

Propuestas metodológicas para el desarrollo de aplicaciones Web: una evaluación según la ingeniería de métodos

Methodological proposals for Web application developments: a method engineering evaluation

M. Mendoza* y J. Barrios**
Postgrado en Computación,
Universidad de Los Andes, Facultad de Ingeniería,
Escuela de Ingeniería de Sistemas,
Mérida, Venezuela.
*nella_1@hotmail.com,
**ijudith@ing.ula.ve

Resumen

La mayoría de los métodos y propuestas metodológicas que existen para desarrollar aplicaciones Web guían al grupo de desarrollo a través de un conjunto de fases y pasos predefinidos sin tomar en cuenta la situación particular, como por ejemplo: elementos del contexto de desarrollo, tipo de aplicación, usuarios, herramientas, tecnología, experiencia de desarrollo, etc. de cada proyecto. Por tal razón, en este artículo se presenta una caracterización de cuatro métodos (seleccionados de veinticinco que fueron revisados) con el fin de determinar cual es su flexibilidad y su capacidad de adaptación a situaciones particulares o contextos de modelado dentro de un dominio de aplicación, atendiendo los conceptos y principios de la ingeniería de métodos e ingeniería Web.

Palabras claves: Aplicaciones Web, métodos de desarrollo de aplicaciones Web, métodos orientados al contexto.

Abstract

Most of the methods and methodological proposals that exist to develop Web applications guide the group of development through a predefined set of phases and steps without taking into account particular project situations, such as the development context, the type of application, the users, the available tools and technologies, the team development experience, etc. For such reason, in this article a characterization of four methods (selected from twenty-five that were reviewed) with the purpose of determining what is their flexibility and capacity to be adapted to particular situations or contexts within an application domain. The evaluation is based on method engineering and Web engineering concepts and principles.

Key words: Web applications, Web applications development methods, context-oriented methods.

1 Introducción

Una aplicación Web es un sistema de software cuya funcionalidad es mostrada a través de la plataforma World Wide Web (WWW). Desarrollar este tipo de aplicaciones es un proceso muy complicado que demanda amplios conocimientos técnicos y experiencia por parte del personal involucrado. Además, este proceso de desarrollo no parece estar acompañado de métodos adecuados que garanticen la calidad de las aplicaciones Web. Un método para desarrollar aplicaciones Web debe estar conformado por un modelo de

producto, que describe las características genéricas de una aplicación Web y un modelo de proceso, que indica las actividades necesarias para desarrollar este tipo de aplicaciones. En la actualidad, han surgido una gran cantidad de métodos bien conocidos para desarrollar aplicaciones Web, entre los cuales se pueden mencionar: métodos ágiles, métodos para aplicaciones hipermedia, métodos para el desarrollo de sitios Web instruccionales, métodos el desarrollo de aplicaciones de e-commerce y métodos para el desarrollo de aplicaciones Web en general.

Estos métodos poseen similitudes y diferencias que los

distinguen unos de otros. El propósito de este artículo es compararlos, y determinar cuáles son las deficiencias que poseen los modelos de proceso y de producto. En general, estos modelos de proceso, permiten guiar al grupo de desarrollo a través de un conjunto de fases y pasos predefinidos, lo cual dificulta su fácil adaptación a múltiples contextos, obligando así a que las decisiones se tomen basándose en la experiencia de los desarrolladores.

El resto del artículo está estructurado como sigue. La sección 2 describe brevemente la definición, características, atributos de calidad y tecnologías que son utilizadas en una aplicación Web en general. Además, se presenta una categorización de las aplicaciones Web, y se propone una manera de clasificarlas, tomando en cuenta el dominio de aplicación para el cual se desarrollan. La sección 3 muestra un compendio de algunos métodos que han sido propuestos para el desarrollo de aplicaciones Web. La sección 4, presenta un modelo de producto genérico para una aplicación Web, el cual describe, basado en el estilo arquitectural de las tres capas, las características generales de una aplicación Web. La sección 5, presenta una evaluación de cuatro métodos seleccionados luego de haber revisado un aproximado de veinticinco. Para realizar la evaluación se definió un marco de referencia basado en las ideas de (Jarke, 1990) y (Barrios y Nurcan, 2002). Finalmente, en la sección 6 se presentan la discusión de los resultados de la evaluación realizada y posibles trabajos futuros.

2 Aplicaciones Web e ingeniería Web

Una aplicación Web consiste en un software basado en Internet, en el cual una población extensa de usuarios, por medio de un navegador, hacen peticiones remotas y esperan una respuesta que puede implicar una mezcla de publicación impresa y desarrollo de software, de mercadeo e informática, de comunicaciones internas y relaciones externas, y de arte y tecnología. Una aplicación Web se distingue por utilizar hipertexto para presentar al usuario el contenido de textos, gráficos, sonido y video, por su actualización constante, por su inmediatez y por la capacidad de interactuar con otros elementos de Internet, o efectuar transacciones automáticas en otros portales de Internet, etc.

