



SISTEMA AUTOMATIZADO DE INTEGRACIÓN DE ARQUITECTURA DE LA INFORMACIÓN EN EL DESARROLLO DE APLICACIONES WEB INTERACTIVAS



Luis-Alberto Rojas-Pino y José-Antonio Macías-Iglesias



Luis-Alberto Rojas-Pino es estudiante del máster en ingeniería informática y de telecomunicación de la *Universidad Autónoma de Madrid*, donde realiza una estancia de estudio e investigación en la *Escuela Politécnica Superior*. Desde 2001 trabaja en el *Departamento de Ingeniería de Sistemas* en la *Comisión Chilena de Energía Nuclear* en la implementación de tecnologías de información en procesos productivos, de gestión y de investigación.

*Universidad Autónoma de Madrid
Escuela Politécnica Superior
Tomás y Valiente, 11. 28049 Madrid, España
luisalberto.rojas@estudiante.uam.es*



José-Antonio Macías-Iglesias es doctor en ingeniería informática y profesor en el *Departamento de Ingeniería Informática* de la *Universidad Autónoma de Madrid*. Su línea principal de investigación es la interacción persona-ordenador. Es vicepresidente de *AIPO (Asociación Interacción Persona-Ordenador)* y Co-Chair del *SIGCHI (Grupo de Interés en Interacción Persona-Ordenador)* de ACM en España. Ha participado en diferentes proyectos de interacción persona-ordenador e ingeniería del software.

*Universidad Autónoma de Madrid
Escuela Politécnica Superior
Tomás y Valiente, 11. 28049 Madrid, España
j.macias@uam.es*

Resumen

Los arquitectos de la información trabajan principalmente en el modelado conceptual y en la concreción de aspectos no funcionales de aplicaciones web interactivas dentro de las primeras fases del ciclo de vida de un proyecto software. La información es procesada posteriormente por ingenieros del software para diseñar la aplicación final. Este flujo de información es difícil de automatizar en algunas ocasiones, debido a la diferencia conceptual entre la información procesada por cada experto. Para facilitar el flujo de información se propone una aplicación llamada *InterArch*, que permite establecer puentes entre la información que procesan ambas profesiones, creando de forma automática información de análisis y diseño para ingenieros del software a partir de los modelos de contenidos creados por profesionales de la información.

Palabras claves

Arquitectura de la información, Modelado del contenido, Interacción persona-ordenador, Desarrollo por el usuario final, Análisis y diseño de sistemas web interactivos.

Title: An automated solution to integrating information architecture into the interactive web-application development process

Abstract

Information architects are mostly focused on the conceptual modeling and analysis of non-functional aspects in the early stages of interactive web projects. The resulting output is then processed by software engineers to design the needed web interface. However, this flow of information is somehow difficult to automate due to the implicit differences in the conceptual level of information processed by each expert. In order to facilitate this flow of information we propose a tool called *Interarch*, which helps to establish bridges between the information processed by both professionals, automatically creating analysis and design information for software engineers from the content models created by the information professionals.

Keywords

Information Architecture, Content Modeling, Human-computer interaction, End-user development, Analysis and design of interactive web systems.

Artículo recibido el 05-10-11
Aceptación definitiva: 07-12-11

Rojas-Pino, Luis-Alberto; Macías-Iglesias, José-Antonio. "Sistema automatizado de integración de arquitectura de la información en el desarrollo de aplicaciones web interactivas". *El profesional de la información*, 2012, marzo-abril, v. 21, n. 2, pp. 160-166.

<http://dx.doi.org/10.3145/epi.2012.mar.06>

1. Introducción

Los equipos de programación de aplicaciones web están formados por profesionales de diversas áreas. Entre ellos los arquitectos de la información trabajan en el modelado conceptual y en la concreción de aspectos no funcionales de una aplicación web interactiva, y tratan de enlazar las ideas conceptuales que surgen en las fases iniciales con la información de diseño que posteriormente necesitarán los ingenieros del software.

