

CARACTERISTICAS DE USABILIDAD EN WIS DESDE EL ENFOQUE MDA: UNA PERSPECTIVA DE USUARIO

Julio César Sierra González

Grupo de Investigación en Sistemas y Tecnologías de la Información y de la Comunicación en Organizaciones “GISTIC”

*Magister en Ingeniería de Sistemas y Computación
Universidad Nacional de Colombia
Bogotá, Colombia*

RESUMEN

El artículo presenta una propuesta de identificación de características de Sistemas de Información Basadas la Web (WIS) desde una perspectiva de usuario, usando los métodos de desarrollo de aplicaciones Web en el dominio de la Ingeniería Web Dirigida por Modelos (MDWE) en general y en el enfoque de la Arquitectura Dirigida por Modelos (MDA) en particular. Una vez identificados los componentes hipermedia de la arquitectura de un WIS (contenido, presentación y navegación), se efectúa una revisión bibliográfica para determinar las características del modelo de contenido en el dominio de Calidad de la Información (QI) y en basados en Abrahão *et al.* (2014) se derivan las características de los modelos de navegación y presentación. Finalmente se plantean las conclusiones de la investigación.

Palabras Clave

WIS, Ingeniería Web, Ingeniería Web Dirigida por Modelos, Arquitectura Dirigida por Modelos

1. INTRODUCCION

El uso de la Web ha crecido explosivamente en los últimos años, pasando de ser un medio de comunicación a un vehículo de difusión de información (Worwa, *et al.* 2011). Muchos Sistemas de Información tradicionales se están desarrollando y/o se han convertido en Sistemas de Información Basados en la Web (Web Based Information Systems, WIS), interactivos y presentes en numerosos dominios, convirtiéndose en sistemas críticos para las estrategias de negocio de muchas organizaciones.

A nivel de implementación, para garantizar el éxito de WIS no es suficiente con satisfacer sus requisitos funcionales, sino que debe lograrse una eficiente usabilidad como aspecto crítico de aceptación por parte una gran cantidad de diversos tipos de usuarios (Abrahão *et al.* 2014) que hacen uso de estas aplicaciones en forma concurrente y remota a través de navegadores (Levin, 2011). Por lo tanto, las tecnologías que apoyan la evaluación de usabilidad de este tipo de sistemas se han convertido en un factor importante para determinar y asegurar esta característica.

Las Interfaces de Usuario (User Interface, UI) de los WIS actuales son basadas en hipermedia, además de tener un alto nivel de complejidad funcionalidad (Abrahão *et al.* 2014), de plataforma de base tecnológica, de los dominios del sistema y externos, así como de modelado de personalización, ubicuidad y adaptación. Por lo tanto para desarrollar metodologías y procesos de evaluación de usabilidad de este tipo de sistemas es necesario remitirse a metodologías de desarrollo de la Ingeniería Web, que abordan en forma explícita los métodos y herramientas con los cuales resuelven los complejos requisitos, haciendo evidentes sus características, relaciones, funciones y prestaciones. En este contexto, la Ingeniería Dirigida por Modelos (Model Driven Engineering, MDE) en general y la Arquitectura Dirigida por Modelos (Model Driven Architecture, MDA) en particular están siendo usados para abordar con éxito la construcción, evolución y adaptación de aplicaciones Web (Koch *et al.*, 2008). Estos enfoques ofrecen metodologías y herramientas para el diseño y desarrollo de la mayoría de los tipos de aplicaciones Web, incluyendo los WIS, utilizando modelos que con el apoyo de compiladores producen la mayor parte de las páginas Web de una aplicación.

El objetivo del presente artículo es derivar de los modelos de desarrollo Web contemplados en MDA, tales como

OOHDM (Schwabe *et al.*, 1996), WebML (Ceri *et al.*, 2000), UWE (Hennicker & Koch 2001), y OO-H (Gómez *et al.*, 2001) entre otros, las características de usabilidad de un WIS como producto terminado y desde la perspectiva de usuario, con el fin de desarrollar una metodología de evaluación de usabilidad de WIS en producción, del cual no se tenga acceso a documentación ni a código fuente.

El resto del artículo está organizado como sigue: La sección 2 presenta la definición, categorías, evolución y una relación de la necesidad del enfoque de la Ingeniería Web en el desarrollo de WIS; la sección 3 presenta las características de la Ingeniería Web Dirigida por modelos (MDWE) y una breve descripción de los métodos más conocidos en ésta área, con el fin de determinar las características abordadas en el problema del diseño de WIS; la sección 4 presenta las características de los WIS desde la perspectiva de usuario, derivando su clasificación de la sección anterior y de una revisión de literatura; y finalmente la sección 5 presenta las conclusiones sobre la investigación realizada.

2. SISTEMAS DE INFORMACION BASADOS EN WEB

Un WIS (Isakowitz *et al.*, 1998) es un paradigma de sistemas de comunicación (Pokhrel *et al.*, 2010) que consiste en un sistema de información respaldado por base de datos, realizado y distribuido sobre la web, para ofrecer procesamiento interactivo de información y servicios a muchos usuarios que acceden en forma concurrente (Levin, 2011) y remota a través de navegadores web (Schewe *et al.* 2005; Fiedler *et al.* 2007). De esta forma, un WIS proporciona información en forma dinámica en múltiples formatos tales como texto e hipermedia, la cual queda disponible a través de páginas web, incluyendo la navegación entre ellas y sitios fuera del sistema.

2.1 Categorías de WIS

Actualmente, el alcance y la complejidad de los WIS es muy amplia, presentándose aplicaciones desde pequeños servicios de corta duración a aplicaciones corporativas a gran escala distribuidas a través de internet, intranets y extranets (Isakowitz *et al.*, 1998; Worwa *et al.*, 2010). Desde esta perspectiva operativa, los WIS se pueden agrupar en las siguientes categorías (Ginige y Murugesan, 2001; Kappel *et al.*, 2005):

Transaccionales. Se caracterizan por una alta interactividad, donde las bases de datos subyacentes permiten un eficiente y coherente manejo de datos y consultas estructuradas. Como ejemplo de este tipo de WIS, Kappel *et al.* (2005) plantea: banca en línea, compras en línea y sistemas de reserva entre otros.