El desarrollo de aplicaciones Web es más que un medio de manipulación y creación de presentaciones, ya que además debe incluir análisis de necesidades, diseño, gerencia, métricas, mantenimiento, etc. (Ginige y Murugesan, 2001). Los desarrolladores Web usan frecuentemente enfoques ad hoc lo que ocasiona pérdida de rigor, de técnicas sistemáticas, de metodologías sólidas y de aseguramiento de la calidad. Por esta razón, surge una nueva disciplina para el desarrollo de aplicaciones Web de alta calidad: la ingeniería Web. La ingeniería Web es el establecimiento y uso de conocimiento científico, principios de gerencia e ingeniería y el enfoque sistemático y disciplinado para desarrollar, entregar y mantener, aplicaciones y sistemas basados en Web de alta calidad (Ginige y Murugesan, 2001).

Sin embargo, las características señaladas anteriormente sólo diferencian una aplicación Web de una local, pero no proporcionan una base útil para evaluar la calidad de las mismas. Es por ello, que Olsina, Lafuente y Pastor (Olsina, Lafuente y Pastor, 2002) presentaron un "árbol de requisitos de calidad" que identifica un conjunto de atributos que debe contener una aplicación Web de alta calidad: usabilidad, funcionalidad, fiabilidad, eficiencia y capacidad de mantenimiento. Este "árbol de requisitos de calidad" está basado en el estándar ISO 9126 (ISO/IEC 9126, 1991), el cual provee un marco de referencia para evaluar la calidad de cualquier tipo de software.

En la actualidad existe una gran variedad de aplicaciones Web, que van desde páginas informativas hasta aplicaciones complejas que ofrecen diversidad de servicios al usuario. Por tal razón, (Ginige y Murugesan, 2001), agrupa las aplicaciones Web según el grado de complejidad en informativas, interactivas, transaccionales, workflow, ambientes de trabajo colaborativo, mercados y comunidades en línea y portales Web. Hemos decidido evaluarlas según el tipo de aplicación, ya que cada una de ellas tienen un propósito definido que permite diferenciarlas entre sí. Esta clasificación permite catalogarlas por tipo de aplicación, quedando definidas las siguientes categorías: Web Informativas (Pressman, 2001), Sitios Web instruccionales (Montilva, Sandia y Barrios, 2001), Portales (Oliveira, 2000), Comercio electrónico (e-commerce) (Oliveira, 2000), Negocios inteligentes (e-business) (Oliveira, 2000), Sistemas de Información Web (SIW) (Torrealba, 1998) y Gobiernos electrónicos (e-government).

3 Métodos para el desarrollo de aplicaciones Web

Según los principios que rigen la ingeniería de métodos (Barrios, 2002), un método está conformado por un modelo de producto y un modelo de proceso. El modelo de producto es una abstracción de las propiedades comunes que se encuentran en cualquier producto de desarrollo. Mientras que el modelo de proceso, especifica el conjunto de acciones requeridas para desarrollar un producto de acuerdo al modelo de producto.

En los últimos años ha surgido un conjunto de métodos para desarrollar aplicaciones Web, los cuales no toman en cuenta estos principios, ni tampoco los que son proporcionados por la ingeniería Web. Estos métodos presentan en forma explícita su modelo de proceso, es decir, la integración de las actividades técnicas y gerenciales que son requeridas para el desarrollo de una aplicación Web de alta calidad, y de manera implícita, describen el producto, es decir, las características arquitecturales de las aplicaciones Web.

Los métodos que a continuación se presentan se han agrupado de acuerdo a su modelo de proceso y al contexto particular donde pueden ser aplicados. Es necesario aclarar que, ninguno de ellos guía al grupo de desarrollo en la construcción de aplicaciones Web para múltiples contextos, sin depender de la experiencia de los miembros de dicho grupo.

Además, los mismos no describen explícitamente al modelo de producto y su integración con el modelo de proceso, lo cual representa una limitación durante el proceso de desarrollo de aplicaciones Web. Entre los métodos más conocidos, se encuentran los siguientes:

3.1 Procesos ágiles para el desarrollo de aplicaciones

Los métodos ágiles para el desarrollo de software están caracterizados por poseer iteraciones cortas, pruebas continuas, equipos autoorganizados, constante colaboración y frecuente replanificación basada en la realidad actual. El desarrollo ágil no abandona la estructura, pero intenta lograr un balance entre el caos y la rigidez. En la actualidad existen varias metodologías basadas en este enfoque, entre las más conocidas se tienen: XP (Extreme Programming) (Acebal, 2002), Open Source (Cáceres y Marcos, 2001), Dynamic Systems Development Method (DSDM) (Cap Ernst, 2002).