Los roles de arquitecto de la información y de ingeniero del software no suelen coincidir, ya que el primero tiene un perfil menos técnico y más orientado a tareas conceptuales, lo cual dificulta el trasvase de información entre ambos. Si este trasvase se pudiera hacer de forma automática, el tiempo y esfuerzo de realización del proyecto se reducirían, permitiendo a cada experto concentrarse en su trabajo en función de sus conocimientos.

Los programas más utilizados por los arquitectos de la información intentan conectar la información que ellos crean con la que esperan los ingenieros del software, mediante imágenes y páginas web en distintos formatos (Pérez-Montoro; Codina, 2010). Este sistema elimina detalles semánticos importantes relativos al análisis, y dificulta la interoperabilidad en la elaboración de aplicaciones web. Ninguno de estos programas permite obtener de forma automática o semiautomática la información de análisis y diseño requerida por los ingenieros del software a partir del análisis conceptual del arquitecto de la información.

1.1. Objetivos de la investigación

El presente trabajo tiene dos objetivos genéricos y varios concretos. Los genéricos son:

- 1) Generar información de análisis y diseño procesable por ingenieros del software a partir de la representación inicial conceptual de los contenidos creados por el arquitecto de la información.
- 2) Sistematizar ese proceso mediante la construcción de una aplicación de fácil uso y aprendizaje por parte del arquitecto de la información, que permita generar automáticamente información de análisis y diseño para ingenieros del software, estableciendo puentes entre las fases iniciales del proyecto (dentro del "dominio del problema"), y las fases técnicas del proyecto (en el "dominio de la solución").

Los objetivos concretos son:

- a) Facilitar al arquitecto de la información la elaboración de sus productos, ayudándole a concentrarse en las tareas de análisis conceptual dentro del dominio del problema.
- b) Proporcionar información de análisis y diseño para ingenieros del software, que garantice la generación y gestión del conocimiento dentro del dominio de la solución.

- c) Validar la generación de información de análisis y diseño mediante una aplicación que cumpla unos requisitos de utilidad, satisfacción, facilidad de uso y aprendizaje.
- d) Aumentar la automatización e interoperabilidad entre las definiciones conceptuales de los contenidos de una aplicación web y los elementos de análisis y diseño requeridos para el proceso de elaboración.

2. Gestión y transformación de modelos de la arquitectura de la información

Es difícil estipular los límites operacionales de la arquitectura de la información (IA), lo que hace necesario muchas veces el uso de diversos tipos de aplicaciones y estándares. No obstante se puede hacer un resumen de los productos más comunes que el profesional de la información debe crear para el análisis de la IA de una aplicación web interactiva. Los más importantes son *blueprints* (planos), *wireframes* (maquetas), modelos de contenido y vocabularios controlados (Erlin; Yanura; Rahman, 2008; Morville; Rosenfeld, 2006). Estos productos pueden representar una cantidad de conocimiento importante, procedente de los diferentes profesionales que participan en proyectos de construcción de sitios web (Pérez-Montoro, 2010), y se hace indispensable compartirlos en diferentes formatos y plataformas para su posterior utilización por los demás miembros del equipo de trabajo.

Sin embargo los modelos de contenido son especialmente trascendentales para ingenieros del software, ya que contienen una representación explícita de aspectos no funcionales de la aplicación en forma de contenidos de información que deben ser procesados posteriormente junto con los elementos funcionales de la aplicación web. Por otro lado, los modelos de contenido son susceptibles de un tratamiento automático para generar de forma directa los diagramas de clases y elementos de contenido que definirán la aplicación en el dominio de la solución.

Es la razón por la cual el presente trabajo se concentra en estos elementos esenciales de cara a una automatización de las salidas del proceso de análisis de la IA y se plantea el diseño, desarrollo y evaluación de una aplicación denominada *InterArch* (Rojas-Pino, 2011).

2.1. InterArch

Se compone de una serie de procesos de gestión y transformación de modelos en un entorno visual orientado al arquitecto de la información, que son:

- modelado visual de los elementos conceptuales;
- transformación del modelo visual en un modelo intermedio; y
- generación textual transportable en diagramas de clases UML (*unified modeling language*) en formato XMI (xml para el intercambio de metadatos) (figura 1).