Basados en el flujo de trabajo. Se caracterizan por permitir el manejo de flujos de trabajo dentro o entre diferentes compañías, autoridades públicas y usuarios privados, donde servicios Web apropiados garantizan la interoperabilidad. Presentan como requisito una cierta estructuración de procesos automatizados y operaciones, y como desafíos la complejidad de los servicios, autonomía de las empresas participantes y la necesidad de que los flujos de trabajo sean robustos y flexibles. Como ejemplo de este tipo de WIS, Kappel *et al.*, (2005) plantea: soluciones Business-to-business (B2B) en comercio electrónico, aplicaciones de administración electrónica en la administración pública y flujos de trabajo de pacientes en el sector salud.

Colaborativos. Se caracterizan al contrario de los WIS de flujo de trabajo de aplicaciones con fines de cooperación ("groupware") en operaciones estructuradas y donde la necesidad de comunicación es alta. Como ejemplo de este tipo de WIS, Kappel *et al.*, (2005) plantea: sistemas de soporte compartido de información y espacios de trabajo con el fin de generar, editar y gestionar información compartida tales como wikis o plataformas de e-learning.

Orientados a portales. Se caracterizan por proporcionar un único punto de acceso para fuentes heterogéneas de información y servicios. Como ejemplo de este tipo de WIS, Kappel *et al.*, (2005) plantea: portales comunitarios, mercados en forma de centros comerciales línea, portales empresariales tales como intranets o extranets, y motores de búsqueda (Detlor, 2000; Wang & Head, 2001).

Ubicuos. Se caracterizan por proporcionar servicios personalizados en cualquier momento, lugar y para cualquier dispositivo. Para lograr esta propiedad, se requiere conocimiento del contexto de uso del WIS, para que este realice en forma dinámica los ajustes correspondientes. Como ejemplo de este tipo de WIS, Kappel et al., (2005) plantea: personalización, servicios dependientes de la localización y entregas independientes del dispositivo.

Basados en conocimiento. Se caracterizan por presentar información en la Web para personas y en forma procesable para la máquina, gestionando el conocimiento con tecnologías de la Web semántica. Como ejemplo de este tipo de WIS, Kappel et al., (2005) plantea: vinculación y reutilización de conocimiento, y la minería Web para adquirir datos de negocio de los competidores.

Los WIS actuales son sistemas plenamente funcionales que proveen comercio electrónico empresa-a-cliente (business-to-customer), empresa-a-empresa (business-to-business) y muchos servicios para muchos usuarios. En lugar de referirnos a visitantes a los sitios Web, ahora nos referimos a usuarios, lo que implica mucha interacción.

2.2 Evolución de los WIS

El uso de la Web ha crecido explosivamente en los últimos años, pasando de ser un medio de comunicación (correo electrónico, transferencia de archivos, grupos de noticias y chat) a un vehículo de difusión de información (Worwa, et al. 2011). Las aplicaciones Web que una vez simplemente mostraban datos a los usuarios se han convertido en sistemas interactivos y altamente funcionales que permiten el procesamiento de información y muchos tipos de negocios. En consecuencia, los WIS ahora están presentes en numerosos dominios que son utilizados por millones de usuarios de todo el mundo y se han convertido en sistemas críticos para las estrategias de negocio de muchas organizaciones.

Algunos autores clasifican la evolución estas aplicaciones en etapas o generaciones (Binemann *et al.*, 2004; Taivalaari *et al.*, 2008; Worwa *et al.*, 2010): La primera generación, es el quiosco de información donde se presentan datos en forma no interactiva para fomentar la actividad de ventas convencional (Worwa *et al.*, 2010). La segunda generación, consiste en adicionarle al quiosco de información formularios Web, de tal forma que permita el registro de pedidos y consultas en forma interactiva, con lo cual el sitio Web resultante se comporta como un catálogo de ventas por correo electrónico. El problema presentado en los sitios Web de esta generación, es que la interfaz suministrada es difícil de aprender y no se ofrece ninguna ayuda en el uso. La tercera generación, consiste en dotar al sitio Web de la capacidad navegación y de procesamiento en línea, lo cual le permite al WIS presentar contenido actualizado, alto rendimiento, valor de marca del proveedor, uso agradable y facilidad de uso.

2.3 Ingeniería Web (Web Engineering, WE)

En la década de los 60s, el desarrollo de WIS se ha caracterizó por ser efectuado de manera ad hoc, careciendo de un enfoque sistemático, control de calidad y procedimientos de garantía, lo cual generó graves problemas para su desarrollo exitoso, implementación, operación y mantenimiento (Murugesan *et al.*, 2001; Kappel *et al.*, 2005, Srivastava *et al.*, 2014). Este problema metodológico de desarrollo de WIS, según Sharma *et al.*, (2014) persiste en la actualidad lo que contribuye a los problemas de usabilidad, facilidad de mantenimiento, calidad y fiabilidad.

En respuesta a la anterior problemática, Murugesan *et al.* (2001) plantearon a la ingeniería Web como una nueva disciplina relacionada con el “*establecimiento y uso de principios científicos, de ingeniería y de gestión de enfoques disciplinados y sistemáticos, para el desarrollo, implementación y mantenimiento de sistemas exitosos de sistemas y aplicaciones basadas en la web de alta calidad*”. Los argumentos para tal planteamiento fueron:

- El proceso de desarrollo de WIS es diferente a la Ingeniería del Software y tiene características únicas. Al respecto, Isakowitz *et al.* (1998), plantea que los WIS primero que todo son sistemas de información y en segunda instancia sistemas Web, por lo cual su desarrollo es diferente a los sistemas de información tradicionales.

- La ingeniería Web es multidisciplinaria y ninguna disciplina individual puede proporcionar una base teórica completa, el cuerpo de conocimiento y la práctica para guiar el desarrollo de los WIS. A respecto, Murugesan *et al.*

(2001), Srivastava *et al.* (2014), relacionan en forma específica, aportes indispensables a la ingeniería Web desde diversas áreas, tales como HCI, interfaz de usuario, ingeniería de software e hipermedia, bases de datos, simulación, ciencias sociales y diseño gráfico, entre otros. A su vez, Sharma *et al.* (2014) plantean que además de su funcionalidad multifacética inherente, los WIS muestran un comportamiento complejo e imponen algunas demandas únicas en su usabilidad, rendimiento y seguridad entre otras.