3.2 Métodos para el desarrollo de sistemas de información Web (SIW)

Estos métodos se caracterizan por seguir una secuencia de fases y pasos requeridos para desarrollar SIW, y asegurar la calidad del mismo. Cubren todo el ciclo de desarrollo de un SIW, emplean técnicas de análisis y diseño orientado a objetos y utilizan un enfoque iterativo. Entre las más conocidos se mencionan: Metodología para desarrollar Sistemas de Información Web (MIDAS) (Cáceres y Marcos, 2001) y el Método para desarrollar SIW (Torrealba, 1998).

3.3 Métodos para el desarrollo de aplicaciones hipermedia

La mayoría de estos métodos sólo cubren parcialmente el ciclo de desarrollo de las aplicaciones hipermedia, dando especial importancia al diseño. Básicamente utilizan dos técnicas en cualquier diseño de aplicaciones hipermedia: Modelo Entidad-Relación y técnicas de Orientación a Objetos. Los más difundidos son: Aplicando modelos de proceso de software al desarrollo de aplicaciones hipermedia. (Montilva, 1996), The Object-Oriented Hypermedia Design Model (OOHDM) (Costagliola, Ferrucci y Francese, 2002), Relationship Management Methodology (RMM) (Costagliola, Ferrucci y Francese, 2002), Hypermedia Flexible Process Modeling (HFPM) (Costagliola, Ferrucci y Francese, 2002) y el Enfoque de Ingeniería de Lowe-Hall's (Costagliola Ferrucci y Francese, 2002).

3.4 Métodos para el desarrollo de sitios Web instruccionales

Este tipo de métodos incorpora una gran variedad de componentes organizacionales, administrativos, instruccionales y tecnológicos con el fin de garantizar la calidad del producto. El objetivo de los mismos es ayudar a los diseñadores

de los cursos y profesores a desarrollar sitios Web instruccionales de alta calidad. Además, estos métodos consideran el elemento humano (aprendices, profesores, tutores, administradores), los recursos de aprendizaje basados en Web, otros recursos de aprendizaje (textos, guías, etc) y la infraestructura tecnológica necesaria para desarrollar el proceso instruccional. También hacen énfasis en la reutilización de componentes para reducir el tiempo y costo de desarrollo. Los más conocidos son: Desarrollo de sitios Web instruccionales – Un Enfoque de Ingeniería de Software (Montilva, Sandia y Barrios, 2002), Simple Web Method (SWM) (Cormack, Griffiths y otros, 2002) y el Modelling Web-Based Instructional Systems (Retalis y Avgeriou, 2002).

3.5 Métodos para el desarrollo de aplicaciones de comercio electrónico: (e-commerce)

Estos métodos están basados en la reutilización de componentes de software y en la integración de los mismos para lograr funcionalidades empresariales para aplicaciones de e-commerce (negociación, mediación; “workflow” inter.-empresarial y notificaciones de eventos). Uno de los más difundidos es el Marco de Referencia Basado en componentes para e-commerce (Fingar, 2000).

3.6 Metodologías para el desarrollo de aplicaciones Web

Estos métodos poseen ciclos de desarrollo cortos, centrandose su mayor esfuerzo en la apariencia del contenido y en el diseño de la estructura de navegación. Los mismos buscan que la aplicación sea entregada en el menor tiempo posible para asegurar su usabilidad, durabilidad y evolución continua de acuerdo a las necesidades de los usuarios. Además, permiten integrar varias tecnologías como la reutilización de componentes y las técnicas de análisis y diseño orientadas a objeto. Entre este tipo de métodos se mencionan: Método basado en componentes para el desarrollo de aplicaciones Web (Montilva y Barrios, 2002), Web Site Design Method (WSDM) (Escalona y Hoch, 2002), Relationship-Navigational Análisis (RNA) (Escalona y Hoch, 2002), UML-based Web Engineering Approach (UWE) (Koch, 2001) y Un método de producción de software en ambientes Web: Object-Oriented Web-Solution (OOWS) (Pastor y Fons, 2002).

4 Modelo de producto de una aplicación Web

El modelo de producto es una abstracción de las propiedades comunes que se encuentran en cualquier producto desarrollado con el uso de un método (Barrios 2002). El modelo de producto que describe a una aplicación Web genérica, permite capturar los conceptos y patrones arquitecturales genéricos de cualquier aplicación Web (Montilva y Barrios, 2002). Estos conceptos, al ser identificados contextualmente, permiten definir un modelo de proceso

flexible y ajustable a un contexto particular de desarrollo.

