational Materials, Software Components, Design Patterns, Semantic Web.

1 Introducción

La real academia de la lengua española define la arquitectura de la siguiente manera: *“estructura lógica y física de los componentes de una computadora”* [1], esta definición lamentablemente es demasiado general. La meta del diseño de software es construir un modelo que se adecue a los requerimientos del cliente, y que lo guíe a una implementación exitosa. El éxito de un software grande depende del éxito de su arquitectura. Dado la importancia de lo anteriormente mencionado definir el diseño de una arquitectura toma gran relevancia, una buena definición es la siguiente: *“el diseño de la arquitectura define la relación entre los mayores elementos estructurales de un software, estilos y patrones que pueden ser usados para llevar a cabo los requerimientos definidos para el sistema, y las restricciones que afectan la manera en la cual la arquitectura puede ser implementada”* [2].

Los componentes de software han emergido como una solución para el desarrollo de software. Poco a poco se han estado estableciendo varios tipos de mercados de componentes de software, ya que son elementos pre construidos que nos permiten desarrollar software, los avances tecnológicos están estimulando el desarrollo de componentes, algunas de las iniciativas más destacadas en el área de componentes de software son las siguientes: componentes VBX [3], CORBA [4], COM [5], DCOM [6], ActiveX [7], JavaBeans [8], etc.

Clements Szyperski es considerado como una de las máximas autoridades en el área de componentes de software, en su libro: *“Component Software: Beyond Object-Oriented Programming”* define un componente de software de la siguiente manera: *“un componente de software es una unidad de composición con interfaces especificadas contractualmente y dependencias de contexto explícitas. Un componente de software puede ser*

entregado independientemente y es tema de composición para terceras partes”. Una premisa fundamental de los componentes es la siguiente: “los componentes son para composición”, esta premisa parece muy obvia, pero no lo es, la composición permite construir cosas para ser reusadas reconfigurándolas de múltiples maneras para desarrollar nuevas composiciones [9].

Elaborar materiales educativos didácticos para la Web es complejo como se menciona en el resumen de la presente propuesta, el profesor debe considerar variados factores para elaborar sus materiales educativos bajo estándares internacionales, las dos más importantes iniciativas en el tema son las siguientes: Aprendizaje Distribuido Avanzado (*Advanced Distributed Learning*, ADL por sus siglas en inglés) [10] y Consorcio de Aprendizaje Global - Sistema de Manejo Instruccional (*Instructional Management System Global Learning Consortium*, IMS GLC por sus siglas en inglés) [11].

Muchas instituciones tienen presencia en la Web con cursos en línea, pero pocas de estas tienen una estrategia de mediano y largo plazo para adaptar sus cursos a los requerimientos cambiantes, la presente propuesta presenta los avances actuales de una innovadora propuesta que hemos ido desarrollado a lo largo de los años, fundada en componentes de software, para ir conformando una plataforma que reduzca la elevada complejidad para desarrollar materiales educativos de alta calidad en el corto plazo, pero teniendo una estrategia de mediano y largo plazo, mejorando el mantenimiento de los diferentes materiales educativos a lo largo de su vida.

2 Estado del Arte

Muchas empresas, organizaciones e instituciones buscan hacerse presentes en la Internet/Web al ver las ventajas que esta ofrece para estas, muchas instituciones educativas han comprendido poco a poco la importancia de utilizar la Internet/Web, con la finalidad de obtener las ventajas inherentes de la misma, aparte de poner los aspectos administrativos de las mismas en línea, tiene una importancia fundamental tener cursos en línea, ya que lentamente fueron haciéndose evidentes las innegables ventajas de la educación en línea, pero inexorablemente se fue manifestando también que desarrollar materiales educativos didácticos en línea es un tema complejo, como se mencionó en la introducción, y la mayoría de las instituciones tienen estrategias de corto plazo, debido a que no manejan el cambio de manera adecuada, y un cambio en los requerimientos del cliente ocasiona que se tengan que readaptar de acuerdo a los nuevos requerimientos, ocasionando que en muchos casos se tenga que reconstruir completamente desde cero de acuerdo a los nuevos requerimientos, ya que no hay una clara estrategia para manejar el cambio de los materiales educativos en el mediano y largo plazo.