- Temas de evolución y gestión del ciclo de vida en comparación a aplicaciones más "tradicionales". Al respecto, Srivastava *et al.* (2014) plantea que la ingeniería Web no es un clon ni un subconjunto de la ingeniería del software, sino que abarca nuevos enfoques, metodologías, herramientas y técnicas para cumplir los requisitos específicos de los WIS.

- Los WIS y las aplicaciones basadas en la Web son omnipresentes y no triviales. La perspectiva de la Web como plataforma seguirá creciendo y debe ser tratada específicamente.

Adicionalmente, la continua aparición de nuevas tecnologías hacen que las aplicaciones Web parezcan cada vez más complejas aplicaciones de escritorio, por lo cual las tareas de diseño, implementación y adaptación de este tipo de aplicaciones se están convirtiendo en una actividad propensa a errores y que consume una cantidad considerable de tiempo (Koch *et al.*, 2008). En este contexto, el enfoque de Desarrollo de Software Dirigido por Modelos (Model-Driven Software Development, MDSD) se ha convertido en un enfoque ampliamente aceptado para el desarrollo de aplicaciones distribuidas complejas en el dominio de la Ingeniería Web, llevando a una nueva disciplina llamada Ingeniería Web Dirigida por Modelos (Model-Driven Web Engineering, MDWE) que se centra específicamente, entre otros, sobre la interoperabilidad de las metodologías existentes en la actualidad para el desarrollo de aplicaciones web.

Los enfoques existentes en la MDWE tales como el Desarrollo Dirigido por Modelos (Model-Driven Development, MDD) y Arquitectura Dirigida por Modelos (Model Driven Architecture, MDA) están siendo usados para abordar con éxito la construcción, evolución y adaptación de aplicaciones Web (Koch *et al.*, 2008), convirtiéndose en un nuevo paradigma de desarrollo de software (Pons *et al.*, 2010), cuya adopción se ha incrementado en la industria. Estos enfoques ofrecen metodologías y herramientas para el diseño y desarrollo de la mayoría de los tipos de aplicaciones Web, utilizando modelos que con el apoyo de compiladores producen la mayor parte de las páginas Web de una aplicación.

3. INGENIERÍA WEB DIRIGIDA POR MODELOS (MODEL DRIVEN WEB ENGINEERING, MDWE)

Desde 2001, la ingeniería Web ha demostrado ser un dominio de aplicación donde MDA se ha aplicado con éxito en aplicaciones caracterizadas por requisitos funcionales, con modelos bien estructurados y plataformas estándar, permitiendo implementar interoperabilidad, evolución de modelo y adaptación de sistemas Web, no obstante la aparición de nuevas plataformas y cambios tecnológicos (Koch *et al.*, 2008).

El objetivo de MDA es la separación de la lógica del negocio y de la aplicación, de su plataforma tecnológica subyacente. Esto permite que los cambios en la plataforma no afectan a las aplicaciones existentes, y que la lógica de negocio pueda evolucionar independientemente de la tecnología que la implementa. Una herramienta que implemente los conceptos MDA permitirá a los desarrolladores de WIS producir modelos de la aplicación, de la lógica de negocio, y generar código para una plataforma de destino por medio de transformaciones. La principal ventaja de este enfoque es que se eleva el nivel de abstracción en el desarrollo de software.

En lugar de escribir código específico de la plataforma en algún lenguaje de alto nivel, los desarrolladores de software se centran en el desarrollo de modelos que son específicos para el dominio de aplicación, pero independiente de la plataforma. En pocas palabras, la MDA es un amplio marco conceptual que describe un enfoque general para el desarrollo de software. Luego, la construcción de un WIS consiste en una secuencia de modelos y

transformaciones entre estos. Este enfoque comienza con un modelo independiente de la plataforma (Platform-Independent Model, PIM), que se transforma posteriormente en un modelo específico de la plataforma (Platform-Specific Model, PSM), que luego se transforma a código.

Varios modelos de representación de una aplicación web se pueden encontrar en la literatura. La mayoría de ellos se han propuesto para el desarrollo de nuevas aplicaciones web y ampliar el desarrollo de los enfoques tradicionales con modelos específicos que describen las características relacionadas con la Web. A continuación seleccionamos un subconjunto de estos modelos de WIS y analizaremos una breve descripción de cada propuesta. En particular, hemos seleccionado los más conocidos (Martínez, 2012) que siguen la tendencia del paradigma orientado a objetos a saber: OOHDM (Schwabe *et al.*, 1996), WSDM (De Troyer and Leune 1998), WebML (Ceri *et al.*, 2000), UWE (Hennicker & Koch 2001), OO-H (Gómez *et al.*, 2001), OOWS (Pastor *et al.*, 2003), con el fin de identificar y analizar sus características, así como los atributos considerados.

3.1 Método de Diseño Hipermedia Orientado a Objetos (Object-Oriented Hypermedia Design Method, OOHDM)

OOHDM (Schwabe *et al.*, 1996) fue el primer método en introducir el paradigma de modelado orientado a objetos para el desarrollo de WIS personalizados (De Troyer & Leune, 1998; Martínez, 2012), centrándose en los aspectos críticos de estas aplicaciones: *estructura de navegación e interface*, por lo cual un WIS es concebido como una vista navegacional sobre un modelo conceptual, enfoque adoptado por UWE (Hennicker & Koch, 2001) y WebML (Ceri *et al.*, 2000). El método se compone de las siguientes actividades iterativas e incrementales de diseño:

Diseño conceptual: en esta actividad se construye un modelo conceptual de dominio de la semántica de la aplicación, utilizando principios de modelado orientados a objetos, sin considerar tipos de usuarios y tareas.