Con el objeto de conceptualizar una aplicación Web y considerando que en la literatura actual, la mayoría de los modelos de producto están definidos implícitamente, se diseñó un modelo de producto genérico que representa los conceptos más comúnmente encontrados en una aplicación Web. Este modelo de producto está organizado en tres capas (Fig. 1) presentación, lógica del negocio y datos. De acuerdo a este modelo de producto se dice que el desarrollo de una aplicación Web involucra la especificación, el diseño, la implementación y la prueba de tres grupos de componentes distribuidos en cada una de las capas.

La capa de presentación describe cómo la información debe ser desplegada. La misma está conformada por el navegador y el servidor Web. El navegador contiene páginas HTML, programas intérpretes (JavaScript, VBScript) y programas compilados (Applets de JAVA), para facilitar la validación de los datos, ayudar en la navegación, y animar la interfaz del usuario. El servidor Web contiene un servidor de páginas dinámicas que posee código embebido (SERVLETS, CGI, ASP, JSP) para preparar el formato de las páginas HTML que son requeridas por el navegador y solicitar a la capa de lógica del negocio los servicios que son demandados por el usuario.

La capa de lógica del negocio crea la correspondencia entre los datos almacenados y la información que será presentada al usuario luego de haber solicitado un servicio a través del servidor Web, es decir, actúa como un middleware para soportar la integración y comunicación entre los componentes (J2EE, .NET, COM). Además, esta capa contiene las clases de objetos que son propios del negocio (un carrito de compras, un cliente, etc.).

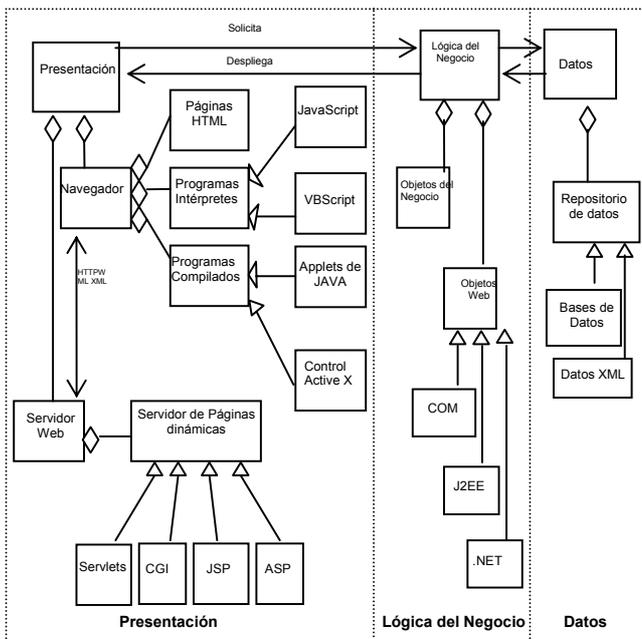


Fig. 1. Modelo de producto de una aplicación Web en general.

Por último, la capa de datos se encarga de describir la representación física de los datos. La misma está compuesta por los repositorios de datos que pueden ser bases de datos relacionales, orientadas a objetos y/o mixtas o datos XML.

5 Evaluación de métodos

Generalmente, la evaluación de métodos va acompañada de marcos o cuadros de referencia que permiten calificar y posicionar un método con respecto a otros, considerando un conjunto de características o atributos que éstos deben cumplir: Una estructura del proceso bien definida para facilitar el desarrollo de la aplicación Web, un modelo claro y explícito que indique las características del producto que se desea desarrollar, la calidad de la aplicación Web producida, que los recursos requeridos: humanos, económicos, tecnológicos, etc. se utilicen de manera óptima en el contexto del desarrollo y una orientación contextual de manera tal que el método sea flexible y adaptable para ser utilizado fácilmente en dominios de aplicación diferentes.

El marco de referencia diseñado está basado en los trabajos de (Jarke, 1992) y (Barrios, 2001). Este marco usa los principios de la ingeniería de métodos, los cuales establecen que un método debe tener especificado un propósito, un dominio, un tipo de problema a ser solucionado, y un conjunto de condiciones o restricciones de uso. Otro principio establece que todo método debe estar conformado por un modelo de producto y un modelo de proceso. El modelo de producto especifica los conceptos que manipula el método, el modelo de proceso especifica cómo manipularlos.