Los patrones de diseño de software son buenas prácticas de programación, estos contienen el conocimiento acumulado de expertos, los cuales están siendo constantemente actualizados. La meta más general que buscan los patrones de diseño de software es facilitar el manejo del cambio en los proyectos de software. Los patrones de diseño de software buscan resolver problemas generales repetitivos, aplicables a escenarios específicos, creando soluciones concretas dentro de nuestros propios desarrollos de software, resultando en que nuestras propuestas estén respaldadas por el conocimiento acumulado de expertos, resultando en aplicaciones más profesionales, ya que muchas de las problemáticas han sido analizadas y resueltas desde una visión de múltiples expertos del área.

El IMS GLC busca promover las tecnologías de aprendizaje en las instituciones educativas, y capacitación en las empresas. La organización es de alcance mundial sin fines de lucro, fundada en 1995. IMS GLC tiene un fuerte apoyo de múltiples instituciones educativas, las cuales han adoptado sus propuestas, además del soporte múltiples empresas, la tarea principal de IMS GLC está enfocada en el desarrollo de estándares internacionales para el desarrollo de tecnologías educativas para el aprendizaje, los cuales pueden ser utilizados de manera gratuita. Las áreas de interés más importantes del IMS GLC son las siguientes: Interoperabilidad, Especificaciones de interoperabilidad, Empaquetados de contenido, y Cartucho Común (*Common Cartridge*) [11].

La iniciativa de ADL es un programa gubernamental de los Estados Unidos, para la investigación y desarrollo de materiales de aprendizaje distribuido, busca coordinar las sinergias de las organizaciones públicas y privadas. Una de las aportaciones más notables de la ADL es el Modelo de Referencia de Objeto de Contenido Compartido (*Sharable Content Object Reference Model*, SCORM por sus siglas en inglés), siendo un conjunto de especificaciones técnicas, que conforman estándares, que buscan la importación y reutilización de los materiales educativos que adopten el estándar, el cual es utilizado para desarrollar materiales educativos estructurados. ADL busca que la gestión de los contenidos esté en formatos distribuíbles, compartibles, con formatos abiertos. ADL con SCORM busca mejorar la reutilización de los materiales educativos, para desarrollar materiales educativos e importarlos desde Sistemas Manejadores de Contenido (*Learning Management System*, LMS por sus siglas en inglés) diferentes desde los cuales fueron desarrollados, siempre y cuando durante el desarrollo de los materiales educativos se basen en el estándar SCORM. ADL/SCORM tiene tres especificaciones principales: Modelo de Agregación de Contenido (*Content Aggregation Model*, CAM por sus siglas en inglés), Ambiente en Tiempo de Ejecución (*Run-Time Environment*, RTE por sus siglas en inglés), y Secuenciación y Navegación (*Sequencing and Navigation*, SN por sus siglas en inglés). La especificación CAM establece el empaquetamiento de los materiales educativos, para el intercambio de un sistema a otro. La especificación del RTE establece un modelo de datos, y

una Interface de Programación de Aplicación (*Application Programming Interface*, API por sus siglas en inglés) para materiales educativos. La especificación SN en conjunto con CAM establecen la conformación del contenido para la entrega a los estudiantes, que permite una secuenciación dinámica de los contenidos, de acuerdo a las reglas de secuenciación establecidas durante el desarrollo de los materiales educativos, así como los eventos que ocurran durante la navegación de los estudiantes en el LMS. Las áreas de interés más importantes de ADL/SCORM son las siguientes: accesibilidad, adaptabilidad, durabilidad, interoperabilidad, y reusabilidad. El SCORM de ADL es una colaboración de diversas especificaciones de múltiples instituciones entre las que destaca IMS GLC, buscando no competir entre sí, sino más bien buscando una colaboración más estrecha entre diferentes instituciones [10].

La Web semántica utiliza tecnologías que permiten a las aplicaciones procesar la información de manera semiautomática y automática. El modelo de conocimientos de un software es otro punto clave, su desacoplamiento es un punto crítico para posibilitar la reutilización y compartición entre aplicaciones. La Web semántica permite la utilización de ontologías y reglas para mover la lógica de negocios de las aplicaciones al modelo de conocimiento del usuario.

La presente propuesta presentara en la primera parte la propuesta de la arquitectura basada en componentes de software y agentes de software para el desarrollo de materiales educativos didácticos Web. La segunda parte mostrara los resultados de la propuesta. Por último se mostrara la sección de conclusiones de la propuesta.