Diseño de navegación: en este método que tiene en cuenta aspectos cognitivos y de perfil de usuario, se considera que los objetos que el usuario navega se construyen a partir de objetos conceptuales para soportar tareas y una presentación adecuada de la información. Se expresa en dos esquemas: de clases de navegación y contexto de navegación. *El esquema de clases de navegación* contiene clases de navegación (nodos) de una aplicación hipermedia, combinando atributos de tipo de datos que reflejan el punto de vista elegido sobre el dominio de aplicación. *El esquema de contexto de navegación* representa la estructura de navegación de la aplicación y consiste de contextos de navegación tales como clases de navegación, hiperenlaces y estructuras de acceso (índices, visitas guiadas, menús) y representan colecciones de objetos que pueden ser explorados de alguna forma (por ejemplo, secuencialmente).

Diseño de interface abstracta: en esta actividad se definen los objetos de la interface serán percibidos por el usuario, así como la forma de visualización y sincronización de objetos de navegación, respuesta a eventos externos y las transformaciones que se llevarán a cabo. Adicionalmente se utiliza un enfoque de diseño de vista de datos para complementar la descripción de la interface.

Implementación: en esta actividad se asignan modelos concretos a los modelos abstractos previamente desarrollados y se implementan en cualquier plataforma hipermedia disponible.

3.2 Método de diseño de Web Sites (Web Site Design Method, WSDM)

WSDM (De Troyer & Leune, 1998), es un método cuyo objetivo principal es el diseño centrado en el usuario, para lo cual se enfoca en un conjunto de visitantes potenciales a un sitio Web como punto de partida, con lo cual el WIS diseñado tendrá mayor usabilidad y proporcionarán una mayor satisfacción de usuario. El método consiste de las siguientes fases:

Modelado de usuario: debido a que un sitio web debe anticiparse a las preguntas de los usuarios y responderlas, la primera fase del método se concentra en los potenciales usuarios del sitio. En esta fase los usuarios se clasifican desde una perspectiva de requisitos de información y desde sus características en relación con un sitio Web

(experiencia, frecuencia de uso, uso obligatorio/discrecional, motivación para el uso, lenguaje, educación, habilidades intelectuales, lingüísticas y edad entre otros) las cuales orientan en cómo la información debe ser presentada, además de que describen requisitos de usabilidad.

Diseño Conceptual: en esta fase se modelan formalmente los requisitos de información expresados en las descripciones de las clases de usuarios, donde se construye el modelo de objetos de usuario. Adicionalmente se aborda el diseño de navegación, donde se construye el modelo conceptual de navegación, que consiste de un número de rutas de navegación, uno por cada perspectiva de usuario donde se describe en términos de componentes y enlaces. Con respecto a los componentes se distingue entre información, navegación y externos.

Diseño de la Implementación: en esta fase se crea la apariencia para el diseño conceptual realizado en la fase anterior y su resultado es un modelo de implementación.

Implementación: en esta fase el resultado de la fase de diseño de la aplicación se implementa en un lenguaje de programación específico

3.3 Lenguaje de Modelado Web (Web Modeling Language, WebML)

WebML (Ceri *et al.*, 2000), es un lenguaje de especificación de alto nivel para la estructura de contenidos en hipertexto de aplicaciones Web de uso intensivo de datos. Este modelo enfatiza en la definición de la navegación e incluye una noción explícita de vista del sitio para satisfacer los intereses de diversos grupos de usuarios considerando sus diversos dispositivos de acceso. La especificación de un sitio en WebML consta de cuatro perspectivas:

Una adaptación de modelos conceptuales apropiados para diseñar de datos es el modelo de datos WebML, tal como se utiliza anteriormente en diferentes disciplinas que se utiliza, por ejemplo, el diseño de bases de datos, la representación del conocimiento y de la ingeniería de software. El modelo Entidad-Relación de datos, empleada en el diseño de la base de datos conceptual, y de clase UML diagramas, empleadas en el modelado orientado a objetos son incompatibles con ella.

Modelo de datos: WebML no propone un lenguaje de modelado de datos, pero es compatible con modelos conceptuales de datos usados en diferentes disciplinas tales como bases de datos.

Modelo de hipertexto: describe la disposición del hipertexto que puede ser publicado, así como la navegación del sitio Web y se compone de dos modelos: *Modelo de composición*, que especifica qué páginas componen el hipertexto, y que unidades de contenido constituyen una página. Se emplean componentes atómicas de contenido para publicar datos explicados en el modelo de datos, llamadas unidades, las cuales para componer páginas en WebML son clasificadas en siete tipos: datos, multi-datos, índice, entradas, filtro, scroller y unidades directas y el *Modelo de navegación*, expresa cómo se vinculan las páginas y las unidades de contenido para formar el hipertexto.

Modelo de presentación: expresa la disposición y el aspecto gráfico de las páginas, independientemente del dispositivo de salida y del lenguaje de interpretación, planteando especificaciones sobre páginas específicas o genéricas.

Modelo de personalización: usuarios y grupos de usuarios se modelan explícitamente en el esquema de la estructura en forma de entidades predefinidas llamadas Usuario y Grupo. Las características de estas entidades se pueden usar para almacenar contenido específico de grupo o individuo, como sugerencias, lista de favoritos y recursos para personalización gráfica.

3.4 Método de Ingeniería Web Basada en UML (UML-Based Web Engineering Method, UWE)

UWE (Hennicker & Koch, 2001) es un método de diseño basado en UML adaptado para el dominio Web, que toma en cuenta el espacio de navegación y aspectos de presentación de la aplicación, los cuales pueden ser personalizados. El método consta de tres pasos iterativos que se orientan al diseño de los siguientes componentes:

Modelo Conceptual: se relaciona con la construcción de un modelo de dominio Web, que partiendo de los requisitos funcionales incorpora las actividades a desarrollar y los diversos tipos de usuarios involucrados haciendo énfasis en la información intercambiada.

Modelo de navegación. Se relaciona con los objetos que pueden ser visitados en el espacio de navegación a través de la aplicación Web y con las formas de accesibilidad de estos objetos en la estructura de acceso. Los principales elementos de modelado de navegación son las clases de navegación y sus enlaces de navegación asociados. Los elementos que se utilizan para diseñar el modelo de navegación UWE son: Clase Navegación, Menú, Nodo externo, Consulta, Visita guiada, Índice y Enlace de Navegación.