El marco de referencia considera cuatro aspectos fundamentales para evaluar un método: la vista del dominio, la vista de uso, la vista del producto y la vista del proceso. Dentro de estas vistas se consideran a su vez las facetas y los atributos que las describen. Cada faceta representa un aspecto a evaluar dentro de una vista. La misma está compuesta por un conjunto de atributos y sus valores correspondientes, los cuales fueron adaptados con el fin de caracterizar los métodos para el desarrollo de aplicaciones Web.

De los métodos descritos en la sección 3, se seleccionaron los cuatro siguientes: (1) Método basado en componentes para el desarrollo de aplicaciones Web (Montilva y Barrios, 2002), (2) Marco de referencia basado en componentes para e-commerce (Fingar, 2000), (3) Desarrollo de sitios Web instruccionales – Un enfoque de ingeniería de software (Montilva, Sandia y Barrios, 2002) y (4) XP (Extreme Programming) (Acebal y Cueva, 2002). El propósito para el cual se diseñaron los métodos y la disponibilidad y completitud de la documentación fueron los criterios utilizados para realizar esta selección. A continuación se detallan los resultados de la evaluación según las vistas de referencia:

5.1 La vista de dominio

La vista del dominio está descrita a través de la faceta alcance. Esta faceta contempla todos los atributos relacionados con el área de aplicación: los factores técnicos, gerenciales y/o aquellos que están relacionados específicamente al dominio de aplicación.

La evaluación de los métodos, desde este punto de vista se puede observar en la Tabla 1. En el atributo área de aplicación se puede observar que los métodos (2) y (3) han sido diseñados para un dominio de aplicación en particular: e-commerce y sitios Web instruccionales, respectivamente. Sin embargo, los métodos (1) y (4) fueron diseñados para desarrollar cualquier tipo de aplicación Web, por lo tanto pueden ser utilizados en cualquiera de los dominios de aplicación considerados, dependiendo siempre de la experiencia del grupo de desarrollo.

Los métodos (1) y (3) cubren los aspectos técnicos, gerenciales y relacionados con el dominio de aplicación (Aplicaciones Web y Sitios Web Instruccionales), lo cual permite guiar de una forma más clara al usuario para adaptar la aplicación a desarrollar al dominio de aplicación. A diferencia de los métodos (2) y (4) que únicamente cubren aspectos técnicos, dejando fuera los aspectos gerenciales y los relacionados con el dominio de aplicación, lo que dificulta el desarrollo del proyecto y no garantiza que la aplicación se adapte a un dominio de aplicación específico.

5.2 La vista de uso

Esta vista está descrita a través de las facetas de aplicabilidad, uso y participación del usuario en el desarrollo de la aplicación Web.

La faceta de aplicabilidad evalúa las fases que cubren el ciclo de vida y la orientación que se le dará a la aplicación Web. El uso describe las características relacionadas con el uso del método: visibilidad, estandarización y eficiencia de uso. La faceta participación del usuario especifica los tipos de usuario esperado y su grado de participación en el desarrollo de aplicaciones Web. El resultado de evaluar estos aspectos se observa en la Tabla 2.

Los cuatro métodos cubren las fases del ciclo de vida de una aplicación Web (análisis, diseño, desarrollo y evaluación). Ninguno de los métodos evaluados cubre la etapa de mantenimiento, lo que no garantiza que las aplicaciones Web estén siempre actualizadas. Sin embargo, el método (4) cubre las etapas de manera ligera y superficial dándole la mayor importancia a las pruebas del software.

Cada método está orientado a un uso particular; el (1) Servicios, comercial el (2) comercial el (3) académico, pero el (4) no posee una orientación definida, ya que el mismo no está diseñado para ser usado en el desarrollo de alguna aplicación en específico.

Otro aspecto importante a considerar es la faceta de uso. Todos los métodos muestran qué hacer pero no todos enseñan cómo se tiene que hacer. Esto está relacionado con

la vista de proceso, en la que se observa que todo los métodos presentan una estructura de modelo de proceso bien clara y definida a través de un conjunto de fases, pasos y tareas, pero no indican en qué situaciones particulares deben ser ejecutadas, ni cómo deben ser aplicadas, lo cual depende de la experiencia del grupo de desarrollo que esté involucrado en cada proyecto particular. En cuanto a la eficiencia de uso, el método (1) es bastante rígido, por lo que es poco adaptable y flexible, asimismo, el método (2) no se considera fácil de usar porque no indica cómo hacer las cosas y no está bien documentado.

5.3 La vista del producto

Esta vista evalúa el modelo de producto usado por el método. Dos facetas son seleccionadas para describir el modelo del producto: la descripción del producto, que caracteriza al mismo por su notación, orientación y perspectiva; y la descripción conceptual la cual describe los conceptos que son usados por el modelo.

