

USO DE IFML EN APLICACIONES WEB

Beatriz Alejandra Olivares-Zepahua¹, Yasmin Rosales Cruz², Celia Romero-Torres³, Luis Ángel Reyes-Hernández⁴

^{1,2,3,4} Instituto Tecnológico de Orizaba, Veracruz, México

¹ bolivares@ito-depi.edu.mx, ² yzn_1808@hotmail.com, ³ cromero@ito-depi.edu.mx, ⁴ l_r_h01@hotmail.com

Resumen

Las aplicaciones Web tienen una complejidad especial para su modelado, debido a que manejan elementos de distintos tipos de acuerdo a la tecnología con que se construyen y se adaptan fácilmente al paradigma de objetos que manejan muchos de los lenguajes de programación para Web; sin embargo, las técnicas tradicionales de modelado de aplicaciones basadas en UML no incluyen el manejo de la interfaz gráfica y la interacción de ésta con el usuario final, lo que en el caso Web deja un gran vacío que IFML (Interaction Flow Modeling Language) busca cubrir. El presente trabajo ejemplifica el uso de IFML para modelar aplicaciones Web independientemente de la tecnología específica de implementación.

PALABRAS CLAVE: *aplicación Web, IFML, modelado de aplicaciones, modelo de interacción, modelo de dominio.*

1. Introducción

Las aplicaciones cliente/servidor trabajan bajo el principio de “divide y vencerás”, es decir, la funcionalidad de la aplicación se ejecuta en dos lugares diferentes, por lo general el cliente representa la parte que interactúa con el usuario e incluye una interfaz gráfica, considerando “cliente ligero” a aquél que sólo incluye la presentación de información y un mínimo de validaciones y “cliente pesado” a aquél que además de eso implementa algún tipo de lógica de negocio y/o provee una interfaz gráfica con mayor riqueza; por su parte, en el servidor se realizan los procesos más fuertes de cálculo, búsqueda y/o implementación de reglas de negocio.

Las aplicaciones Web son un tipo especial de aplicaciones cliente/servidor. El cliente lo constituye un visualizador Web instalado en una computadora de escritorio, en una laptop o en dispositivo móvil como tablet o teléfono inteligente, en tanto que el servidor comúnmente se encuentra en un espacio físico distante y lo constituye al menos un equipo; la comunicación entre el cliente y el servidor se realiza generalmente mediante los protocolos HTTP, TCP/IP y otros (Shklar, 2003). Las aplicaciones Web tradicionales cuentan con clientes ligeros, ya que por cuestiones de seguridad los visualizadores se implementaron evitando casi cualquier acceso a los recursos del equipo cliente como son

disco duro y periféricos; sin embargo, a medida que los equipos aumentaron sus capacidades de cómputo, los visualizadores incrementaron también sus posibilidades de acción, con la implementación de funcionalidades muy comunes en aplicaciones de escritorio pero inexistentes en la mayoría de los visualizadores hasta 2008 aproximadamente, lo que abrió la posibilidad de contar con clientes pesados en las aplicaciones Web, tanto en lo concerniente a funcionalidad como en lo relacionado a la riqueza de las interfaces gráficas de usuario disponibles, dando origen a un nuevo tipo de aplicaciones Web: las aplicaciones enriquecidas de Internet o RIAs por su siglas en inglés.

En el caso de las aplicaciones Web tradicionales generalmente se manejaban como lenguajes en el cliente HTML, CSS y JavaScript, mientras que del lado del servidor se utilizaban tecnologías como PHP o Java (JSP y/o Servlets o bien marcos de trabajo como JSF o Struts) o ASP.NET o Python o Perl entre otros muchos; si, además, la aplicación requería almacenar información en una base de datos, también era posible combinar PostgreSQL o MySQL o SQLServer u Oracle o Informix o incluso obtener información de otros sistemas mediante la invocación de servicios Web. En el caso de las RIAs, también intervienen en el lado del cliente XML, XHTML, JSON y una adecuación de HTTP, XMLHttpRequest, que permite la solicitud y respuesta parcial de páginas (Dong, 2011).

Por todas las complejidades que implican las aplicaciones Web tradicionales y las RIAs, al hacer uso de muy diversos lenguajes y/o marcos de trabajo y de muy diversas capacidades, se hace indispensable el uso de técnicas de Ingeniería de Software para asegurar que las aplicaciones cumplen con las expectativas del usuario (Laaz, 2016).