Modelo de Presentación: Se relaciona con la creación de objetos de navegación y sus primitivas de acceso, que son accesibles por el usuario. Los elementos que se utilizan para el diseño de este modelo son: Alternativas de Presentación, Grupo de Presentación, Grupo de presentación iterada, Formulario de Entrada, Página de presentación, Ficha, Botón, Anchor, Texto, Imagen, Objeto Media, Selección, Subir Archivo, Componente de personalización, Deslizador, texto de entrada e imagen de entrada. La estructura de navegación es transformada por modelo de presentación en un conjunto de modelos.

3.5 Método Hipermedia Orientado a Objetos (Object-Oriented Hypermedia Method, OO-H)

OO-H (Gómez *et al.*, 2003) es un modelo basado en el paradigma orientado a objetos, que proporciona al diseñador la semántica y notación necesarias para el desarrollo de interfaces personalizadas basadas en la Web y su conexión con aplicaciones previamente existentes. El enfoque OO-H plantea entre otros los siguientes modelos:

Modelo de navegación: se refiere a la estructuración de la navegación del sistema teniendo en cuenta aspectos tales como el comportamiento deseado de navegación, la selección de la población objeto, el orden en que los objetos deben ser navegados o aspectos del acceso, características que son modeladas mediante diferentes tipos de patrones de navegación, filtros asociados a los enlaces y colecciones.

Modelo de presentación: se refiere a la implementación de la presentación del sistema mediante un conjunto de diagramas abstractos de presentación (Abstract Presentation Diagram, APD), que reflejan la estructura de las páginas abstractas de la interface y que pueden ser derivados de forma automática de diagramas NAD. Un diagrama APD da una interfaz funcional con ubicación predeterminada y estilos para cada elemento de información o la interacción.

3.6 Método de Producción de Soluciones Web Orientadas a Objeto (Object-Oriented Web Solutions, OOWS)

OOWS (Pastor *et al.*, 2003) es un método que amplía OO-H (Gómez *et al.*, 2003) en nuevos modelos para apoyar características particulares de navegación y presentación de aplicaciones Web. Estos modelos son:

Modelo de Usuario: Permite especificar los tipos de usuarios que pueden interactuar con el sistema. Los tipos de usuarios son organizados en forma jerárquica por medio de relaciones de herencia que nos permiten especificar la especialización de navegación.

Modelo de navegación: Define la estructura de navegación del sistema, mediante la descripción de la navegación permitida para cada tipo de usuario por medio de un Mapa de Navegación. Este mapa de navegación proporciona una estructura de accesibilidad a la información y funcionalidad, que personaliza el sistema en función del tipo de usuario. Una vez construidos los mapas de navegación, se plantean mecanismos adicionales para estructurar el acceso del usuario a información mediante índices a listas de información resumida y filtros de consulta para restringir el espacio de búsqueda de datos dentro del contexto de navegación.

Modelado de presentación: Especifica las propiedades visuales de la información que se muestra, tales como disposición de la información (registro, tabular, maestro-detalle entre otros), orden (ascendente o descendente) y cardinalidad (número de objetos a presentar).

4. CARACTERÍSTICAS DE LOS WIS DESDE LA PERSPECTIVA DEL USUARIO

El objetivo principal de un WIS es la generación automática de presentaciones hipermedia de los datos del sistema, es decir del contenido que se encuentra dentro o fuera de la aplicación, en un dominio de aplicación específico (Aroyo & Dicheva, 2002; Vdovjak & Houben, 2002). Adicionalmente, en un WIS se utilizan hipervínculos que proporcionan la conexión entre contenidos y/o sitios relacionados, es decir la capacidad de navegación, permitiendo que el diseño de la estructura del sitio se pueda implementar a través de una red de enlaces escalable (Wang & Head, 2001).

Luego, en un WIS se presentan dos aspectos hipermedia: hipervínculos y multimedia. Desde el enfoque MDA relacionado anteriormente, las características hipermedia presentes en un WIS según los métodos de desarrollo Web analizados, se implementan en las vistas de: contenido, navegación y presentación. A continuación se describen estas características hipermedia desde la perspectiva orientada al producto, donde la usabilidad se define como un componente de calidad del producto.

4.1 Modelo de contenido y calidad de la información

La multimedia proporciona medios de comunicación de información tales como gráficos, audio y video, además de texto. Al incorporar este tipo de recursos al WIS, la información que este suministra puede ser presentada en forma más aproximada al mundo real, de tal forma que los usuarios pueden recibir información por este medio de presentación de manera más eficiente y eficaz, lo cual les permitirá evaluar alternativas de manera más apropiada (Wang & Head, 2001).

En relación con la calidad de la información que debe presentar este modelo, Wang y Strong (1996) definen la Calidad de Información (Information Quality, IQ) como “*datos que son aptos para uso por los consumidores de datos*” y proponen desde la perspectiva de usuario el modelo TDQM (Total Data Quality Management) compuesto de cuatro categorías con sus respectivas características: *Intrínseca*, que se relaciona con la calidad de los datos en sí mismos (credibilidad, precisión, objetividad y reputación); *contextual* relacionada con requerimientos en diferentes contextos (valor adicional, plenitud, pertinencia, oportunidad y adecuación); *representacional* relacionada con la utilización de información como interpretable (consistencia, concisión, interpretabilidad y comprensible); y *accesibilidad* (accesibilidad y seguridad).

Basándose en el modelo de Wang y Strong (1996) que clasifica la información como producto, Khan *et al.* (2002) extienden el concepto de información a servicio, planteando un modelo conceptual de la IQ como calidad conforme a las especificaciones y como expectativas de los clientes. Las características de IQ por cada categoría son: *Intrínseca* (credibilidad, precisión, objetividad y reputación); *contextual* (valor adicional, plenitud, pertinencia, oportunidad y adecuación); *representacional* (consistencia, concisión, interpretabilidad y comprensibilidad); y *accesibilidad* (accesibilidad y seguridad).