Estos procesos tienen por objetivo tomar como entrada el diseño visual de contenidos realizado por el arquitecto de la información y generar como salida diagramas de clases para el ingeniero del software.

InterArch permite manipular, formatear y relacionar los elementos visuales de contenido para la elaboración de diagramas de la IA, facilitando al profesional de la información la realización de modelados visuales (figura 2).

a) Entorno para la construcción de diagramas para la IA

En la parte C de la figura 2 se muestran los iconos de trabajo que permiten enriquecer la interacción visual y construcción de los diagramas elaborados por el arquitecto de la información. Existen principalmente dos tipos de elementos para el modelado visual:

- de interpretación de contenido del modelado visual-conceptual de la IA (tres primeras filas de iconos);
- de enlace, que permiten definir tipos de asociaciones y relaciones entre elementos de contenido (dos últimas filas).

Todos tienen por objetivo facilitar la aplicación de reglas de transformación para obtener una generación adecuada de diagramas de clases.

b) Entorno de trabajo visual

La parte B de la figura 2 permite manipular y relacionar los distintos elementos visuales de la aplicación. En este ejemplo se pueden apreciar las relaciones entre diferentes elementos de contenido, que describen la estructura y atributos de los anuncios en una web de venta de productos de segunda mano.

c) Formato y edición de componentes

En la parte A de la figura 2 se muestran las opciones de formato y edición para manipular los elementos visuales dentro del entorno de trabajo. Estas opciones permiten configurar cada elemento visual tanto de forma individual como grupal, así como definir y manipular los diagramas.

2.2. Reglas de transformación entre modelos de distinto nivel conceptual

InterArch incorpora una capa de interpretación compuesta por un conjunto de reglas de transformación que actúan sobre la asociación y jerarquía de los modelos de contenido elaborado por los profesionales de la información. Se clasifican en reglas de jerarquía y de configuración.

a) Reglas de jerarquía

Criterios de transformación que actúan sobre la jerarquía de los elementos dentro de los diagramas de modelado del contenido, y permiten identificar clases, atributos, métodos y asociaciones de los diagramas de clases generados. Se aplican para cada elemento del contenido, analizando cada caso para ir realizando las correspondientes transformaciones:

R₁: Los elementos que contienen elementos descendientes son considerados directamente como clases en el diagrama de clases.

R₂: Los elementos descendientes se considerarán atributos que se incluirán en los elementos de los que descendien.

R₃: El elemento principal del diagrama del modelado de contenido se corresponde con la clase raíz en el diagrama de clases.

R₄: Los elementos descendientes correspondientes a nuevas clases generan una asociación directa con el elemento del que descendien.

R₅: Por cada asociación generada en el diagrama de clases se crean tres métodos (*get*, *set* y *new*) en la clase donde se inicia la asociación.

La figura 3 muestra el diagrama de clases UML resultante después de la aplicación de las reglas anteriores sobre el ejemplo de modelo de contenido de la figura 2 (parte B). A partir de la aplicación de las reglas, surgen varias clases: *RegistroDeAnuncio*, *DatosDelAnuncio*, *PrecioDeVenta* y *DatosDelAnunciante*, con sus atributos y métodos de creación y manipulación asociados. El diagrama resultante se puede generar con mayor nivel de especificación mediante las reglas de configuración, lo que permite la creación de clases no funcionales de diseño para ser incorporadas posteriormente al proyecto.

b) Reglas de configuración

Propiedades específicas del nivel de visibilidad, acceso y navegabilidad de las clases, atributos, métodos y asociaciones de los diagramas de clases. Estas reglas se con-