3 Metodología usada

La organización del contenido SCORM (<organization>) es una representación estructurada compuesta con unidades de instrucción, que define la estructura del curso. La estructura jerárquica del contenido está estructurado por medio de un árbol de ítems (<item>) anidados, únicamente los ítem sin hijos permiten referenciar recursos. La etiqueta recurso (<resource>) es una colección de referencias a recursos. El activo (asset) es un bloque de construcción de los recursos de aprendizaje, un activo puede ser una colección de otros activos. Un Objeto de Contenido Compartible (Sharable Content Object, SCO por sus siglas en inglés) es una colección de uno o más activos que representa un recurso de aprendizaje desplegable por el RTE basado en SCORM que se comunica con el LMS.

En la parte izquierda de la Figura 1 se muestra el diagrama de despliegue de la propuesta basada en el Lenguaje de Modelado Unificado (*Unified Modeling Language*, UML por sus siglas en inglés), la nueva propuesta de los componentes ha sustituido el canvas de Flash y ActionScript 3.0 utilizados en propuestas anteriores, por el canvas de HTML 5 y JavaScript, para la programación de los Componentes Orientados a Objetos de Aprendizaje Reusables Inteligentes (*Intelligent Reusable Learning Components Object Oriented*, por sus siglas en inglés IRLCOO) [12-15], para la sustitución de los multimedios de la plataforma Flash, se utilizaron las etiquetas multimedia de HTML 5: video, imágenes y audio, los patrones de diseño de software patrón comando y método plantilla fueron reescritos desde ActionScript a JavaScript con las limitaciones que implican que este no sea un lenguaje completo bajo el paradigma de Programación Orientada a Objetos (*Object Oriented Programming*, OOP por sus siglas en inglés). Los cambios anteriormente mencionados han permitido eliminar la dependencia de contexto del plug-in Flash Player de nuestras propuestas previas, logrando que la nueva propuesta soporte estándares Web respaldados por el Consorcio Web Mundial (*World Wide Web Consortium*, W3C por sus siglas en inglés) [16].

En propuestas previas se usó la dependencia de contexto Flash Player para programar los IRLCOO, Flash era un integrador de multimedios, que nos permitió construir interfaces de usuario avanzadas y ricas en multimedios, pero su ventaja principal era que del lado del cliente podíamos programar en un lenguaje de OOP denominado ActionScript 3.0, lo que facilitó la implementación de los patrones de diseño de software. JavaScript no es actualmente un lenguaje completo de OOP, afortunadamente esto ha comenzado a cambiar con la liberación de ECMA Script 6 [17], siendo Google Chrome el navegador Web que está implementando más rápidamente ECMA Script 6, pero hay que esperar todavía para que JavaScript sea un verdadero lenguaje de OOP. El principal patrón del lado del cliente es el de composición, ya que es uno de los fundamentos de la Programación Orientada a Componentes (*Component-Oriented Programming*, COP), la principal ventaja del patrón es que nos permite construir componentes complejos en base a componentes más simples, en la parte izquierda de la Figura 1 se muestra los contenidos educativos didácticos divididos en los siguientes componentes: contenedor, múltiples contenidos/evaluaciones y navegación; esto con el objetivo de separar el contenido de la navegación de los materiales educativos, y maximizar la reutilización de los mismos; además de posibilitar composición y secuenciación dinámica en tiempo de ejecución.

En propuestas anteriores además del patrón composición del lado del cliente se utilizaron los siguientes patrones de diseño de software: método de fábrica, decorador, comando y método plantilla [12-15]. Pero con la muerte de Flash anunciada para 2020 [18], y la liberación del Lenguaje de Marcado de Hiper Texto (*Hyper Text Markup Language*, HTML por sus siglas en inglés) en su versión 5.0 [19], nos obligó a movernos a estándares

Web soportados por el W3C. Resultando que en la nueva propuesta solo se hayan implementado los siguientes patrones de diseño de software: composición, comando y método plantilla. Por lo anterior estamos en una búsqueda constante de nuevas arquitecturas basada en los estándares Web soportados por el W3C como: HTML 5, Lenguaje de Marcado Extensible (*eXtensible Markup Language*, XML por sus siglas en inglés), Hojas de Estilo en Cascada (*Cascading Style Sheets*, CSS por sus siglas en inglés), JavaScript, etc.