La evaluación de esta vista se puede observar en la Tabla 3. Es indispensable que un método disponga de modelo de producto con el fin de garantizar el entendimiento del resultado que se obtendrá por el uso del método. Sin embargo, el método (4) no cuenta con un modelo de producto, los métodos (1) (2) y (3) poseen un modelo explícito con notación semi-formal (gráficos e imágenes) e informal (texto) que permite caracterizar cada uno de los productos que se generarán al utilizar dicho método.

Tabla 1. Vista del dominio (¿para qué sirve? ¿para qué es concebido?)

Faceta	Atributos	Valores	1)	2)	3)	4)
Alcance	Área de aplicación	Hipermedia				
		Educativo				
		e-commerce				
		e-business				
		Portales				
		Sistemas de Información Web				
	Aspectos	e-government				
		Técnicos				
		Gerenciales				
		Inherentes al área de Aplicación.				

Tabla 2. Vista de uso (objetivos del que los usa)

Faceta	Atributos	Valores	1)	2)	4)	5)
Aplicabilidad	Ciclo de vida	Análisis				
		Diseño				
		Desarrollo				
		Evaluación				
		Mantenimiento				
	Orientación	Académico				
		Comercial				
		Publicitario				
		Servicios				
		Uso	Visibilidad	Siempre muestra qué hacer		

	Estandarización	Siempre muestra cómo hacerlo
		Uso de técnicas estándares
		Uso de notaciones estándares
	Eficiencia de uso	Adaptable y expandible
		Simple y fácil de usar
		Bien documentado
Participación del usuario	Tipo de usuario	Desarrolladores de software
		Diseñadores de páginas Web
		Publicistas
		Personal directivo de grandes organizaciones
	Grado de Participación	Total
	Parcial	
	Ninguno	

Tabla 3. Vista del producto (resultado de aplicar el método)

Faceta	Atributos	Valores	1)	2)	3)	4)	
Descripción del producto	Modelo del producto	Explicito					
		Implícito					
		No existe					
	Notación del modelo	Formal (Fórmulas matemática)					
		Semi-formal (Gráficos, imágenes)					
		Informal (Textual)					
	Orientación del modelo	Orientado a objetos					
		Orientado a procesos					
		Basado en ejemplos					
	Perspectiva del producto	Indefinido					
		Perspectiva funcional					
		Perspectiva tecnológica					
Descripción conceptual	Conceptos usados por el modelo	Perspectiva estructural					
		Conceptos técnicos					
		Conceptos estéticos					

5.4 La vista del proceso

Esta vista contiene un conjunto de facetas y atributos que describen el proceso de desarrollo de la aplicación que se desea producir. Se evalúa el modelo de proceso usado por el método a través de cinco facetas que consideran la orientación y el enfoque, las características y los procesos cubiertos por el modelo. La Tabla 4 muestra los resultados de esta evaluación.

Todos los métodos evaluados siguen un enfoque evolutivo (adaptativo), lo que se considera una gran fortaleza, debido a que permiten hacer cambios de requerimientos o de diseño a medida que se va evaluando la aplicación con el usuario y/o cliente. Todos los métodos presentan una estructura del proce-

so bien definida, sin embargo, el (2) y el (4) no dividen sus fases en pasos y tareas, lo que hace que las actividades no se indiquen de manera clara y explícita al grupo de desarrollo.

Los métodos (2) y (4) no consideran la gerencia del proyecto, lo que quiere decir que no incluyen dentro del modelo de proceso las actividades de planificación y control, organización del equipo de trabajo, gestión de la configuración del software y documentación. El único aspecto que ambos métodos consideran es la revisión de técnicas de validación y verificación, debido a que todos ejecutan el proceso en forma iterativa y siguen enfoques evolutivos.

En cuanto a la faceta de cobertura del ciclo de desarrollo, el método (4) cubre en forma muy ligera y simple, cada una de las etapas, pero no contempla un análisis de objetos del dominio. Este sólo realiza una definición informal de requerimientos, sin profundizar en su definición.

El método propone incorporar los cambios en la medida en que éstos van apareciendo. El diseño es muy general y ligero y no contempla la etapa de implementación como tal, su mayor importancia radica en las pruebas, las cuales son rigurosas y se realizan de cada uno de los componentes, de integración, de la aplicación y finalmente una validación y verificación de la aplicación.

El método (2) cubre la etapa de análisis, definición de requerimientos; dándole especial importancia al diseño de las arquitecturas y a la comunicación de los objetos distribuidos. Además, cubre la etapa de pruebas de cada uno de los componentes, las de integración, las cuales son las de mayor atención, y las de la aplicación.

Por último, se puede observar que los métodos (1) y (3) son los que cubren completamente las etapas del ciclo de desarrollo.