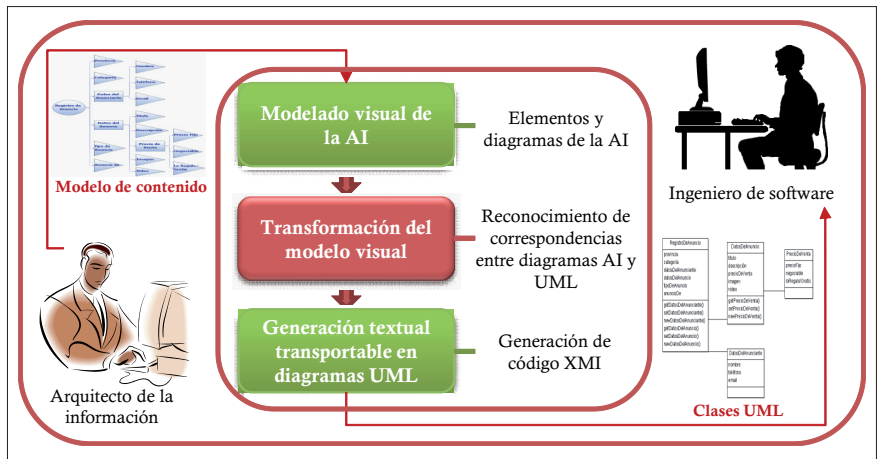


Figura 1. Esquema modular de *InterArch*

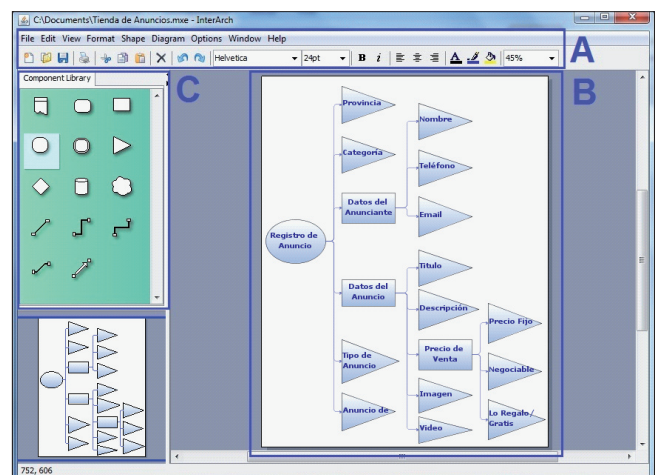


Figura 2. Interfaz de usuario de *InterArch* dividida en tres partes: A, B y C

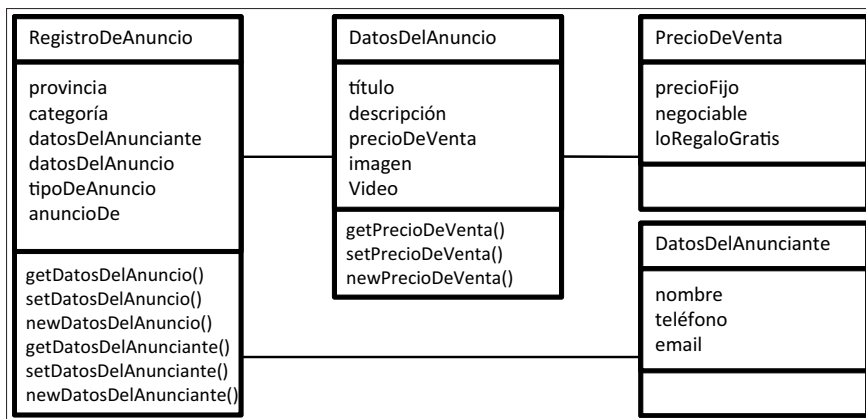


Figura 3. Diagrama de clases resultante de la transformación del modelo de contenido

centran en los elementos que componen los diagramas de clases y las diferentes propiedades que ellos pueden contener.

El conjunto de reglas de configuración se aplica al modelo de contenido elaborado inicialmente por el arquitecto de la información, lo que posibilita concretar mucho más la generación final del diagrama de clases resultante. En la figura 4 se muestra el módulo de *InterArch* que gestiona las reglas de configuración, el cual permite especificar y ajustar las distintas propiedades de las etiquetas de los elementos de los diagramas de clases de acuerdo con las necesidades específicas de los ingenieros del software. En la figura se puede apreciar el conjunto de reglas seleccionadas de forma predeterminada y que serán aplicadas por defecto, las cuales se agrupan en reglas de clase, atributo, método y asociación.