En la parte derecha de la Figura 1 muestra la arquitectura de la propuesta del lado del servidor, el principal patrón es el Modelo Vista Controlador (*Model View Controller*, MVC por sus siglas en inglés), lo que permite maximizar la reutilización de las partes del proyecto, y mejorar el mantenimiento, la implementación de la propuesta utilizó Struts 2 [20]. La propuesta utiliza Hibernate [21] para la parte Mapeado Objeto Relacional (*Object-Relational Mapping*, ORM por sus siglas en inglés). La parte del modelo de la propuesta está dividida en dos partes: base de datos y base de conocimientos. La parte de la base de datos esta implementada utilizando MySQL[22], la capa de persistencia de la aplicación permite almacenar y recuperar información de la aplicación en tiempo de ejecución, al utilizar ORM posibilita escribir la lógica de negocios en un modelo de OOP, lo que permite mejorar el mantenimiento. La base de conocimientos complementa la capa de persistencia de la propuesta, basada en el modelo IEEE LTSC 1484 [23], la base de conocimientos se implementó utilizando el framework Jena [24], que cuenta con un motor de inferencias y el lenguaje de consultas se denomina SPARQL, la base de conocimientos posibilita a la aplicación procesar la información de manera automática, utilizando en nuestro caso la ontología Amigo de un Amigo (*Friend Of A Friend*, FOAF por sus siglas en inglés) [25], para establecer las relaciones de los usuarios de la propuesta. Por último en la Figura 1 muestra el módulo del Sistema Multi-Agente (Multi-Agent System, MAS) para el procesamiento automático de la información de la propuesta, liberando a los profesores de muchas tareas repetitivas [15].

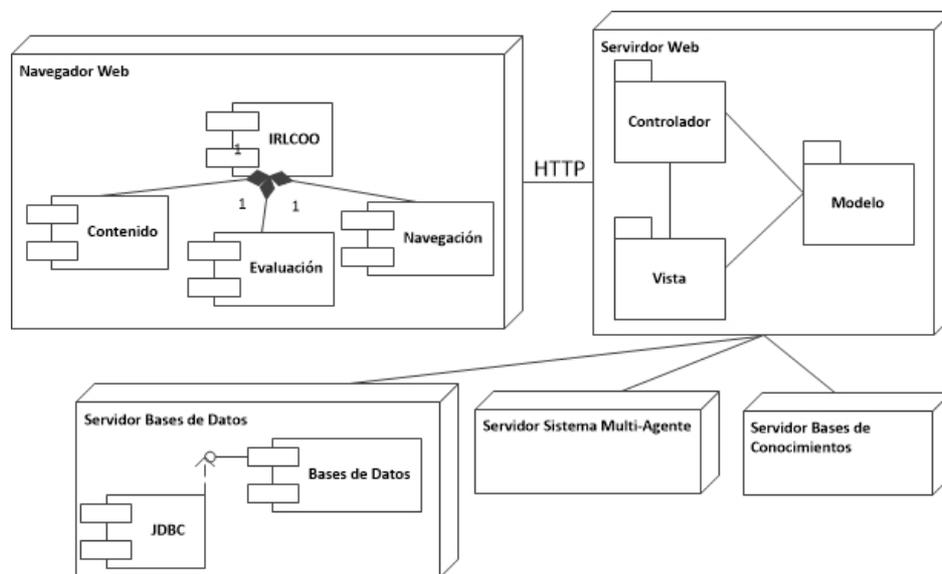


Figura 1. Diagrama de despliegue UML de la propuesta.

4 Resultados experimentales

La Figura 2 muestra una aplicación Web del lado del cliente construida bajo estándares Web, la de la izquierda es un editor Web de diagramas Entidad-Relación versión 1.0 para base de datos, la aplicación tiene una dependencia de contexto llamada joint.js, este marco de trabajo tiene una gran cantidad de componentes para graficar diferentes tipos de diagramas, en nuestro caso particular utilizamos los diagramas Entidad-Relación para base de datos. La aplicación Web implementa el patrón MVC, y es una herramienta de apoyo para desarrollar diagramas editables Entidad-Relación de base de datos, desarrollada bajo la arquitectura de la propuesta y basada en estándares Web, basada en el modelo de componentes mostrados en la presente propuesta, para apoyo a los docentes del área de base de datos o afines. Un aspecto principal que hay que destacar de la aplicación hasta el momento, es la migración de tecnologías propietarias que fueron utilizadas en propuestas previas como la plataforma Flash, y su plug-in Flash Player, hacia estándares HTML 5 y el uso del canvas, que utiliza esta nueva herramienta de apoyo, logrando reducir las dependencias de contexto de la propuesta, estando trabajando en la versión 2.0 actualmente, faltando todavía por construir un conjunto de servicios de apoyo adicionales para el docente. En la Figura 2 en el lado derecho se muestra una aplicación Web del lado del cliente construida bajo