El método (1) no guía al grupo de desarrollo en lo que a construcción de bases de datos se refiere, tampoco considera mecanismos de seguridad. Ningún método cubre la etapa de mantenimiento del proceso de post-desarrollo.

Tabla 4. Vista del proceso (desarrollo del producto)

Faceta	Atributos	Valores	1)	2)	3)	4)	
Modelo de proceso de desarrollo	Orientación	Orientada al producto					
		Orientada a la actividad					
		Orientada a la decisión					
	Enfoque	Ad-hoc					
		Predictivo (cascada)					
		Evolutivo (adaptativo)					
Características del proceso	Claridad	Estructura del proceso bien definido					
		Estructura del proceso (qué hacer)					
	Modo de ejecución del proceso	Iterativo					
		Secuencial estricto					

Gerencia de proyecto	Visibilidad	Incremental	Construcción de la Base de Datos		
		Definición clara del flujo de trabajo		Implementación de las reglas del negocio	
	Progreso visible externamente	Programación del lado del cliente			
	Dependencia tecnológica	Independiente de las herramientas		Programación del lado del servidor	
	Adaptabilidad	Modularidad		Implementación de políticas de seguridad (SSL, etc).	
		Procesos modificables			Comunicación de objetos distribuidos.
	Guía	explícita		Pruebas	
		Completa			Prueba de cada uno de los componentes
		Implícita			Pruebas de integración
	Planificación y control	Definición del plan del proyecto		Prueba de la aplicación Web.	
		Estimación de recursos			Validación y verificación de la aplicación Web.
	Monitoreo y evaluación del progreso	Monitoreo y evaluación del progreso		Implantación de la aplicación Web	
		Equipo de trabajo			Soporte de mantenimiento
	Tamaño del equipo	Estructura del equipo		Proceso de post-desarrollo	
Individual		Instalación			
Equipos pequeños (<= 5 personas)		Mantenimiento			
Aseguramiento de calidad	Equipos Grandes (> 5 personas)	Gerencia de proyecto			
	Calidad en la planificación y control		Calidad del equipo		
	Revisión de técnicas de validación y verificación				
Gestión de la configuración	Planificación y ejecución de la gestión de configuración del software	Cobertura del ciclo de desarrollo			
Entrenamiento	Documentación de plantillas		Análisis		
	Entrenamiento de usuario			Análisis de objetos del dominio.	
Definición de requerimientos	Entrenamiento de equipo	Diseño			
	Entrenamiento de usuario		Diseño de arquitecturas		
Implementación	Definición de requerimientos informal	Implementación			
	Técnicas de especificación		Producción de multimedia		
Implementación	Diseño de interfaces	Implementación			
	Creación del sitio Web		Creación del sitio Web		

6 Discusión y trabajo futuro

Con la evaluación realizada de los cuatro métodos señalados anteriormente, se cubrió el objetivo fundamental de esta investigación: caracterizar los elementos de los métodos para desarrollar aplicaciones Web que cumplen con los principios de la ingeniería de métodos y de la ingeniería Web. Los métodos evaluados anteriormente, han sido diseñados para ser utilizados en dominios de aplicación particulares, a excepción del método basado en componentes para el desarrollo de aplicaciones Web (Montilva y Barrios, 2002), y el de la programación Extrema (XP), los cuales se pueden adaptar a cualquier tipo de aplicación Web, dependiendo siempre de la experiencia del grupo de desarrollo. Sin embargo, este último cubre las etapas de desarrollo de manera ligera y superficial dándole mayor importancia a las pruebas del software. Los métodos (1), (2) y (3) poseen un modelo de producto explícito que permite describir los productos que se desarrollarán. El método (4) no presenta este modelo lo que dificulta el entendimiento del resultado que se debería obtener a través del uso del mismo.

Todos los métodos presentan una estructura bien definida. Sin embargo, el (2) y el (4) no dividen las fases en pasos y tareas, lo que no garantiza que las actividades se indiquen de manera clara y explícita al grupo de desarrollo.

Los métodos (2) y (4) no consideran la gerencia del proyecto, es decir, que no garantizan el uso óptimo de los re-

cursos requeridos (humanos, económicos, tecnológicos, etc.) en el contexto del desarrollo.

Finalmente, cabe destacar que ninguno de los métodos evaluados está orientado al contexto, es decir, que solo son un conjunto de pasos lógicamente ordenados para guiar el desarrollo de una aplicación Web dentro de un dominio y para contexto predeterminado.

Por tal razón, como trabajo futuro, se pretende diseñar un método orientado al contexto atendiendo a los principios de la ingeniería de métodos y de la ingeniería Web. Este método ofrecerá la posibilidad, al grupo de desarrollo, de elegir entre múltiples alternativas aquella que es más conveniente para un dominio de aplicación y una situación particular de proyecto de desarrollo; por ejemplo, los elementos del contexto de desarrollo, el tipo de aplicación, los usuarios, las herramientas y tecnología, la experiencia de desarrollo, etc.