Estas reglas de configuración se corresponden con los tipos de etiquetas propuestos por la *OMG* (2005) para adaptar los esquemas y documentos que se producen utilizando XML (*XML metadata interchange*, una versión etiquetada e interoperable del lenguaje de modelado UML, *unified modeling language*).

La información generada por la aplicación puede ser adaptada a las opciones de configuración de *InterArch*. También se puede ajustar la información de análisis y diseño a partir de requerimientos más específicos por parte de los ingenieros del software, mediante la importación del archivo XML en otras herramientas de modelado. Esto permite además aprovechar otras ventajas con las que cuentan estas aplicaciones, como son la ingeniería inversa de código fuente, la incorporación de bases de datos y las restricciones OCL (*object constraint language*, lenguaje formal para expresar restricciones) entre otras, aportando un valor añadido.

Esta implementación de la metodología de gestión de conocimiento conceptual en el dominio del problema, y su posterior transformación en un modelo cercano al dominio de la solución, permite validar parte de los planteamientos iniciales, en concreto el primer objetivo de esta investigación.

3. Evaluación de la usabilidad de *InterArch*

Para evaluar la usabilidad de la aplicación se ha utilizado un test con usuarios, con los protocolos de *Análisis retrospectivo* y *Thinking aloud* (pensar en voz alta):

El *Análisis retrospectivo* consiste en revisar los registros de vídeos almacenados durante la realización del test, lo que facilita recopilar información adicional (*Nielsen*, 1993).

El protocolo *Thinking aloud* consiste en pedir al usuario final que vaya comen-

tando en voz alta mientras interactúa con el sistema. Al verbalizar los usuarios sus pensamientos, se entiende cómo ven el sistema, lo que hace que sea más fácil identificar conceptos erróneos (*Nielsen*, 1993).

Con estos dos protocolos se consigue medir y analizar detenidamente a posteriori la interacción del usuario con la aplicación, reflejando de forma detallada aspectos como la forma en que resolvían sus tareas o los tiempos empleados en cada una.

Se contó con 12 personas que trabajan de forma habitual en empresas y consultoras de proyectos informáticos, específicamente en tareas concernientes a IA, 9 hombres y 3 mujeres, con edades comprendidas entre 24 y 43 años (media = 32, desviación estándar = 8,062). En general, los usuarios contaban con conocimientos sobre análisis y documentación en la estructuración y categorización de contenidos de sitios web.

Se llevaron a cabo los siguientes pasos:

- Demostración de las funciones y características de *InterArch*, con una duración promedio de 7,5 minutos.
- Se solicitó a los usuarios elaborar un modelo de contenido relacionado con el proceso real de creación de registros de anuncios de una empresa de anuncios de productos de segunda mano. En concreto, se dispuso de una “maqueta” sacada de uno de los anuncios de la versión impresa de la revista *Segunda mano* y se solicitó crear el modelo de contenido utilizando *InterArch*. La experiencia de ela-