estándares Web, siendo un editor Web de ecuaciones cuadráticas versión 1.0, en este caso este no tienen alguna dependencia de contexto explícita, la aplicación implementa el patrón MVC, y es una herramienta de apoyo para desarrollar ejercicios didácticos de ecuaciones cuadráticas, desarrollado bajo la arquitectura de la propuesta y basada en estándares Web, basada en el modelo de componentes mostrados en la presente propuesta, para apoyo a los docentes del área de matemáticas o afines.

Actualmente nos encontramos desarrollando otras herramientas de apoyo basadas en la arquitectura que muestra la presente propuesta, buscando disminuir la elevada complejidad técnica para desarrollar materiales educativos Web, que simplifiquen el trabajo del profesor.

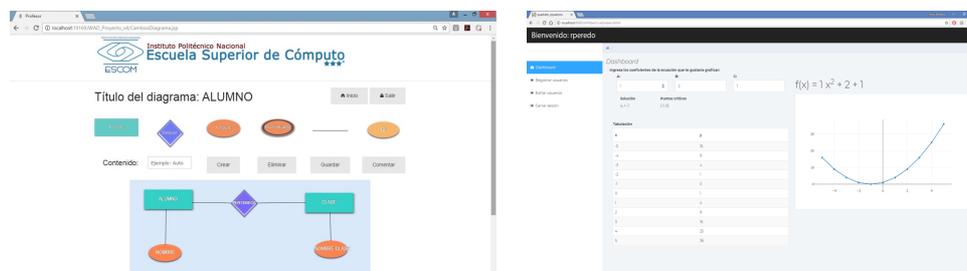


Figura 2. Aplicaciones Web del lado del cliente construidos bajo estándares Web.

5 Conclusiones y Trabajo Futuro

Es importante resaltar que la presente propuesta está eliminando dependencias de contextos propietarios, por tecnologías y estándares abiertos para la Web controlados por el W3C como: HTML 5, XML, CSS, y JavaScript, etc., buscando mayor compatibilidad y permitiendo que los materiales educativos puedan desplegarse en cualquier navegador Web que soporte los estándares Web liberados por el W3C.

La presente propuesta muestra un nuevo avance de un modelo basado en componentes de software innovador, que nos permitió seccionar las partes de los contenidos en tres subcomponentes reconfigurables principales: contenedor, múltiples contenidos/evaluaciones y navegación, que busca desarrollar materiales educativos didácticos, diferenciándose de propuestas anteriores en una migración de tecnologías propietarias, hacia estándares Web controlados por el W3C, el desarrollo de materiales educativos es complejo como se puede ver en la presente propuesta, y para un profesor promedio son tareas monumentales, al ser editables los materiales educativos interactivos que desarrolla el profesor pueden ser modificables por los estudiantes, e incluso pueden ser ejercicios dinámicos para los mismos estudiantes, ya que las herramientas pueden presentarse a los estudiantes, y editarlo como ejercicio dinámico interactivo, almacenando los resultados de los estudiantes, para que el profesor posteriormente puede evaluar de manera manual, o de manera automática, esta última con una respuesta concreta o por medio del MAS de una manera más compleja; lo cual ofrece prácticas más personalizadas para los estudiantes y profesores. La arquitectura basada en componentes de software en conjunto con los patrones de diseño de software, reducen la complejidad del desarrollo de materiales educativos avanzados, un punto clave es que los materiales educativos basados en componentes son auto contenidos, y configurados con archivos XML, lo que significa que pueden reconfigurarse, en cuanto a contenidos, y en cuanto a la secuencia a seguir, permitiendo adaptar los materiales educativos a nuevos escenarios y modelos educativos. Los componentes pueden reconfigurarse en tiempo de ejecución para poder utilizarse en diversos escenarios escolares, y bajo diferentes modelos pedagógicos. Se está comenzado a trabajar con ontologías para una mejor integración de las metodologías de aprendizaje.