2. Conceptos

A continuación se describen los conceptos básicos necesarios para un mejor entendimiento del presente documento.

2.1. Aplicación Web

Una Aplicación Web es un conjunto de páginas web estáticas y dinámicas que los usuarios utilizan al acceder a un servidor web a través de Internet o de una intranet. En una aplicación web, se consulta, procesa y analiza una gran cantidad de datos volátiles que se almacenan en grandes estructuras. Estas aplicaciones tienen gran interacción con el usuario, por lo tanto requieren un diseño de interfaz claro, simple y que se estructure y personalice para cada tipo de usuario (Shklar, 2003).

2.2. RIA (Aplicación Enriquecida de Internet)

Las aplicaciones enriquecidas para Internet también conocidas como RIA son aplicaciones que combinan las ventajas de las aplicaciones Web y las de las aplicaciones de escritorio. Las

RIAs son básicamente aplicaciones Web que ofrecen funcionalidades y características similares a las aplicaciones de escritorio, por lo cual son más completas. Con las aplicaciones Web comparte la arquitectura cliente/servidor y el uso del visualizador, mientras que de las aplicaciones de escritorio toma los clientes enriquecidos (interfaz gráfica de usuario que permite cosas como el reordenamiento de información, arrastrar y soltar elementos, cambios de tamaño, adaptación a distintos ambientes visuales entre otros) (Adobe, 2016).


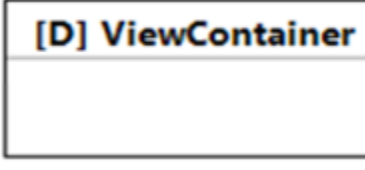
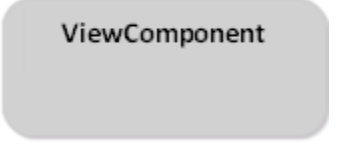


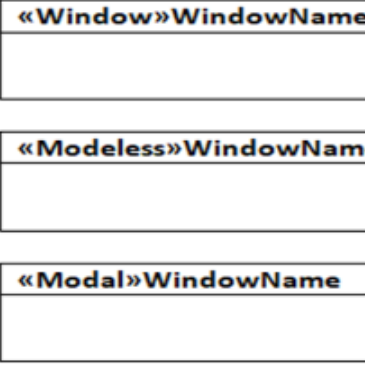
2.3. IFML

IFML (*Interaction Flow Modeling Language*) es un lenguaje para modelar los flujos de interacción dentro de aplicaciones, tanto Web tradicionales como Enriquecidas de Internet; se trata, desde 2013, de un estándar del OMG (*Object Management Group*, Grupo de Administración de Objetos) para integrar el diseño del *front-end* de las aplicaciones al modelado del resto de la aplicación; entre sus objetivos se encuentra expresar el contenido, la interacción con el usuario y el comportamiento de la interfaz gráfica de usuario que se ha vuelto más compleja a partir de la cantidad de elementos existentes. Otros lenguajes de modelado, como UML, cuentan con diagramas que permiten modelar interacción entre elementos (entre ellos los usuarios), sin embargo, resultan insuficientes respecto a la complejidad las aplicaciones actuales. IFML se basa en otros lenguajes como WebML y herramientas como WebRatio, pero tiene la ventaja de ser independiente del lenguaje de implementación, lo que no sucede con WebRatio. Debido a lo novedoso del estándar, existen relativamente pocos artículos al respecto, aunque se espera un crecimiento en el corto tiempo (IFML, 2016).

3. Elementos IFML

IFML define dos modelos, el modelo de interacción que es enteramente nuevo y representa los elementos de la interfaz gráfica de la aplicación así como la forma en que el usuario pasa de un elemento a otro y la forma de hacerlo, para este modelo se tienen elementos centrales (*core elements*) y elementos extendidos (*extended elements*). Dentro de los elementos centrales se encuentran diversos contenedores de vistas (*view container*), acciones y eventos a los que reacciona la interfaz, flujos de navegación y datos, entre otros. Los elementos extendidos consideran formularios, listas de datos que permiten seleccionar información, la presentación de los datos seleccionados así como el manejo de ventanas (Brambilla, 2015). La tabla 1 muestra algunos de los elementos centrales y extendidos de IFML.

Tabla 1. Ejemplos de elementos centrales y extendidos de IFML.