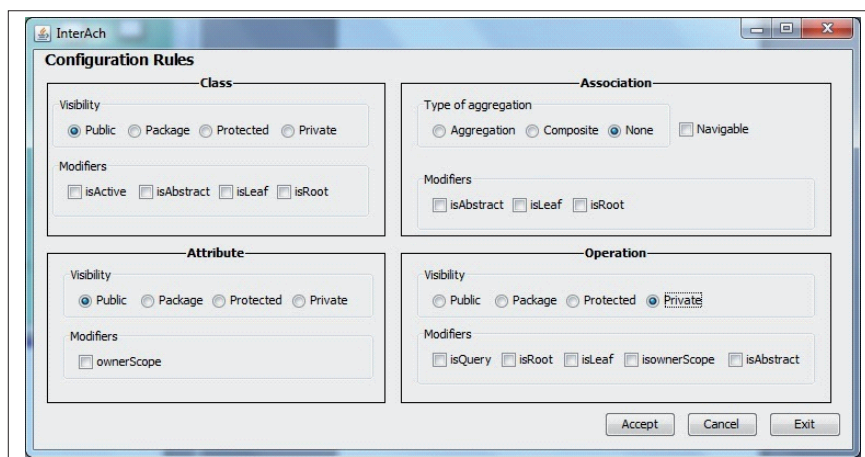


Figura 4. Módulo de reglas de configuración de *InterArch*

boración de este modelo tuvo una duración promedio de 12 minutos.

Con *Thinking aloud* se recogió información valiosa sobre cómo los usuarios utilizan la aplicación, identificando conceptos erróneos en algunas opciones de los menús contextuales y elementos interactivos. De igual forma, el *Análisis retrospectivo* facilitó un estudio más exhaustivo de las interacciones con *InterArch* mediante la revisión de las grabaciones de audio y vídeo de los usuarios. Si bien no se encontraron errores graves durante la sesión experimental, los resultados obtenidos se utilizarán para mejorar la aplicación, de modo que en el futuro responda de forma más eficiente y en concordancia a las necesidades.

c) Se pidió a los usuarios completar un cuestionario para medir la usabilidad de *InterArch*. Se utilizó el cuestionario *USE* (Lund, 2001), con algunas variaciones proporcionadas por los cuestionarios de *Percepción de utilidad y facilidad de uso* de Davis (1989), y de *Usabilidad de Purdue* (Lin; Choong; Salvendy, 1997). El cuestionario contenía 31 preguntas cerradas para medir las variables de utilidad (8), facilidad de uso (10), facilidad de aprendizaje (6) y satisfacción (7), lo que se corresponde con las 4 dimensiones medidas para la estimación de la usabilidad. Cada pregunta se evaluó en una escala *Likert* entre 1 (mínimo) y 10 (máximo). Además se incluyeron 4 preguntas abiertas para obtener información sobre aspectos tanto positivos como negativos.

InterArch establece puentes entre las representaciones conceptuales de alto nivel de la AI y la representación no funcional e ingenieril del contenido

3.1. Análisis de la usabilidad y resultados

En la tabla 1 se muestran los resultados del cuestionario de evaluación, con los valores medios de las frecuencias medidas para cada variable. La variable *Facilidad de aprendizaje* es la que obtiene la puntuación promedio más alta, con una media de 8,3 y le siguen *Facilidad de uso* (7,9), *Utilidad* (7,5) y *Satisfacción* (7,4).

| | Utilidad | Facilidad de uso | Facilidad de aprendizaje | Satisfacción | Media |
|---------------------|----------|------------------|--------------------------|--------------|-------|
| Frecuencia | 7,5 | 7,9 | 8,3 | 7,4 | 7,8 |
| Desviación estándar | 1,09 | 1,08 | 1,23 | 1,06 | 1,14 |

Tabla 1. Resultado de la evaluación de la usabilidad de *InterArch*

El promedio de las cuatro variables es de 7,8, con una desviación estándar de 1,14. Todas las variables obtienen puntuaciones superiores a 7, con lo cual esta media se puede considerar un buen indicador de la usabilidad general de la aplicación en base a la percepción de los usuarios.

3.2. Fiabilidad de la evaluación

La fiabilidad es el nivel de consistencia de una determinada medición, o sea, si se hallan resultados similares en repe-

tidas mediciones. Para medir la fiabilidad de la evaluación realizada a los 12 usuarios se ha utilizado el *indicador Alfa de Cronbach*, calculándolo para las 31 preguntas cerradas del cuestionario. El resultado muestra un valor de fiabilidad del 95,2% ($\alpha = 0,95$), lo cual indica que el cuestionario tiene un nivel de fiabilidad excelente, pues supera en un 25,18% el umbral de aceptación. El recíproco del *Alfa* se utilizará como nivel de confianza para efectuar el contraste de hipótesis en el análisis de la varianza (*1-alfa de Cronbach*; $\alpha = 0,05$).