A nivel de normas internacionales, ISO/IEC 25000 (Software Quality Requirement Evaluation, SQuaRE) (ISO, 2005) en la división ISO/IEC 2501n – Quality Model Division, la parte 25012 (ISO, 2008) se centra en la calidad de los datos. Este modelo define características consideradas desde dos puntos de vista diferentes: inherente y dependiente del sistema. La calidad inherente de los datos se refiere al grado en el que las características de calidad de los datos tienen el potencial intrínseco para satisfacer necesidades del usuario cuando los datos se utilizan bajo condiciones especificadas. Las características de este conjunto son específicamente: Precisión, Integridad, coherencia, credibilidad y actualidad. La calidad de los datos dependiente del sistema, se refiere al grado en el que la calidad de datos se logra y conserva dentro de un sistema informático, cuando los datos se utilizan en las condiciones especificadas. Las características de este conjunto son específicamente: disponibilidad, portabilidad y recuperabilidad. El conjunto de características de ambos puntos de vista se compone de accesibilidad, cumplimiento, confidencialidad, eficiencia, precisión, trazabilidad y comprensibilidad.

Con respecto a la IQ aplicada específicamente a contenidos de WIS, Caro *et al.* (2008), desarrollaron una encuesta para clasificar dimensiones de IQ y como resultado presentaron el modelo de evaluación PDQM (Portal Data Quality Model), adaptado de Wang y Strong (1996). Basándose en las expectativas de los consumidores y en la funcionalidad que la información ofrecen a los usuarios, plantea a su vez cuatro categorías de IQ: Intrínseca (precisión, objetividad, credibilidad, reputación, duplicados, vencimiento, trazabilidad); contextual (valor agregado aplicabilidad, completo, validez, flexibilidad, novedad, confiabilidad, pertinencia, especialización, oportunidad); representacional (consistente, conciso, interpretabilidad, comprensibilidad, cantidad, atractivo, documentación, organización); y operacional (accesibilidad, seguridad, tiempo de respuesta, Interactividad, disponibilidad, atención al cliente, facilidad de operación).

A su vez, Moraga *et al.* (2010) presentan el modelo SPDQM (SQuaRE Portal Data Quality Model), que es una adaptación para portales Web de las características de ISO/IEC 25012, mediante el uso del modelo de calidad de datos PDQM (Caro *et al.*, 2008). Las dimensiones se identifican desde los puntos de vista inherente y dependiente del sistema. La vista inherente se relaciona con la categoría *Intrínseca*, que denota que los datos tienen calidad en su propio derecho. La vista dependiente del sistema, se divide en tres las categorías: primera la *operacional*, que enfatiza la importancia de la función del sistema, es decir, debe ser accesible pero seguro, segunda la *contextual*, que pone de relieve que la calidad de datos debe ser considerada en el contexto de la tarea en cuestión y tercera, la *representacional* que denota que el sistema debe presentar datos de una manera tal que sean interpretables, fáciles de entender y representados de manera concisa y consistente.

4.2 Modelo de navegación

El modelo de navegación de WIS tiene como objetivo definir vistas, acceso a estructuras y vínculos a contenidos para facilitar su fácil acceso y navegación por los usuarios. La mayoría de los métodos de WE basan sus modelos de navegación en dos primitivas de modelado basado en grafos: *Nodo*, que representa vistas lógicas sobre clases del modelo de la aplicación, y *vínculo*, que representa la implementación hipermedia de las asociaciones del modelo de aplicación. El problema de estos vínculos, es que pueden generen problemas de navegación al usuario, dado que este puede fácilmente perderse en la red de vínculos o estar expuesto a una sobrecarga cognitiva (Kappel *et al.* 2005). Al respecto, Nielsen (1993) afirma que este es uno de los principales problemas de usabilidad con entornos hipermedia a gran escala, como la Web.

Con respecto al modelo de navegación, las propiedades gráficas de una aplicación Web reflejan su organización estructural de hipertexto y por lo tanto esta puede ser considerada como un grafo (Kaur & Dani, 2011), con atributos tales como: *Tamaño*, medido como el número de nodos (o contextos de navegación), el número de vínculos de navegación o una combinación de éstos; *profundidad*, medida como la longitud de la trayectoria más larga desde el nodo raíz a un nodo hoja dado; *amplitud*, medida como el número máximo de nodos en cualquier nivel dado; y ratio (proporción borde a nodo), medida que aumenta entre más vínculos se añadan entre nodos (Abrahão *et al.*, 2003).

Desde esta perspectiva gráfica del modelo de navegación, Abrahão et al (2003) define las siguientes medidas de un mapa de navegación: *Indicador de densidad* de un mapa de navegación, que se calcula dividiendo el número de nodos entre el número de vínculos; *profundidad de un mapa de navegación* que se relaciona con la distancia en el mapa de navegación del nodo raíz a un nodo terminal, indicando la facilidad con la que se puede llegar a este y la probable importancia de su contenido. Se puede medir a su vez, la profundidad máxima, mínima y media de un mapa de navegación para cada usuario; *amplitud de un mapa de navegación* que se relaciona con la cantidad de nodos alcanzables a través de un vínculo manual desde un nodo dado. Esta característica se relaciona con la dificultad de comprensión del WIS por parte del usuario, debido a que entre más opciones simultáneas tenga para escoger más complejo le será la comprensión de la interacción; *compactación*, que se refiere al grado de interconexión entre los nodos que pertenecen a un grafo hipermedia., donde una alta compactación significa que de

cada nodo se puede llegar fácilmente a cualquier otro nodo en el mapa de navegación, lo cual puede indicar un mapa altamente conectado que puede conducir a la desorientación. Una baja compactación indica un número insuficiente de vínculos, lo que puede significar que las partes de un mapa se desconectan, presentándose nodos huérfanos o sin salida; *camino mínimo entre nodos*, que se relaciona con la cantidad mínima de vínculos que son necesarios para pasar desde un nodo dado a un nodo destino; y *número de caminos entre nodos*: La cantidad de caminos alternativos que unen a dos nodos dentro de un mapa de navegación.

Adicionalmente, una característica importante de un WIS es que posea vínculos externos y de esta manera provea redirección a los usuarios a otros dominios, por lo cual debe carecer de *enlaces rotos*, es decir debe presentar *integridad de vínculos* para no afectar negativamente su credibilidad (Aggarwal *et al.*, 2014).