Elemento	Símbolo	Elemento	Símbolo
Contenedor de vista		Contenedor de vista por omisión	
Componente visual		Flujo de navegación	
Lista		Ventana (normal, no modal y modal)	

Por otra parte, el modelo de dominio se basa en el diagrama de clases de UML, con algunas diferencias notables como es el hecho de marcar un atributo como identificador o llave primaria a un nivel de entidad, como sucede en los diagramas Entidad-Relación; debido a que UML no provee cómo marcar un atributo con tal característica, IFML propone el uso del estereotipo <<PK>>. Es importante hacer notar que no se trata exactamente del concepto de llave primaria de las bases de datos relacionales, puesto que el modelo de dominio IFML representa un nivel de abstracción muy alto, se trata de una clase o entidad que, durante la etapa de diseño técnico, probablemente se convierta en una o más tablas de una base relacional que requerirá efectivamente una llave primaria (Brambilla, 2014; Brambilla, 2015).

4. Modelado de una Aplicación Web

Un comportamiento muy común en las aplicaciones Web es mostrar una lista de elementos del dominio y permitir obtener información más detallada de dicho elemento en otro espacio; cuando se trata de aplicaciones tradicionales la información se muestra en una nueva página que sustituye a la primera, en las aplicaciones enriquecidas es común colocar un dialogo modal que se sobrepone a la página original. En la figura 1 se muestra un ejemplo del comportamiento de una aplicación tradicional y en la figura 2 se muestra el ejemplo correspondiente a una aplicación enriquecida.

Clave	Nombre Completo	Operación	
2	Hugo	Modificar	Borrar
3	Paco	Modificar	Borrar
4	Luis	Modificar	Borrar

Crear Consultor

Clave

Contraseña

Repite Contraseña

Nombre

Primer Apellido

Segundo Apellido

RFC

Carrera

Maestría

Nivel

Figura 1 Ejemplo de aplicación tradicional.



Figura 2 Ejemplo de aplicación enriquecida.

Para que una aplicación se muestre también es común contar con un acceso restringido por clave y contraseña y solo entonces permitir la visualización de los datos. Este comportamiento se modela en IFML con un diagrama de interacción como el que se muestra en la figura 3 (caso enriquecido) y un diagrama de dominio que se ilustra en la figura 4, estos diagramas se realizaron con la ayuda de la herramienta WebRatio (WebRatio, 2017) en su versión comunitaria.

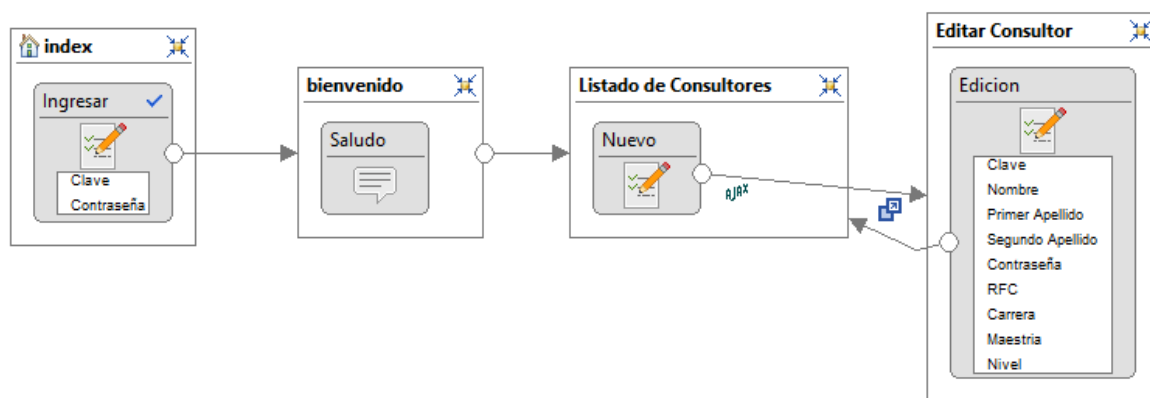


Figura 3 Diagrama de Interacción.

En el diagrama de interacción se indica que la aplicación requiere mostrar cuatro vistas distintas, que el usuario requiere ingresar en un formulario una clave y una contraseña correctas para llegar a una pantalla de bienvenida con un texto y la opción de visualizar la lista de consultores, donde se cuenta con la posibilidad de editar un consultor. El hecho de que la edición se realice con un diálogo modal no es visible directamente en la herramienta WebRatio aunque IFML sí describe la manera de indicarlo visualmente.