3.3. Análisis multivariable

Adicionalmente se estudió la posible variabilidad de los valores medios de las variables mediante un *análisis de la varianza (anova)* de un factor sobre las 4 variables estudiadas. Antes se analizó si las cuatro variables pueden ser adecuadamente modeladas mediante una distribución normal. Para realizar este análisis se utilizó el *test de Shapiro-Wilk*, obteniendo valores de significación de 0,85, 0,24, 0,15 y 0,23 para *Utilidad*, *Facilidad de uso*, *Facilidad de aprendizaje* y *Satisfacción* respectivamente. Según estos resultados se puede concluir que los datos experimentales siguen una distribución normal, debido a que cada variable tiene un valor de significación mayor a 0,05, correspondiente al nivel aceptado en base al cálculo del recíproco del *Alfa de Cronbach*. Para realizar el estudio del *anova* se planteó como hipótesis inicial que las medias de las 4 variables son similares. Este análisis consiste en poner a prueba la hipótesis inicial mediante el cálculo del estadístico $F_{test'}$ que indica la proporción entre los estimadores de la varianza poblacional sobre las medias que se están comparando. El resultado del cálculo del *anova* obtuvo un valor para el estadístico F_{test} de 2,16 y un nivel crítico F_{crit} de 2,85 ($\alpha = 0,05$; 3; 44; 1). Debido a que F_{crit} es mayor a $F_{test'}$ no es posible rechazar la hipótesis inicial, lo que permite indicar que no hay diferencias entre las 4 variables analizadas, es decir, tienen niveles similares de aceptación entre los usuarios.

Los resultados experimentales obtenidos corroboran que *InterArch* es fácil de usar y de aprender por parte del arquitecto de la información. Además permiten también afirmar que tanto el formalismo como la aplicación son un aporte importante a paradigmas de investigación relacionados con el desarrollo por el usuario final, cuyo objetivo es involucrar a usuarios no expertos en computación en el ciclo de vida del software.

4. Conclusiones y trabajo futuro

En este trabajo se ha presentado un sistema de información válido para el análisis y diseño de aplicaciones web consistente en un formalismo que permite representar el conocimiento proveniente de las definiciones conceptuales del arquitecto de la información, así como su posterior transformación en información de análisis y diseño para el ingeniero del software. Se ha materializado en una aplicación *CASE (computer-aided software engineering)* llamada *InterArch*, fácil de usar y de aprender por parte del profesional de la información.

InterArch se ha desarrollado con el objetivo de enlazar las representaciones conceptuales de alto nivel de la IA y la representación no funcional e ingenieril del contenido.

Con esto se obtienen finalmente clases de análisis y diseño necesarias para la implementación de software dentro del dominio de la solución. La aplicación genera automáticamente diagramas de clases a partir de la definición de modelos de contenidos de la IA de un sitio web interactivo, utilizando XMI como lenguaje intermedio de representación que además puede ser procesado por un gran número de herramientas de modelado CASE, lo que posibilita una mayor interoperabilidad para integrar clases funcionales y no funcionales durante el ciclo de vida de una aplicación web interactiva.

Los resultados de las evaluaciones de usabilidad de la aplicación muestran valoraciones positivas y aceptables de la percepción que tienen los usuarios sobre las variables *Utilidad*, *Facilidad de uso*, *Facilidad de aprendizaje* y *Satisfacción* de *InterArch*.

Una línea prometedora de trabajo futuro sería mejorar la aplicación con características de aporte semántico. Por ejemplo, la inclusión de comentarios del profesional de la información en los elementos de contenido, la dotaría de información semántica (Macías, 2008) para su posterior gestión por los ingenieros del software, de forma que se puedan crear restricciones más avanzadas para el dominio de la solución. Ello también posibilita la incorporación automática de propiedades explícitas de accesibilidad y usabilidad, que podrían incluirse de forma temprana en el análisis de contenidos del proyecto software.