4.3 Modelo de presentación

Como ya se anotó, el modelo de presentación tiene como objetivo visualizar información multimedia (texto, gráficos, audio y video) en la interface de usuario (User Interface, UI) del WIS (teniendo como entrada el modelo de navegación), de tal forma que los usuarios puedan evaluar alternativas de manera apropiada (Wang & Head, 2001). Con respecto a la información multimedia que se presenta, esta debe tener ciertas características de calidad que la hagan comprensible, útil, agradable y fácil de usar para el usuario.

En este contexto, el estándar SQuaRE (ISO, 2005) en su definición de usabilidad aborda dentro de sus sub-características, a la *estética de la UI* que se refiere al grado en el cual la UI proporciona placer y satisfacción al usuario cuando interactúa con el sistema, al *reconocimiento adecuado* que se refiere al grado de reconocimiento de los usuarios sobre lo apropiada que puede una aplicación Web ser a sus necesidades, y a la *operatividad*, que se refiere al grado en que una aplicación web tiene atributos que la hacen fácil de operar y controlar. Con respecto a estas sub-características, Abrahão *et al.* (2014) plantean que son definidas por el estándar en forma genérica y que para usarse deben subdividirse en atributos medibles “*con el fin de cubrir un conjunto de aspectos de usabilidad web lo más ampliamente posible*”, para lo cual proponen un desglose teniendo en cuenta criterios ergonómicos y otras directrices de usabilidad para el desarrollo Web.

Con respecto a la sub-característica *estética de la UI*, Abrahão *et al.* (2014) plantean como atributos medibles: *Uniformidad de elementos*, que se relaciona con la uniformidad de características que deben tener todos los elementos presentados en cualquier componente de la UI, como tipo de letra, color y posición; *personalización de apariencia*, que se relaciona con el ajuste de la UI a las preferencias de los usuarios; y *grado de interactividad*, que se relaciona con “*el grado en que los usuarios pueden participar en la modificación de la forma y el contenido de un entorno de medios de comunicación en tiempo real*”.

En relación con la sub-característica *reconocimiento adecuado*, Abrahão *et al.* (2014) plantean como atributos medibles: *Legibilidad óptica*, que se refiere a que tan legibles son los textos e imágenes, en lo que respecta a tamaño de fuente, contraste y posición del texto entre otros; *legibilidad de la información*, que se refiere a atributos de la información que involucran aspectos de cohesión y densidad de información, y el apoyo de paginación; *familiaridad*, que se refiere a la facilidad con la que un usuario reconoce los componentes de la UI y ve su interacción como natural; *reducción de la carga de trabajo*, que se relaciona con la reducción del esfuerzo cognitivo del usuario; *guía al usuario*, que se relaciona con la disponibilidad de mensajes y la retroalimentación informativa en respuesta a las acciones de este; y *navegabilidad*, que se relaciona con la facilidad de acceso a contenido por parte del usuario.

En relación con la sub-característica *operatividad*, Abrahão *et al.* (2014) plantean otras sub-características relacionadas con los aspectos técnicos de las aplicaciones Web tales como: *compatibilidad* con otros productos de software o agentes externos de la aplicación Web; *gestión de datos* de acuerdo con la validez de los datos y su privacidad; *control en la ejecución* de acciones tales como cancelar y deshacer; *capacidad de adaptación* que es la capacidad de la aplicación web para ser adaptada por el usuario; *adaptabilidad*, que es la capacidad de la aplicación web para adaptarse a las necesidades de los usuarios; y *consistencia* en el comportamiento de enlaces y controles.

5. CONCLUSIONES

Las características de los WIS que se encuentran en el modelo de contenido se orientan hacia el dominio de calidad de datos (DQ) y calidad de la información (IQ) y se relacionan con el dominio de bases de datos. En la revisión bibliográfica se encontró que la mayoría de los autores no hacen diferencia entre DQ e IQ lo cual muestra confusión y evidencia que se requiere más investigación en esta área. A su vez las características de los modelos de navegación y presentación se relacionan entre sí y como se esperaba se orientan básicamente a características hipermedia, donde se presentan algunas variables multidimensionales.

Aunque la mayoría de los métodos MDA analizados son los más conocidos en el área, los modelos con los que abordan el problema de la representación de la arquitectura de WIS tienen marcadas diferencias. No obstante para el modelo de navegación es indiferente si se habla de mapas navegacionales o NADs, debido a que su representación es realmente un grado dirigido y en este contexto se realizó el análisis.

Aunque se habla en la revisión de métodos MDA de clases en el contexto de programación orientada a objetos, el concepto no es relevante a nivel de derivación de generalizaciones conceptuales.

Como limitación del presente artículo, se reconoce que los conceptos y conclusiones planteadas están en función de la literatura y autores revisados, y que es posible que se encuentren reportados trabajos de gran importancia que no hayan sido abordados.