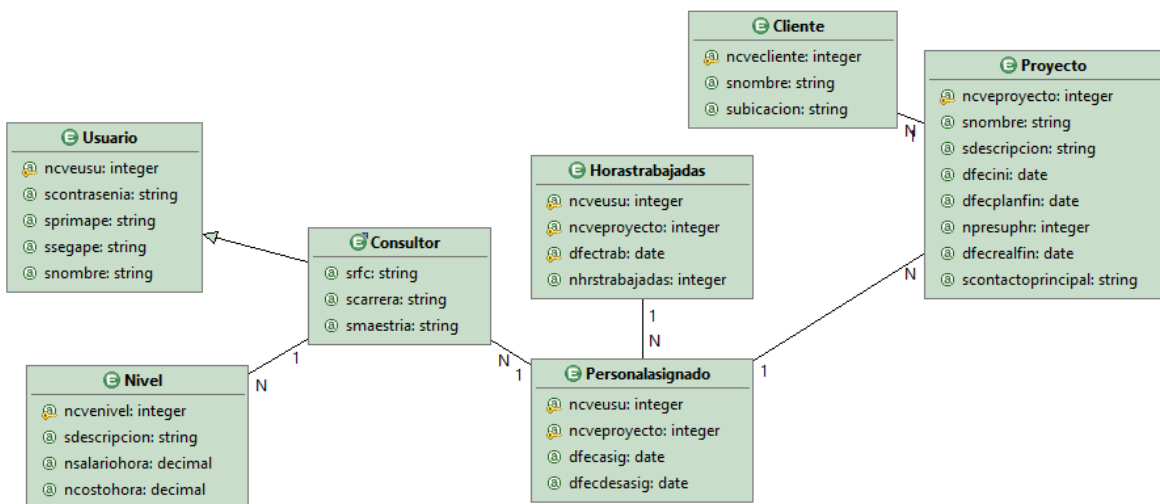


Figura 2 Diagrama de Dominio.

Por otra parte, en el diagrama de dominio se presentan los conceptos principales del dominio del negocio indicando los atributos que los forman así como sus tipos de datos y qué atributos funcionan como identificadores únicos de una instancia.

5. Conclusiones

El modelado de la interacción de los usuarios con una aplicación Web, tradicional o enriquecida, permite identificar en etapas tempranas del desarrollo los requerimientos de navegación que será necesario implementar y tomar en consideración durante las etapas de diseño y construcción. El uso de IFML para modelar tanto la interacción como el dominio de la información a mostrar cubre un aspecto que se había dejado de lado en otros lenguajes de modelado como UML.

Sin embargo, el modelo por sí mismo no es suficiente para disminuir la resistencia que existe en los equipos de desarrollo respecto al consumo de tiempo en etapas tempranas

respecto al código, por lo que es importante que el modelo creado se vea reflejado rápidamente en líneas de código que simplifiquen el desarrollo de aplicaciones.

6. Trabajo a Futuro

Los siguientes pasos incluyen analizar tanto las herramientas existentes para modelar IFML, la posibilidad de obtener los modelos en algún lenguaje que funcione como entrada para generadores de código que transformen los modelos en código fuente útil durante el desarrollo de aplicaciones.

7. Agradecimientos

Al Tecnológico Nacional de México, por el apoyo económico otorgado para la realización de la investigación.

8. Referencias

Adobe (2016) Adobe - Rich Internet Applications. Recuperado de https://www.adobe.com/resources/business/rich_internet_apps

Brambilla, M.; Fraternali, P. (2015) Interaction Flow Modeling Language. Morgan Kaufmann.

Brambilla, M. et al. (2014) IFML: Building the front end of Web and Mobile applications with OMG's Interaction Flow Modeling Language. International Conference on Web Engineering ICWE, 2014

Dong, S. et al. (2011) Research on AJAX technology application in web development. International Conference on E-Business and E-Government ICEE.

IFML (2016) Sitio Oficial. Recuperado de <http://www.ifml.org>

Laaz, N.; Mbarki, S. (2016) Integrating IFML models and owl ontologies to derive UIs web-Apps. International Conference on Information Technology for Organizations Development (IT4OD)

Shklar, L. (2003) Web Application Architecture: Principles, Protocols and Practices, 1 ed., Wiley

WebRatio (2017) Sitio Oficial. Recuperado de <http://www.webratio.com>