5. Agradecimientos

La realización del presente trabajo ha sido posible gracias a la subvención del proyecto CCG10-UAM/TIC-5772, cofinanciado por la DGUI de la Comunidad de Madrid y la Universidad Autónoma de Madrid, y de los proyectos TIN2008-02081/TIN y TIN2011-24139, financiados por el Ministerio de Ciencia e Innovación.

6. Referencias

Davis, Fred. "Perceived usefulness, perceived ease of use, and user acceptance of information technology". *MIS quarterly*, 1989, v. 13, n. 3, pp. 319-340. <http://dx.doi.org/10.2307/249008>

Erlin; Yunus, Yunura; Rahman, Azizah. "The evolution of

information architecture". *Information technology*, ITSim, 2008, v. 4, pp. 1-6.

<http://dx.doi.org/10.1109/ITSIM.2008.4631890>

Lin, Han; Choong, Yee-Yin; Salvendy, Gavriel. "A proposed index of usability: a method for comparing the relative usability of different software systems". *Behaviour and information technology*, 1997, v. 16, n. 4/5, pp. 267-278.

<http://www.idemployee.id.tue.nl/g.w.m.rauterberg/bit-25/lin-et-al-1997.pdf>

<http://dx.doi.org/10.1080/014492997119833>

Lund, Arnold. "Measuring usability with the USE questionnaire". *Usability and user experience special interest group*, 2001, v. 8, n. 2.

http://www.stcsig.org/usability/newsletter/0110_masuring_with_use.html

Macías-Iglesias, José-Antonio. "Intelligent assistance in authoring dynamically-generated web interfaces". *World wide web – Internet and web information systems*, 2008, v. 11, n. 2, pp. 253-286.

<http://dx.doi.org/10.1007/s11280-008-0043-3>

Morville, Peter; Rosenfeld, Louis. *Information architecture for the world wide web*. O'Reilly Media Inc, 2006. ISBN: 978 0596527341

Nielsen, Jacob. *Usability engineering*. Morgan Kaufmann Publishers, 1993. ISBN: 0125184069

OMG. *MOF 2.0/XMI Mapping specification*, 2005.

<http://www.omg.org/spec/XMI/2.1/PDF>

Pérez-Montoro, Mario. *Arquitectura de la información en entornos web*. Gijón: Trea, 2010. ISBN: 978 8497045032

Pérez-Montoro, Mario; Codina, Lluís. "Software de prototipado para la arquitectura de la información: funcionalidad y evaluación". *El profesional de la información*, 2010, v. 19, n. 4, pp. 417-424.

<http://dx.doi.org/10.3145/epi.2010.jul.12>

Pressman, Roger. *Ingeniería del software: un enfoque práctico*. McGraw-Hill, 2005. ISBN: 9701054733

Rojas-Pino, Luis-Alberto. *Soporte CASE para el análisis de la arquitectura de la información en entornos de desarrollo web*. Trabajo fin de máster dirigido por José-Antonio Macías-Iglesias. *Escuela Politécnica Superior, Universidad Autónoma de Madrid*, octubre 2011.

Si te interesan los

INDICADORES EN CIENCIA Y TECNOLOGÍA,

y todos los temas relacionados con la medición de la ciencia, tales como:

Análisis de citas, Normalización de nombres e instituciones, Impacto de la ciencia en la sociedad, Indicadores, Sociología de la ciencia, Política científica, Comunicación de la ciencia, Revistas, Bases de datos, Índices de impacto, Políticas de open access, Análisis de la nueva economía, Mujer y ciencia, etc.

Entonces **INCYT** es tu lista. Suscríbete en:

<http://www.rediris.es/list/info/incyt.html>

Mi Biblioteca

Actualidad bibliotecaria

Fotografía momentos de lectura

Opinión

Recomendaciones de lectura

Escolares

Públicas

Reportaje

Entrevistas

Recursos Universitarias

Con pasado

Cartas a Mi Bil

Bibliotecarios

Con futuro

Noticias

Agenda de Formación

Nuevas Tecnologías

Gestión documental

www.mibiblioteca.org