REFERENCIAS

- Abrahão, S., Condori-Fernández, N., Olsina, L., & Pastor, O. (2003, September). Defining and validating metrics for navigational models. In *Software Metrics Symposium, 2003. Proceedings. Ninth International*. IEEE.
- Abrahão, S., Insfran, E., & Fernandez, A. (2014). Designing Highly Usable Web Applications.
- Aroyo, L., & Dicheva, D. (2002). Authoring framework for concept-based web information systems. *Proceedings of Concepts and Ontologies in Web-based Educational Systems, ICCE2002*, 41-48.
- Aggarwal, S., Van Oostendorp, H., Reddy, Y. R., & Indurkha, B. (2014, September). Providing Web Credibility Assessment Support. In *Proceedings of the 2014 European Conference on Cognitive Ergonomics* (p. 29). ACM.
- Binemann-Zdanowicz, A., Kaschek, R., Schewe, K. D., & Thalheim, B. (2004, January). Context-aware web information systems. In *Proceedings of the first Asian-Pacific conference on Conceptual modelling-Volume 31* (pp. 37-48). Australian Computer Society, Inc.
- Calero, C., Ruiz, J., & Piattini, M. (2005). Classifying web metrics using the web quality model. *Online Information Review*, 29(3), 227-248.
- Caro, A., Calero, C., Caballero, I., & Piattini, M. (2008). A proposal for a set of attributes relevant for Web portal data quality. *Software Quality Journal*, 16(4), 513-542.
- Ceri, S., Fraternali, P., & Bongio, A. (2000). Web Modeling Language (WebML): a modeling language for designing Web sites. *Computer Networks*, 33(1), 137-157.
- De Troyer, O. M. F., & Leune, C. J. (1998). WSDM: a user centered design method for Web sites. *Computer Networks and ISDN systems*, 30(1), 85-94.
- Detlor, B. (2000). The corporate portal as information infrastructure: towards a framework for portal design. *International Journal of Information Management*, 20(2), 91-101.
- Fiedler, Gunar, et al. "Process Improvement for Web Information Systems Engineering." *Proceedings of the 7th International SPICE Conference on Process Assessment and Improvement*.
- Ginige A., Murugesan S. (2001) Web Engineering: An Introduction. *IEEE Multimedia*, 1-3, pp. 14-18.
- Gómez, J., & Cachero, C. (2003). OO-H Method: extending UML to model web interfaces. *Information modeling for internet applications*, 144-173.

- Hennicker, R., & Koch, N. (2001). Systematic Design of Web Applications with UML. Unified Modeling Language: Systems Analysis, Design and Development Issues, pp. 1–20.
- Isakowitz, T., Bieber, M., & Vitali, F. (1998). Web information systems. *Communications of the ACM*, 41(7)78.
- ISO/IEC (2005), ISO/IEC 25000: Software Engineering – Software Product Quality Requirements and Evaluation (SQuaRE) – Guide to SQuaRE, International Organisation for Standardisation, Geneva
- ISO/IEC (2008), ISO/IEC 25012:2008, Software Engineering – Software Product Quality Requirements and Evaluation (SQuaRE) – Data Quality Model, International Organisation for Standardisation, Geneva
- Kahn, B. K., Strong, D. M., & Wang, R. Y. (2002). Information quality benchmarks: product and service performance. *Communications of the ACM*, 45(4), 184-192.
- Kappel, G., Pröll, B., Retschitzegger, W., & Schwinger, W. (2003). *Customisation for ubiquitous web applications*. na.]
- Kappel, G., Pröll, B., Reich, S., & Retschitzegger, W. (2005). Web Engineering.
- Kaur, A., & Dani, D. (2011). The Systematic Review of Web Measurement. *International Journal of Computer Science & Engineering Survey (IJCSSES) Vol,2*.
- Koch, N., Meliá-Beigbeder, S., Moreno-Vergara, N., Pelechano-Ferragud, V., Sánchez-Figueroa, F., & Vara-Mesa, J. (2008). Model-driven web engineering. *Upgrade-Novática Journal (English and Spanish), Council of European Professional Informatics Societies (CEPIS) IX (2)*, 40-45.
- Levin, M. S. (2011). Towards Configuration of applied Web-based information system. *arXiv preprint arXiv:1108.6223*.
- Martínez, A. F. (2012). A Usability Inspection Method for Model-driven Web Development Processes.
- Moraga, C., Moraga, M. Á., Caro, A., & Calero, C. (2010, July). Modeling the intrinsic quality of web portal data. In *13th Workshop on Quantitative Approaches on Object-Oriented Software Engineering and Related Paradigms, Málaga, Spain* (pp. 1-12).
- Murugesan, S., Deshpande, Y., Hansen, S., & Ginige, A. (2001). Web engineering: A new discipline for development of web-based systems. In *Web Engineering* (pp. 3-13). Springer Berlin Heidelberg.
- Nielsen, J. (1993). Usability Engineering, *Academic Press, Boston, MA*.
- Pastor, O., Fons, J., & Pelechano, V. (2003, July). Oows: A method to develop web applications from web-oriented conceptual models. In *International Workshop on Web Oriented Software Technology (IWWOST)*.
- Pokhrel, P., & Vemulapalli, S. P. B. (2010). MEASURING SATISFACTION OF USAGE FOR WEB-BASED INFORMATION SYSTEM (WBIS) FOR e-SERVICES.
- Pons, C., Giandini, R. S., & Pérez, G. (2010). Desarrollo de software dirigido por modelos.
- Schwabe, D., Rossi, G., & Barbosa, S. D. (1996, March). Systematic hypermedia application design with OOHD. In *Proceedings of the the seventh ACM conference on Hypertext* (pp. 116-128). ACM.
- Schewe, Klaus-Dieter, and Bernhard Thalheim. "Conceptual modelling of web information systems." *Data & Knowledge Engineering* 54.2 (2005): 147-188.
- Sharma, A., Sharma, D., RAINA, A. K. A., VishalSharma, P. Y. P. Y., Solanki, Y. S. V., Solanki, Y. S. V., ... & NALAWADE, P. (2014). Web Engineering: For Security and Development purpose.
- Srivastava, S., Kurda, S., & Dubey, P. (2014). Web Engineering Security: Essential Elements. in *IRACST- International Journal of Research in Management & Technology (IJRMT), ISSN: 2249-9563 Vol. 4, No.5, October 2014*.
- Taivalasari, A., Mikkonen, T., Ingalls, D., & Palacz, K. (2008). Web browser as an application platform: The lively kernel experience.
- Thakur, S., & Rawat, S. (2014). Development of Web-based Systems: Web Engineering has a new discipline.
- Vdovjak, R., & Houben, G. J. (2002, January). Providing the semantic layer for WIS design. In *Advanced Information Systems Engin//eering* (pp. 584-599). Springer Berlin Heidelberg.

- Wang, F., & Head, M. M. (2001). A model for Web-based information systems in e-retailing. *Internet Research, 11*(4), 310-321.
- Wang, R. Y., D. M. Strong. (1996). Beyond accuracy: what data quality means to data consumers. *Journal of Management Information Systems 12* (4) 5-33.
- Worwa, K., & Stanik, J. (2010). Quality of Web-based information systems. *Journal of Internet Banking and Commerce, 15*(3).