Este método, a diferencia de los que existen actualmente, contendrá una descripción clara y explícita del modelo de producto y del modelo de proceso, mostrando la integración entre ambos. Además, se adaptará a diferentes contextos de modelado, a través de la toma de decisiones según la situación particular enfrentada en cada proyecto; y garantizará una guía completa al equipo de desarrollo, tomando en cuenta las necesidades particulares del dominio de aplicación seleccionado (comercio electrónico, aplicaciones instruccionales, sistemas de información Web, etc.).

Referencias

- Acebal C y Cueva J, 2002, Extreme programming (XP): un nuevo método de desarrollo de software, Universidad de Oviedo, Dpto. de Informática, Area de lenguajes y sistemas informáticos.
- Barrios J y Surcan S, 2002, MEDIC: A Method engineering proposal for the analysis and representation of the organizational impact of change, SERP'2002, Las Vegas.
- Barrios J, 2002, Material del seminario de ingeniería de métodos, Universidad de Los Andes, Postgrado en Computación.
- Cáceres P y Marcos E, 2001, Procesos ágiles para el desarrollo de aplicaciones Web, Universidad Rey Juan Carlos, Dpto. de Ciencias Experimentales e Ingeniería.
- Cormack G, Griffiths G y otros, 2002, Web engineering: methods & tools for education, School of Computing & Mathematics, University of Teesside Middlesbrough.
- Costagliola G, Ferrucci F y Francese R, 2002, Web engineering: models and methodologies for the design of hypermedia applications, Dipartimento di Matematica e Informatica, Università di Salerno, Italia.
- Escalona M y Koch N, 2003, Ingeniería de requisitos en aplicaciones para la Web: un estudio comparativo, Universidad de Sevilla, Lenguaje y sistemas informáticos, España.
- Fingar P, 1999, Marco de referencia basado en componentes para e-commerce, ACM.
- Gemini E y Young, 2002, Dynamic systems development a recipe for repeatable success, DSDM competence center.
- Ginige A y Murugesan S, 2001, Web engineering: an introduction, Universidad de Western Sidney, Australia.
- ISO/IEC 9126; ISO 9126: The standard of reference, information technology, software product evaluation, quality characteristics and guidelines for their use.
- Jarke M, 1992, A proposal for a scenario classification framework, RWTH, Lehrstuhl Informatik Germany.
- Koch N, 1999, A comparative study of methods for hypermedia development, Technical report 9905, Ludwig-Maximilians-Universität München.
- Koch N y Kraus A, 2002, The expressive power of UML-based Web engineering, Ludwig-Maximilians-Universität München, Germany.
- Koch N y Wirsing M, July 2001, Software engineering for adaptive hypermedia applications?, Third workshop adaptive hypertext and hypermedia at the 8th international conference on user modeling, Southofen, Germany.
- Marcos E, 2002, MIDAS: A methodology for Web information systems development, Rey Juan Carlos University, Kybele Research Group.
- Montilva J, Sandía B y Barrios J, 2001, Developing instructional Web sites, A software engineering approach, Universidad de Los Andes, Postgrado en Computación.
- Montilva J y Barrios J, 2002, A component-based method for developing Web applications, Universidad de Los Andes, Facultad de Ingeniería, Escuela de Ingeniería de Sistemas, Dpto. de Computación, Mérida, Venezuela.
- Montilva J, 1996, Aplicando modelos de procesos de software al desarrollo de aplicaciones hipermedia. Universidad de Los Andes, Facultad de Ingeniería.
- Olsina L, Lafuente G y Pastor O, 2002, Towards a reusable repository for Web metrics, Journal of web engineering, Vol. 1, No. 1, pp 061-073.
- Pastor O y Fons J, 2002, OOWS: Un método de producción de software en ambientes Web, Departamento de Sistemas Informáticos y Computación, Universidad Politécnica de Valencia.
- Pressman R, 2002, Ingeniería de software: un enfoque práctico, Quinta edición, Mc Graw Hill.
- Prince 2, 2002, Dynamic systems development method overview; BBs Ltd.
- Retalis S y Avgeriou P, 2002, Modeling Web-based instructional systems, Journal of Information Technology Education, Vol. 1 No. 1.
- Torrealba JC, 1998, Método para el análisis y diseño de sistemas de información Web (SIW), Trabajo especial de grado para optar al título de Magíster en Computación, Universidad de Los Andes, Facultad de Ingeniería, Postgrado en Computación.