La capa de persistencia está dividida en dos secciones, la primera basada en una base de datos y la segunda una bases de conocimiento, esta última utilizando la ontología FOAF, para procesar la información de manera automática, una ventaja fundamental de utilizar tecnologías de Web semántica consiste en el desacoplamiento del modelo de conocimientos de la aplicación, permitiendo la reutilización de este por otras aplicaciones. Ambas capas de persistencia tienen como objetivo una personalización más enriquecedora para los estudiantes.

Agradecimientos

Los autores de este artículo agradecen a la Universidad Politécnica de Querétaro, al Instituto Politécnico Nacional (IPN) por su apoyo para este trabajo dentro del proyecto SIP: 20182206 y a la Escuela Superior de Cómputo (ESCOM) por su apoyo para este trabajo. Los autores desean reconocer a todos sus colegas y estudiantes que participaron en el diseño y desarrollo del curso descrito en este artículo.

Referencias

- [1] Diccionario de la lengua española - Edición del Tricentenario <http://dle.rae.es/?id=3dyUvi4>
- [2] Qian, K., Fu, X., Tao, L., Xu, C., Díaz, J., Software Architecture and Design Illuminated (2010) Missussauga, Ontario, Canada.
- [3] ActiveX Controls: VBX Control Migration, URL: [https://msdn.microsoft.com/enus/library/aa268983\(v=vs.60\).aspx](https://msdn.microsoft.com/enus/library/aa268983(v=vs.60).aspx).
- [4] Welcome To CORBA Web Site!, URL: <http://www.corba.org/>.
- [5] COM: Component Object Model Technologies, URL: <https://www.microsoft.com/com/default.mspx>.
- [6] Distributed Component Object Model, URL: <https://technet.microsoft.com/en-us/library/cc958799.aspx>.
- [7] ActiveX Controls (Internet Explorer), URL: [https://msdn.microsoft.com/en-us/library/aa751968\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/aa751968(v=vs.85).aspx).
- [8] Trail: JavaBeans(TM), URL: <https://docs.oracle.com/javase/tutorial/javabeans/>.
- [9] Clemens Szyperski, Component Software: Beyond Object-Oriented Programming (2002) Boston, MA, USA.
- [10] Advanced Distributed Learning Initiative, URL: <http://www.adlnet.org>.
- [11] Global IMS Learning Consortium, URL: <http://www.imsproject.org/>.
- [12] Peredo, R., Balladares, L., Sheremetov, L.: Development of intelligent reusable learning objects for Web-based education systems. Expert Systems with Applications. 28(2). (2005) 273-283.
- [13] Canales, A., Peña, A., Peredo, R., Sossa, H., Gutiérrez, A.: Adaptive and intelligent Web based education system: Towards an integral architecture and framework. Expert Systems with Applications. 33(4). (2007) 1076-1089.
- [14] Peredo, R., et al. Intelligent Web-based education system for adaptive learning. Expert Systems with Applications (2011), doi:10.1016/j.ESWA.2011.05.013.
- [15] Rubén Peredo Valderrama, Alejandro Canales Cruz, Adriana N. Ramírez Salazar, Juan Carlos Caravantes Ramírez: Personalized knowledge management in environments of Web-based education. International Journal of Information Technology & Decision Making. Vol. 12, No. 2 (2013) 277-307.
- [16] W3C, URL: <http://www.w3.org/>.
- [17] ECMAScript 6: New Features: Overview and Comparison, URL: <http://es6-features.org/>.
- [18] Adobe anuncia la muerte de Flash, URL: <http://www.eluniversal.com.mx/articulo/techbit/2017/07/26/adobe-anunciala-muerte-de-flash>
- [19] HTML 5, URL: <https://www.w3.org/TR/html5/>
- [20] Struts 2 - The Apache Software Foundation!, URL: <http://struts.apache.org/>.
- [21] Hibernate. Everything data. – Hibernate, URL: <http://hibernate.org/>.
- [22] MySQL, URL: <https://www.mysql.com/>.
- [23] IEEE Learning Technology Standards Committee, URL: <http://ieeesa.centraldesktop.com/ltsc/>.
- [24] Apache Jena - Home, URL: <http://jena.apache.org>.
- [25] The Friend of a Friend (FOAF) project | FOAF project, URL: <http://www.foaf-project.org/>.